



A Database Publication

apple user

Vol. 4. No. 4. April 1984 £1

-the new name for
Windfall

**TK! Solver: Now the micro
starts answering back**

**We test ProDOS - Apple's
new operating system**

**Face the ultimate games
challenge in Speedway**

HAGQ DENSHI

disk II



DRIVE 2

disk II



apple IIe

**INSIDE: 16 page guide to Apple
accessories and peripherals**



Apple '84

● All the latest on next month's star-studded Apple exhibition. **12**

Business

● Spreadsheet. Nick Levy sees who'll benefit from TK! Solver. **18**

Books

● Reviews of Visicalc for the Apple II Plus and The Elementary Apple. **65**

Feedback

● Your letters on the economics of using spreadsheets; Incredible Jack; Patience problems. **66**

Games

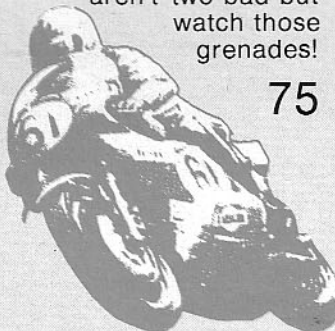
● Reviews of Wizardry III, The War of the Samurai and The Spy Strikes Back. **54**

apple user

Volume 4
Number 4
April 1984



● Speedway. Put your riding skills to the acid test with this easy-to-list game. The oilslicks aren't two bad but watch those grenades!



75

Graphics

● Peter Gorry's guided tour round the plotting instructions maze. **24**

Introduction

● John Vann tells how the Apple bug bit **53**

Macintosh

● We report the big response to the launch of Apple's latest micro. **21**

News

● Six senior Apple men disappear; robots roll out the Macintoshes; and those cooking Apples. **9**

New products

● Expert eye on the latest products on the market for Apple users. **79**

Pascal

● Our tutorial looks at WHILE loops and introduces Boolean variables. **35**

Reviews

● Peter Ashford looks at a batch of parallel printer cards to speed printout production. **70**

● Get through a page faster with Rapid Reader. **19**

● Christopher Roper compares two word processing packages. **28**

● Keith Lander concludes his review of ProDOS. **30**

<table border="1"> <tr><td>Main system ROM and RAM</td><td>Adjustment</td><td>\$3000</td></tr> <tr><td>Peripheral ROMs</td><td>System I/O</td><td>\$1000</td></tr> <tr><td>System gloves (\$5000 \$1000)</td><td>Available to system prog</td><td>\$2000</td></tr> <tr><td>System prog loaded here</td><td>Available to system prog</td><td>\$2000</td></tr> <tr><td>Text screen I</td><td>Zero page stack input buffer</td><td>\$2000</td></tr> </table>	Main system ROM and RAM	Adjustment	\$3000	Peripheral ROMs	System I/O	\$1000	System gloves (\$5000 \$1000)	Available to system prog	\$2000	System prog loaded here	Available to system prog	\$2000	Text screen I	Zero page stack input buffer	\$2000	<table border="1"> <tr><td>Language card RAM</td><td>ProDOS MLI</td><td>\$1000</td></tr> <tr><td>ProDOS MLI</td><td>Alternate \$2000 RAM card bank</td><td>\$2000</td></tr> <tr><td>ProDOS MLI</td><td>Free</td><td>\$2000</td></tr> </table>	Language card RAM	ProDOS MLI	\$1000	ProDOS MLI	Alternate \$2000 RAM card bank	\$2000	ProDOS MLI	Free	\$2000
Main system ROM and RAM	Adjustment	\$3000																							
Peripheral ROMs	System I/O	\$1000																							
System gloves (\$5000 \$1000)	Available to system prog	\$2000																							
System prog loaded here	Available to system prog	\$2000																							
Text screen I	Zero page stack input buffer	\$2000																							
Language card RAM	ProDOS MLI	\$1000																							
ProDOS MLI	Alternate \$2000 RAM card bank	\$2000																							
ProDOS MLI	Free	\$2000																							

Technique

● Save your secrets with Jonathan Lewis's encoding programs. **60**



Apple User Managing Editor **Derek Meakin** Production Editor **Peter Glover** Features Editor **David Creasey**
 Technical Editors **Cliff McKnight** **Max Parrott** Editor-in-Chief of Database Publications **Peter F. Brameld**
 Marketing Manager **Sue Casewell** Advertisement Manager **John Riding** Sales **Peter Nowell**

Telephone: 061-456 8383 (Editorial) 061-456 8500 (Advertising) 061-480 0171 (Subscriptions) Telex: 667664 SHARET G. Prestel: 614568383.

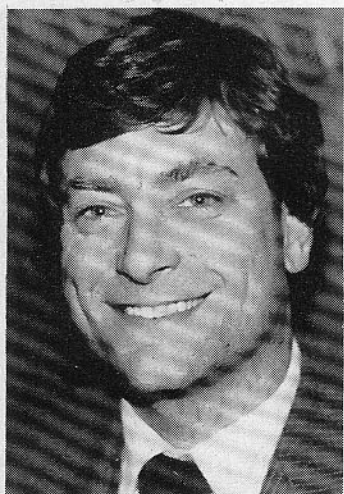
Published by Database Publications Ltd, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

Trade distribution in the UK and overseas: Contact Steve Fletcher, Circulation Manager of Database Publications at the above address or telephone him on 061-480 4153. Apple and the Apple symbol are the registered trade marks of Apple Computer Inc. Apple User is an independent publication and Apple Computer is not responsible for any of the articles in this magazine, nor for any of the opinions expressed.

Writing for Apple User: Articles and programs relating to the Apple are welcome. Articles should preferably be typed or computer-printed, using double spacing. Unsolicited manuscripts, discs etc, should be accompanied by a self addressed stamped envelope, otherwise their return cannot be guaranteed. Unless agreed, material is accepted on an all rights basis.

© 1984 Database Publications Ltd. No material may be reproduced in whole or in part without written permission. While every care is taken, the publishers cannot be held legally responsible for any errors in articles or listings.

Subscription rates for 12 issues, post free:
 £12 UK
 £13 Eire (IR £16)
 £18 Europe
 £15 USA (surface)
 £25 USA (airmail)
 £15 Rest of world (surface)
 £30 Rest of world (airmail)



Keith Hall . . . former sales and marketing director

Six senior Apple faces disappear

SIX senior employees at Apple UK have left their jobs. Apple denies that there has been a purge and says none of the staff changes are connected.

Top men now missing from the company line-up include Keith Hall (former sales and marketing director), Dennis

O'Connor (formerly product marketing manager), Ross McBeath (dealer sales manager), Dave King (vertical markets), Dave Walker (technical support) and Brian Reynolds (Lisa division).

Dennis O'Connor has joined Lotus as UK general manager, where he will be retaining close

ties with Apple through the company's software development program.

Bob Kissach, who is based at Apple's Paris office and is responsible for Lisa and Macintosh marketing in Europe, takes over as marketing director from Keith Hall.

US smuggling charges

FIVE American companies face criminal charges of smuggling counterfeit Apple computers and software into the United States from Taiwan.

Previous legal action has all been in the civil courts over copyright infringements.

Now the US government has stepped in and charged several people with importing computer parts in falsely-labelled boxes.

Apple are reported to have helped customs officers by lending them test equipment and giving technical advice.

Early in the New Year a Taiwan court jailed six computer company executives for copying Apple software.

This was the country's first-ever jail sentence imposed for computer piracy, though it may not be the last.

But there's promotion for Mike



PROMOTION can come fast at Apple Computers.

Just nine months ago Michael Spindler was appointed Apple's general manager for Europe.

Now he becomes vice president of the company's European group, which has direct operations in Britain, France, Germany, Italy, Ireland, Sweden, Austria and the Netherlands.

Apple says its European sales have increased by over a third in the last year, and Spindler's appointment reflected his part in this success.

Spindler, 41, is married with three children, and speaks three languages. He joined Apple in 1980 after working for Intel Corporation and Digital Equip-



Apple on guard

EVER since industry automated its machine tools, there's been a problem with swarf — those slivers of metal produced by drilling and cutting — but now Apple is helping out.

Swarf tangles cause all sorts of problems, including broken tools and faulty operation of workholding and tool changing systems.

Hiring machine minders defeats the object of automation, so the Machine Tool Industry Research Association are examining a system based on an Apple II linked to a Pye

video camera.

This compares the image of the tool before use with its image after. Any swarf clinging round the tool shows up as extra information and when this goes beyond prescribed limits, the Apple sounds a warning.

The same system can also detect broken tools, because this changes the shape of the image, too.

Other jobs suitable for vision systems include workpiece recognition, detection and orientation, work inspection and setting tool offset correction.

Macintosh milestone

MACINTOSH marks the fourth major milestone in the short history of microcomputing, according to Peter Cobb, managing director of Apple UK.

Naturally enough the first, he said, was the advent of the original Apple II (now superseded by the IIe).

Next came the Visicalc software package which brought "What if . . . ?" into common usage in the businessman's vocabulary and which revolutionised the way information work was performed.

"The entry of IBM — although not of their particular machine — was also important, lending to the new industry the credibility of the world's largest computer company", he said.

"And we believe that in Macintosh we have the fourth milestone. We have broken the barrier of fear and prejudice that, before Macintosh, kept 95 per cent of people away from computers in the past".

Cooking Apples..

EIGHT Apple micros at a busy Channel Islands bakery firm are being used by executives as intelligent terminals on-line to the company's business computer.

This enables department managers to dip into the firm's database for instant graphic reports on whether their sales and profit figures are within acceptable limits.

As well as handling their wages and general administr-

ation by computer, Le Brun's bakery of Jersey also relies on the system for purchasing, sales, quality control, deliveries – and for printing out 1,500 invoices daily, rising to 2,000 in the tourist season.

The company is so satisfied with the way computers have streamlined its business over the past dozen years that it has just appointed a full time data processing manager to develop further applications.



ZYNAR LOPS NETWORK COST

PRICE reductions of up to 25 per cent have been announced by Zynar for its Elf, Cluster One and Plan 4000 networking systems.

The cuts are apparently the result of lower manufacturing costs caused by higher production.

According to Rob Thorpe of Zynar, the company has grown by 150 per cent a year in the four years since its formation.

Under the new pricing the Zynar Elf 10 package, which can link up to five Apples to a 10Mbyte central hard disc, costs £7,000 (down from £8,500).

A Plan 4000 system with a 137Mbyte central disc store and linking up to 255 Apples and IBM PCs, is reduced from £23,500 to £18,500.

Hi-tech tax cuts move

APPLE has joined the Ad Hoc Electronics Tax group, an American pressure group dedicated to getting tax cuts for high technology firms.

According to the 20 hi-tech companies forming the group they pay about 10 per cent more in taxes than the US industrial average.

At the moment they are allowed concessions on their high R and D costs.

These are due to expire next year and the group wants them to be permanent.

They also want to write off the cost of short-life technical equipment against expenses, rather than depreciating it over five years, as at present.

Moreover, they want their overseas operations taxed at the lower US rate, instead of the hefty 46 per cent designed to encourage domestic industries.



Automated Macintoshes on the march

Universities sales drive

TWENTY four American universities have placed orders for Macintosh – and by the end of this year Apple anticipates at least 50,000 Macintosh computers will be in colleges throughout the US.

As part of the new Apple University Consortium (AUC) program, institutions such as Yale, Harvard, Princeton, Stanford, Dartmouth, Brown and the University of Michigan will each buy \$2 million of Apple products – mainly Macintosh computers – over a three-year period.

Apple reports that more than \$60 million in commitments have been made to date.

Both students and faculty are expected to use the computers as tools for curriculum development, and exploring new edu-

cational applications.

Under the agreement with Apple, each university is required to develop new courseware on Apple products and to share information with other consortium members.

APPLE is poised to penetrate the Latin American market with a home-grown Apple IIe.

The company has signed an agreement with the Mexican industrial giant, Grupo Manzanana, to build and market micros in Mexico.

The new company, Apple de Mexico, will be jointly owned, with Apple holding 49 per cent of the equity.

Apple sets the course

THE ubiquitous Apple is now at sea. One of the star exhibits at the Boat Show this year was an Apple-based system for yachts.

It seems that the navigator of a modern seagoing yacht is bombarded with so much information from echo sounders, anemometers, radio compasses and even satellites that he needs a computer to make sense of it all.

The Hercules 190 system from B. & G. is based on an Apple II complete with disc drives. It is mounted in a watertight box and connected to a waterproof remote keyboard on the chart table.

Armed with the Apple, the modern mariner measures his actual performance against a computerised model of ideal performance.

And he gets instant updates on his position, plus all sorts of other advice – including the best tacks to take and what allowances to make for current.

All of which may seem a bit like cheating to your average weekend dinghy sailor. However, in the world of the giant 80 foot yachts, such systems are necessary merely to keep up with the Jones's.

Robots roll out the Macintoshes

AUTOMATED guided vehicles, otherwise known as remote controlled robots, are part and parcel of daily working life at a new Californian factory.

By the end of the year it will be churning out around 132 Macintosh computers an hour.

The 160,000 square foot plant with advanced integrated computer control and auto-

mated storage and retrieval systems, was custom built at a total cost of \$20 million.

Apple claims that it is one of the most advanced manufacturing facilities in the computer industry. And it is run by just 300 people.

Its large volume production is possible in part because the Macintosh contains only one 10 x 10 inch digital board with fewer than 50 chips.

The factory combines both automated production systems and material handling and delivery systems.

It has two printed-circuit board assembly lines – a digital line, where the printed circuit boards are assembled, and an analog line, where components such as connectors and clips are added to the boards.

Eighty five per cent of the components are sequenced and automatically inserted.

A key feature of the new factory is the way material flows through the system. It is known as "just-in-time" management.

Parts are received daily, but only the number of components needed to produce enough systems to meet current demands. In this way, stocking costs are minimised.

As each component is unwrapped and placed into one of four material handling systems, its identification is

entered into the factory's control computer operation.

Large parts are carried by an overhead parts-delivery conveyor to the appropriate work station.

Small parts are handled by an automated storage and retrieval system that delivers them to the correct work stations as required.

When workers need more parts they place empty containers on the conveyer. The system automatically recognises the containers, retrieves them and sends full ones back to the work stations.

Parts are also brought to work stations by automated guided vehicles. These battery-powered robots crisscross the factory following wires embedded in the floor.

Apple is planning to introduce robotics and surface-mounted device technology into printed-circuit board, final assembly and packaging areas in the near future.

Surface-mounted device technologies provide lower profile component packages that are mounted directly to the surface of a printed circuit board, thereby increasing package density and making it possible to add more components, and system capability, or reduce product size and weight.

Sales forecast

WORLDWIDE sales of Macintosh could total 350,000 units this year, according to Californian industry analyst Jean Yates of Yates Ventures in Palo Alto.

Seventy per cent would go to businesses, 20 per cent to colleges and universities and the remainder to home users.

Aaron Goldberg, of International Data Corp, said: "There is no doubt Apple has a winner with this product. The market has been waiting for this combination of technology, ease of operation and price".

Leasing scheme

APPLE Computer (UK) has announced Apple MicroLease, a programme for business users. The scheme, set up with the Lloyds Bowmaker Finance Group, offers purchasers an alternative credit source, freeing capital for use elsewhere.

The scheme involves a "stepped payments" option. This means rentals can be cheaper at the outset, while staff are being trained, and only rise as the full value of the systems is put into effect.

More equipment can be added any time up to the last year of the agreement.

Bit of a giggle

APPLE says it is putting the fun back into computing with Macintosh, illustrating it with their light-hearted approach to a demonstration.

An Apple Macintosh man, suitably garbed in gown and mortar, runs a Macintosh school for people wanting to get their hands on a machine.

The "class" we attended had 20 Macintoshes, two per desk, and relied heavily on a cut-down version of the Guided Tour of Macintosh – the disc supplied

with each system sold – which demonstrated some of the capabilities of the MacWrite and MacPaint software (included in the price of a Mac).

In general the introduction, in conjunction with a spoken cassette tape (female, American accent) was excellent.

We also liked the graffiti Apple staff had strewn over the walls of the "classroom".

They included "Flash your Mac" "Cuddle a Mouse" and, best of all, "Orwell had one".

Record Manager Gallups to top

MOVING to the top of the charts in the record retailing trade in the UK is an Apple-based program, Record Manager.

It literally keeps a record of records – but is basically a sales and stock/ordering system which has been developed by Systems Analysis and Micro Software of London.

More significantly Gallup, which is responsible for compiling pop music charts based on weekly sales performances, has given its official approval to the program.

Gallup is particularly impressed with the protocol, developed by Sams, which allows the Apple IIe to communicate with the Gallup main computer via GPO phone lines.

And the Independent Labels Association, which recently announced the establishment of the first computerised register of record labels, is also working closely with Sams.

They hope their register will reduce potential confusion over names and prefixes in an industry which deals with more than 2,000 record labels.



EVERYONE has heard of JR, even if they don't watch Dallas, But whether you have heard of TK! will depend on who you are.

If you are a busy business executive travelling all over the country you would recognise TK as the Travel Key - a special card issued by BR which enables the holder to claim discounts on travelling and hotel expenses.

On the other hand, if you work for the firm that produces 57 varieties of foods (although the number of varieties they produce is nearer to 157) and you talk about TK, then you will be talking about tomato ketchup.

Now TK has taken a new meaning: TK! Solver stands for Tool Kit Solver - a program from Software Arts for solving mathematical problems.

Software Arts are the people who brought out Visicalc, and it looks like they have another winner on their hands.

Its appeal, compared to Visicalc, will however be to a more limited market.

The object of this review will not therefore be to tell you how good TK! Solver is, but to identify the kind of people and what sort of operations would benefit from using the program.

It is not a conventional electronic spreadsheet, so it bears no resemblance to Visicalc.

I think the best way to introduce it is to describe some of the "games" I had with it, and leave it to the reader to decide if this program could indeed provide solutions to problems that even the most sophisticated electronic spreadsheet cannot tackle.

How would you like to be able to solve the kind of mathematical problems Robert Robinson poses in his "Ask The Family" TV show?

For example, three countrymen met at a cattle market. "Look here", said Hodge to Jakes, "I'll give you six of my pigs for one of your horses, and then you'll have twice as many animals here as I've got".

"If that's your way of doing business", Durrant told Hodge, "I'll give you 14 of my sheep for a horse, and then you'll have three times as many animals as

TK! SOLVER

Powerful program that would have answered the Sphinx in a flash

me".

"Well, I'll go better than that", said Jakes to Durrant. "I'll give you four cows for a horse, and then you'll have six times as many animals as I've got here".

No doubt this was a very primitive way of bartering animals, but it is an interesting little puzzle to discover just how many animals Jakes, Hodge and Durrant had taken to market.

To solve this problem I booted TK! Solver and a screen

displayed several approximations to find the nearest answer which fits the equation.

So first I had to tell it to make the ridiculous assumption that the three farmers brought one animal each.

This was done by bringing to the screen a supporting sheet which was automatically created for each of the variables in the formula - at the same time as when they were written.

I then entered, in the

IF you like puzzles and problems, or are a businessman, or are simply curious to discover something about a new software program from the people who brought you Visicalc - read on! Here NICK LEVY examines TK! Solver, not to see whether it is any good, as it already looks a winner, but to investigate the type of people and operations the program might benefit.

like Exhibit I appeared on my monitor. I then entered the following three equations in the rule sheet (in the lower half of the screen):

**JAK+6-1=2*(HOD+6+1)
HOD+14-1=3*(DUR-14+1)
DUR+4-1=6*(JAK-4+1)**

As I was entering the above formulae, the expressions JAK, HOD and DUR appeared automatically under the column "Name" on the upper half of the screen. JAK, HOD and DUR stand, of course, for the number of animals Jake, Hodge and Durrant brought to market.

The program has two ways of tackling formulae - the direct solving and the interactive methods.

To solve the above problem TK! Solver had to resort to the interactive solving method, a procedure that starts by the person seeking a solution making wild guesses at what each variable could be.

TK! Solver then takes over and makes a series of succes-

sive approximations to find the nearest answer which fits the equation.

That was enough to give TK! Solver something to work on, and after a few seconds it came up with the answers: JAKe had 7 animals, HODge had 11 and DURrant 21 animals.

This should give you some idea of the power of TK! Solver.

As you can see, you have to present your problem in the form of a mathematical equation or formula, so the first requisite for anyone wishing to use the program is the ability to express quantitative problems as algebraic expressions.

Here is another example. How would you put the following problem:

A group of cyclists out on a trip stopped at a pub. They all agreed to have a feast together. "Put it all on one bill" they told the landlord, "we will share the cost equally".

A bill for £4 was promptly presented, but unfortunately two of the cyclists sneaked

outside and fled.

So each of the remaining cyclists had to pay 10 pence more than his share. They no doubt settled later with those rogues, but how many were there in the party when they set out?

If you can't formulate the above problem algebraically the TK! Solver program and manual will not be of much help to you. My presentation was:

$$4/X = 4/(X-2) - .1$$

where X stands for the number of cyclists in the original party.

What the formula states is £4 divided by X number of cyclists must be equal to £4 divided by (X-2) cyclists less 10 pence.

After telling TK! Solver to assume that there was only one cyclist in the party, the program used the approximation method and within a few seconds it came with the correct answer, which was of course 10.

Next I tried to solve one of the country's most burning problems which could save the nation millions of pounds every year. Foregoing offers of thousands of pounds I shall now reveal a formula which determines scientifically how long a traffic light should stay at green. The formula is:

$$Tg = .0033 * N * S * t + (0.2V)$$

where Tg=time at green, in seconds.

N=number of vehicles crossing the intersection from one direction during the peak five minute period.

S=time spacing between vehicles in seconds after they leave the intersection.

$$t=60 \text{ seconds.}$$

V=average speed in mph attained by vehicles after leaving intersection.

Entering 50, 3, 60 and 20 for N, S, t, and V respectively, TK! Solver works out that Tg should be 33.7 seconds, say 34 seconds. So far, this calculation could have been carried out on a calculator or on Visicalc, so why use TK! Solver?

Suppose you have to find out the value of N if Tg was changed to 30 seconds. Using Visicalc you would have to rewrite and rearrange the whole formula, but TK! Solver can work backwards.

Just input 30 against Tg and blank out the previous input for N and a new output value will

appear for N, showing N=43.77 (say 43).

To perform this kind of backward calculation TK! Solver used the direct solving method.

So if your problems can be expressed using algebraic notations, and if you know how to formulate problems in algebra, then it could be worth your while to invest some of your precious time to learn to use the program.

But don't be carried away. It does not handle matrix calculations – multiplying a table by a column – or integral calculus, or any formulae in the abstract where the variables cannot be quantified.

According to the promotional material, buying the TK! Solver becomes an economic proposition even if only used for 15 minutes a week.

The only reservation I would have about that statement is that unless you use the program regularly for an average of at least an hour a week, you are bound to forget how to use it properly, and you will not get the best out of it when you most need it.

TK! Solver also has facilities for plotting, creating tables and lists, and for interchanging information with other pro-

grams through DIF files – topics not covered by this review.

I used it to find the right answers to a number of mathematical puzzles I could never tackle before.

Although I had a fairly good idea how to express them mathematically, I could never solve the equations.

Now with the aid of TK! Solver, the problems are solved almost as soon as I have written in the proper formula and any allied information. However it is not a fast program if you have complex formulae and equations.

I am sure all the puzzles

discussed so far have their equivalent in the world of industrial and mechanical engineering as well as in scientific calculations.

I used puzzles to demonstrate the program's power because they are much more interesting and universally understood than using examples involving formulae for calculating volume of a buoy, or rates of evaporation.

But I was finally beaten by

the problem that follows.

The combined ages of Mary and Anne are 44 years, and Mary is twice as old as Anne was when Mary was half as old as Anne will be when Anne is three times as old as Mary was when Mary was three times as old as Anne.

How old is Mary?

Any suggestions on how I could formulate this problem so that it can be understood by my TK! Solver?

```
(1r) Rule:
For Help, type ?
===== VARIABLE SHEET =====
St Input      Name      Output      Unit      Comment
-----
===== RULE SHEET =====
S Rule
-----
```

Exhibit 1: A typical TK! Solver input screen

Product: TK! Solver
Price: £300
Publisher: Software Arts
Distributors: Marketing Micro Software

REVIEW

Slow readers should start here...

HAVE you finished yet? Only just started, eh? Maybe you need a speed reading course like Rapid Reader from Silicon Valley Systems.

Basically, the program presents words, phrases or sentences and you read them.

You can increase the speed of presentation – although at really high speeds I experienced some interference from persistence of the image on the screen – and you can test how well you're doing.

The phrases and sentences are generated from a bank of words, so that sentences like, "This fat dog drank loudly with a friend", are produced.

However they don't always make as much sense as this example.

The test takes the form of a sentence presented briefly and then re-presented with a word missing. The reader's task is to supply the missing word.

If you get them all right, you're likely to get a message like, "This is much too slow for you!"

There are five banks of words, two of which are fixed. One is in English, the other Spanish.

Of the three user-modifiable banks, two are English and one Spanish.

Each bank contains nine

groups, a sentence being built up by selecting one member of each group.

All groups have 14 members, although the group of articles – this/that/the/a – contains repetitions. Even so, a large number of combinations are possible.

Rapid Reader may help you increase the speed you read things on the Apple, since only the usual upper-case text is used. However, I have reservations about packages like this.

Firstly, I'm not sure that practice with upper-case will necessarily generalise to the more usual mixed-case printed material.

More seriously though, I

think we tend to assume that once a child can read then the only variable left to bother with is speed.

There is very little educational emphasis on *how* we read, the strategies we adopt in relation to different material, and how different purposes affect reading.

Reading is a complex skill, and speed is only one small aspect of it.

If all you want to do is increase your reading speed, this package may help you.

If you want to increase your *understanding*, you'll have to look elsewhere.

Cliff McKnight

Macintosh bandwagon is gathering speed

MACINTOSH momentum is gathering. The machine should be available in the UK towards the end of this month and already dealers who have been able to demonstrate one of the early versions produced for the American market report a fantastic public response.

Apple UK was still cagey about releasing a British price for the machine as *Apple User* went to press. But it is expected to sell for around £2,000 with the Imagewriter printer.

Some of the excitement surrounding the Macintosh launch has spilled over into the third party peripherals and software development market.

Sharing in this excitement is the Lisa 2 series. You can buy a basic 512k Lisa for just £2,625 and start running Macintosh programs on it.

Later both can be upgraded in storage and memory, and also to use the more powerful, multi-tasking Lisa software.

Apple says versions of

CP/M86 and MS-DOS, the standard operating system for the IBM-PC, will both be available for Lisa this year.

This means that you can use your Lisa in IBM or CP/M86 emulation mode should you want to do without windows, cut and paste and the other Lisa features.

But it doesn't mean that the unique aspects of the Lisa operating system are transferred to, or endowed upon MS-DOS or CP/M86 software.

Macnews flashes

ACKNOWLEDGING the existence of "Big Blue" IBM (and of its market share), Winterhalter has developed the DataTalker II for the Macintosh.

It plugs into the Macintosh serial port and, under software control, allows the Apple to access IBM and IBM-compatible terminal systems.

The emulation software, delivered on two discs, is downloaded from Macintosh into the DataTalker II — "thus relieving Macintosh of all communications overhead and overload", says Winterhalter.

The device, with software, costs \$995.

★ ★ ★

MACDISK, a complete family of peripherals for Macintosh, is due for release this month by Davong Systems.

It includes storage systems ranging from 5 to 32mbytes. Scheduled for introduction soon are a 28mbyte streaming tape back-up system and a multi-tasking local area network.

★ ★ ★

AN exciting prospect for system developers and home enthusiasts is Mac-Forth from Creative Solutions.

It integrates the interactive nature, speed and flexibility typical of Forth systems with the graphics windows, menus, mouse selection and file structure of Macintosh to provide a powerful tool for application problem solving.

★ ★ ★

APPLE chose Multiplan as the spreadsheet program for the Apple IIe — which replaced the II Plus in February last year — and now Microsoft has released a version of Multiplan as one of the first applications packages available for Macintosh.

It is one of four Microsoft releases for the machine. The others are Basic, Chart (graphics), Word and File (database management).

Games galore on their way

THERE will be games galore for Macintosh.

Already Sirotech has announced a Macintosh version of its Wizardry fantasy game, which has sold over 150,000 copies in its Apple II version and has won a variety of awards.

Wizardry designers are completely re-writing the program to take full advantage of Macintosh's capabilities.

"The Macintosh is the most incredible 'programmer's machine' I have ever seen", said Robert Woodhead, a Wizardry developer.

"The machine provides a standard human interface for programs automatically, allow-

ing the programmer to concentrate on what the program does, as opposed to how it looks to the user.

"All the other computers in my office — except my Lisa — are gathering dust".

Another major games company, Infocom, says it is looking forward to the market demand that will be created by Macintosh.

Due for release in the summer are Infocom's fantasy, science fiction, murder, mystery and adventure games including the Zork trilogy, Enchanter and Sorcerer, Planetfall, Suspended, The Witness, Deadline and Infidel.

MAC, THE FARMER'S BOY

THE Apple II has been used in the widest possible range of applications, from North Sea oil rigs and the Space Shuttle to communicating with dolphins, and Macintosh looks set to do the same.

Already 12 software packages for farmers and ranchers have been developed by AgDisk/Harris Technical Systems, one of the largest suppliers of agricultural software in the USA. Another 11

are planned for release soon.

"We made a major commitment to Macintosh the moment we saw the machine", said Bob Harris of AgDisk.

"Farmers don't particularly like the record-keeping part of their business. They aren't comfortable sitting in front of a screen full of classical computer display.

"Macintosh helps us avoid the normal boring list of numbers or letters. We can

show farmers quickly what the figures mean, rather than simply what they are".

Using Macintosh windowing and graphics AgDisk products can depict several things at the same time and link easily between them.

A program can show input in one area, have the corresponding output in another area, and in a third have high resolution graphics display of what the results mean to the farmer.



THIS month we continue the Apple User graphics package with two routines that use shape tables – those incredibly useful but infuriatingly abstruse blocks of hexadecimal code.

If you are lucky enough to have been born with 16 fingers and toes you can skip over the next section. However if shape tables are a mystery to you I'll try to shed a little light on the subject.

In addition to being able to plot points and draw lines in hi-res graphics the Apple can also rapidly draw predetermined shapes anywhere on the hi-res screen.

The shapes are conveniently stored in tabular form, and Figure 1 shows the layout of such a table.

The shapes are not stored in a graphical form, but rather as a series of plotting instructions.

Unfortunately the compact way plotting instructions are stored makes secret codes look like child's play. I shall not be discussing the way this encoding is performed here, but will come back to it another time.

Shape tables can be put anywhere in memory, and it's up to you to decide where that is going to be. This is often a problem, and there are several safe places.

Plotting instructions make a spy's cyphers look simple

PETER GORRY delves deeper into the mysteries of graphics

The first, for short tables, is down at 768 – a spot much favoured for short machine code routines too.

The second popular solution is to put the table underneath HIMEM and then to move HIMEM down to below the table to protect it.

The third place occurs if you use the technique, shown last month, to start the Applesoft program past one or other of the hi-res pages.

The memory below hi-res Page 1 now becomes available – starting at 2048 (\$800 in hexadecimal).

This is where the shape table given this month is going to sit – so remember to set the program

start past hi-res Page 1 (POKE 103,1 : POKE 104,64 : POKE 16384,0).

The table contains shapes that will be used for plotting data points on graphs. Figure II shows the symbols contained in it.

In fact two more shapes are not shown – they are used for the hi-res CURSOR routine.

To create and save the shape table carry out the instructions given in Listing 1. Each line is finished by typing Return.

*Note: There will be a star * in front of each line when you type in the table. Remember also that there are only zeros in the table – no letter Os.*

After you have issued the BSAVE command you will get a few extra digits thrown back at you – ignore them and type Ctrl-C to return to Basic.

So far so good, but we have to tell the Apple where we have put the table, and locations 232 and 233 are reserved for just this job. The Apple looks at these locations each time a DRAW or XDRAW command is encountered.

Using these locations it is possible to use more than one table. You just have to set 232 and 233 to point to the particular table you want.

As usual, the values in 232 and 233 have to be in the hexadecimal low-byte, high-byte form.

We now come to the graphics routines proper for this month. (See listing II.) They should be typed in after the ones from last month.

They don't make use of any routines from the March issue of Apple User, but you will need

the ones from the February issue.

Shape table loader

It is a universal truth that you will always need one more shape table than you anticipated, and that the first place you decide to put them will be inconvenient.

This housekeeping routine is designed to make adding shape tables, and relocating others, as simple as possible.

It controls the loading into memory of up to nine tables. Information on them is stored in two arrays, ZT() and ZTS().

The first contains the numerical data about the tables – the second the table filenames on disc:

ZT(1): Memory address of Table 1 (decimal)

ZT(9): Memory address of Table 9

ZT(10): Number of tables to

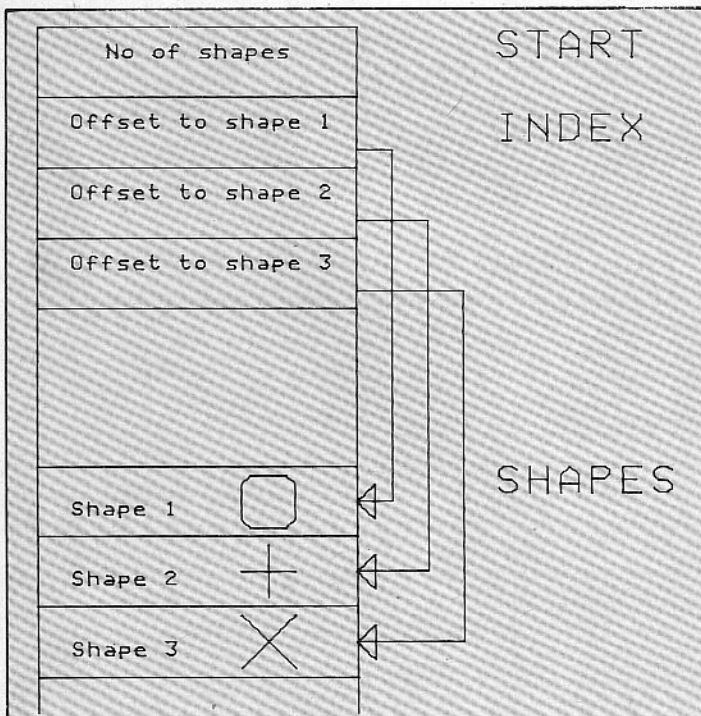


Figure 1: A typical shape table layout

```

ICALL-151
0800: 0A 00 18 00 22 00 2B 00
0808: 38 00 43 00 49 00 4E 00
0810: 55 00 5A 00 69 00 00 00
0818: 12 3F 20 64 2D 15 36 1E
0820: 07 00 24 36 2D 3F 3F 2D
0828: 36 04 00 0C 0C 94 92 1C
0830: 1C 18 1C 94 92 0C 04 00
0838: 09 1C 1C 94 92 0C 9C E3
0840: 0C 04 00 3A 24 2D 36 07
0848: 00 34 3D 2F 26 00 0C 94
0850: E2 1C 94 22 00 E1 94 E2
0858: 04 00 2D 2D 2D 2D 2D 2D
0860: 2D 2D 2D 2D 2D 2D 2D 2D
0868: 00 36 36 36 36 36 36 36
0870: 36 36 06 00
*BSAVE PLOT TABLE,A$800,L$73
    
```

Listing 1: Create and Save Table

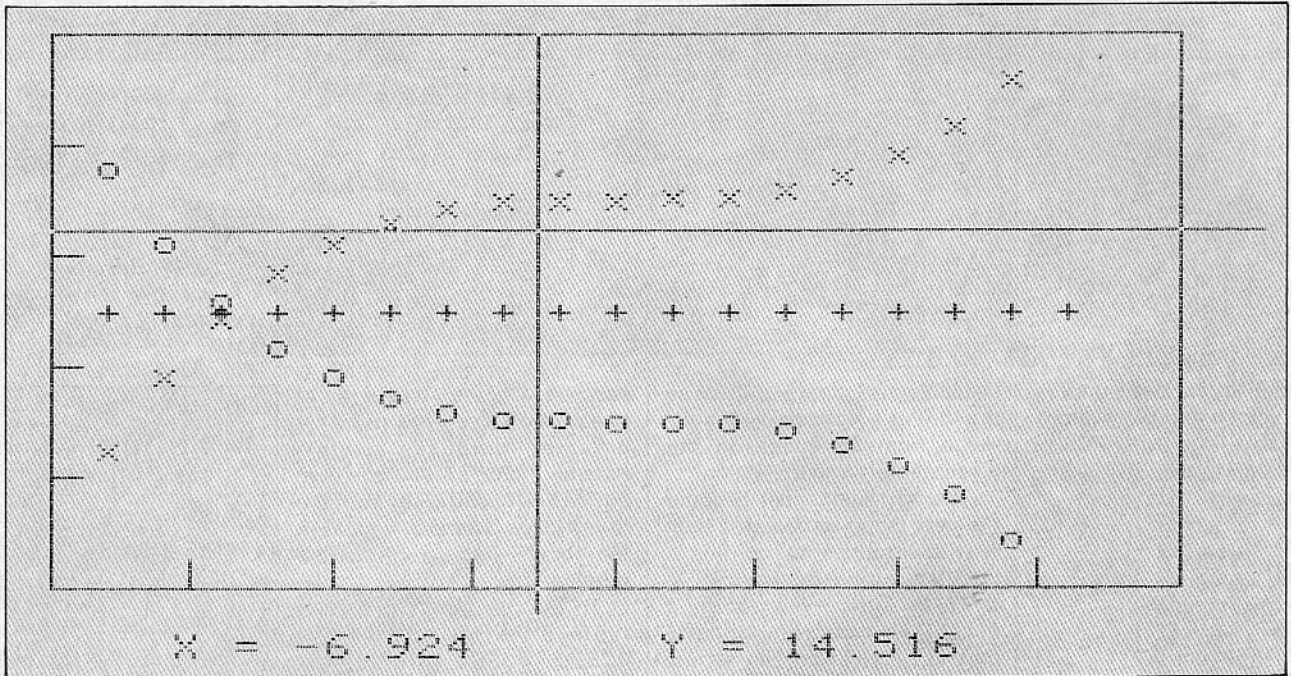


Figure III: Output from the example program

be loaded.
and

ZT\$(1): Filename of Table 1
ZT\$(9): Filename of Table 9

This routine can also be used, in conjunction with the next one, for any program using shape tables.

Shape table pointer

This sets 232 and 233 to point to the desired table. The table number is stored in ZT(0) – for example, line 42640 in which Table 1 is selected.

The address of the table is picked up from the ZT() array and the appropriate low-byte, high-byte values are calculated from this.

The strength of doing everything this way lies in the fact that if we put Table 3, say, in another place we only have to alter one value in the shape table loader routine. Everything else remains the same.

Point plotter

This plots data points held in ZX(),ZY() on the screen. Only those points within the graphics window are plotted so you don't have to worry about going off the screen.

The arrays ZX() and ZY() are going to be our standard arrays for all data plotting, and future routines will adopt this convention.

Since the default size of 10 is too small they are dimensioned in line 90 as having 100

elements.

Obviously the dimension size can be altered to suit your requirements. The number of data points to be plotted is stored in ZN (line 170).

There are four basic symbol shapes in two sizes. The shapes have been constructed so that, as far as possible, they maintain their shape at Scale values greater than 1.

More complicated symbols, or solid versions, can be constructed by plotting data points twice with different symbols.

Setting HCOLOR to black can be used to erase data points from the screen.

Cursor routine

This routine uses the games paddles or joystick to provide a pair of cross-hair cursors on the hi-res screen.

The vertical and horizontal positions are set by the paddle

values.

Cursors are widely used to read information from the screen, and many graphics terminals have built-in cursor hardware.

At first it might seem that producing a cursor on the hi-res screen is a simple matter – just a couple of HPLOTs.

The unfortunate thing is that cursors should be non-destructive. That is, they should pass over the screen leaving it unchanged.

The problem with using H PLOT becomes obvious the moment you try it.

To use H PLOT successfully one needs to draw the cursors on one page and the graph on the other.

It is then necessary to flip the pages back and forth to overlay them. This is unpleasant and very wasteful on memory.

Ideally we would like a cursor

that works on the same hi-res page as the screen image and requires only Basic to achieve it. The routine given uses XDRAW to achieve this.

The shape table contains a mini horizontal and vertical line for this purpose. The cursor inverts anything it lies over but returns the image to normal as it moves on.

This isn't quite the ideal, but the best we can hope for at this level.

The main function of a cursor is to read values directly from the screen and to display them – in user units of course.

To do this we use the inverse mapping functions given in the February article.

In order to circumvent, for the time being, the problem of displaying text on the hi-res screen the present routine uses hi-res Page 1.

It then uses the bottom four lines of the text screen, in mixed graphics/text mode, to display the X,Y coordinate values.

Although it is useful to have such information displayed on the screen, it is even more useful to have it available to the program.

For this reason the routine includes a section to store up to two X,Y values read by the cursor.

Reading is initiated by pressing a game button and a "beep"

NO	SYMBOL	NO	SYMBOL
1	o	5	□
2	+	6	+
3	x	7	*
4	◇	8	*

Figure II: Plotting symbols available in table

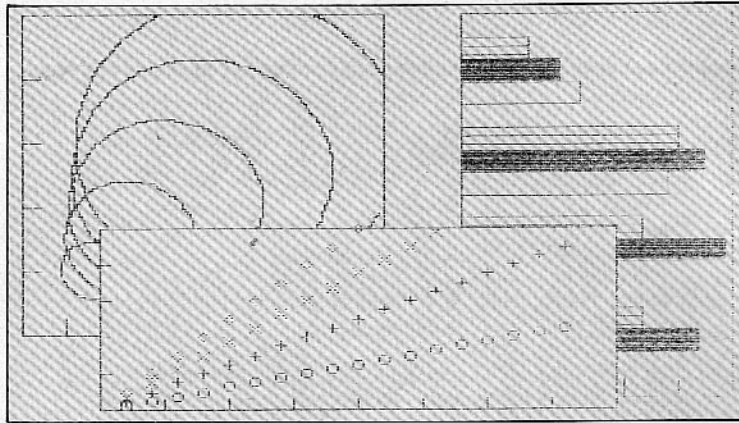


Figure IV: Results of using the routines supplied so far

signals that it has been done. Both screen and user values are available from array ZC().

- ZC(1) User X1
- ZC(2) User Y1
- ZC(3) User X2
- ZC(4) User Y2
- ZC(5) -
- ZC(6) Screen X1
- ZC(7) Screen Y1
- ZC(8) Screen X2
- ZC(9) Screen Y2
- ZC(10) 1,2 number of co-ordinates to be taken.

The reason two X,Y pairs are

available is so that we can use the cursors to select a sub-range from an image on the screen before magnification. The X,Y values just become the new mapping values. I shall provide routines to do this a little later in the series.

● *Have you ever taken a holier-than-thou attitude only to have it rebound on you almost immediately? - Well here I am eating humble pie. I stressed in the February article how important it was to*

keep track of variables and not to use dedicated ones twice for different purposes. Unfortunately I didn't follow my own advice and I used ZP as both the hi-res page variable and as a temporary flag in the circle routine.

To put this right you should replace ZP by ZQ in lines 40620, 40660, 40670 and 40680.

Suitably chastened I now have a foolproof(?) system to prevent it happening again.


```
Listing II
90 DIM ZX(100),ZY(100): REM
   DAT A POINT ARRAYS
100 REM
EXAMPLE PROGRAM
110 GOSUB 42400: REM BLOAD
   SHAPE TABLES
120 ZC = 3:ZP = 1:ZF = 1: GO
   SUB 40000: REM SET PAGE
130 ZH(1) = - 50:ZH(2) = 50
   :ZH(3)
```

Turn to Page 72 >

TWO WAYS TO ENSURE YOU GET



1. Complete and mail subscription form on Page 85
2. Hand this form to your newsagent.

Please reserve me a copy of  **apple user** magazine every month until further notice.

- I will collect
- I would like it delivered to my home.

Name _____
Address _____

Note to newsagent: Apple User should be obtainable from your local wholesaler, or contact Steve Fletcher, Circulation Manager on 061-480 4153.

You won't get a dirty Macintosh from us



But we'll expose the very latest in Apple 32 Technology and show you how you can use it in your business

CALL US NOW FOR FREE CONSULTATION AND A CHANCE TO SAMPLE OUR WIDE RANGE OF APPLE & LISA SYSTEMS

STOP PRESS

Low Price Apple II+ III systems
Our educational prices start from £499 - call for latest details

 **ozwise computers**

236 Imperial Drive, Rayners Lane, Harrow, Middlesex. Tel: 01-429 1060.

Put it in writing

CHRISTOPHER ROPER tries out two word processing packages which could prove particularly useful to students

WORD processing is now available to a new market of home users, including children who want to do their homework on an Apple.

Two of the latest programs for the Apple II and IIe are Homeword from Sierra On-Line and Pen-Pal from Howard W. Sams.

Although both come from the United States, Pen-Pal is written by two British programmers, Chris and Ann Moller.

So it is sad to report that while Homeword is everything such a system should be, Pen-Pal might have been considered an honourable first shot in 1977.

Moller is a computer consultant, and apparently assumes his customers have degrees in computer science. The manual constantly uses words like parameter and configuration, without explanation.

As one types, the text appears without wordwrap, on a matrix of dots.

The manual was confusing to this 44-year-old, who has used roughly 15 different word-processing packages, including

four on the Apple II.

The effect on a reasonably literate computer-using 13-year-old was traumatic.

Our favourite instruction from the manual was: "If you don't know the number of the slot in which your printer card is installed, consult your printer manual".

This is clear evidence of sadism on the part of the Mollers! Not only will your printer manual not provide this information, the easy method is to look under the lid of your Apple.

Two features of Pen-Pal may give a clue to its intended market. It has a nice, but not exceptional, utility for editing Basic programs, and another for sending files on-line to a bulletin board or to another Apple user.

These features could make Pen-Pal attractive to that terrifying figure of current American folklore, the computer whizz kid, who is about to start World War III by dialling into the Pentagon's command system and ordering the destruction of Moscow.

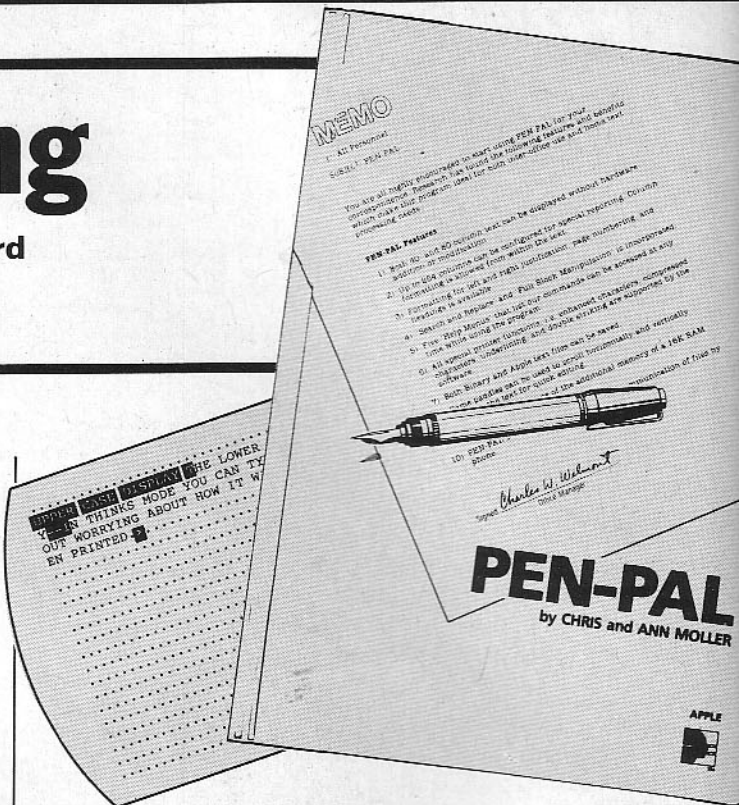
Homeword is a very different matter. It is clearly aimed at the younger user, and Sierra On-Line has excellent credentials for the job.

They produce Screenwriter II, one of the best full-feature word processing systems for the Apple II, and several members of the staff are former teachers and educationalists.

The Homeword package comes with an audio cassette, which walks the first-time user through the program.

It is accessible to someone who has never touched a computer before, and rigorously avoids computerese, to an extent which I occasionally found irritating. But then I live with a computer.

The key to using Homeword



is a series of Lisa-like icons which appear at the bottom of the screen and are selected with the right and left cursor arrows and the Return key.

There are features of Homeword which are a distinct improvement on Screenwriter II. I particularly liked the way a map of the current page is displayed in the bottom right-hand corner of the screen, showing me the layout of the page as it exists at that moment.

Shrinking lines tell how much room I have left in the memory and on the disc. Since I am the kind of person who loses files by trying to cram quarts into pint pots, I can do with all the idiot aids.

The formatting and typestyle commands are also very clear and easy to use.

It is not as powerful as Screenwriter II and professional users would quickly be frus-

trated by its limitations. But it isn't aimed at them.

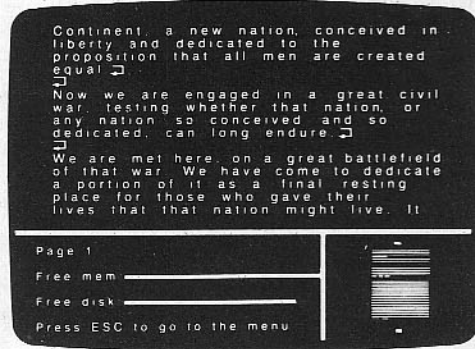
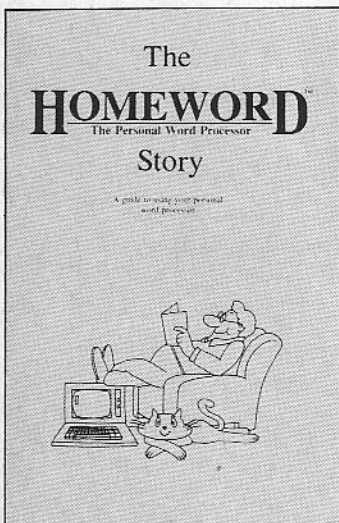
Its application in offices is limited, but I can imagine it being useful for someone who occasionally wants to type their own letters or memoranda, but does not want to learn how to use a more complex program.

Users of Applewriter II and Screenwriter II are familiar with convenient but complicated commands for altering margins and merging a form letter with a list of addresses.

The introduction to a document is the equivalent of a small computer program. You can also chain documents together when printing.

These options are not provided in Homeword, which is designed to allow you to write, edit and format letters, memoranda, notices, reports, essays, and notes simply and well.

It is not designed for



Homeword's screen is divided into three sections. Typed-in words appear in the top area while the lower left-hand corner gives information about storage space and memory remaining. The Page Sketch in the lower right shows how the page will look.

secretaries or professional writers, and it does not have features aimed at the computer hobbyist.

It is, if you like, the electronic equivalent of the portable typewriters which were standard equipment among American college students a few years ago.

One suggestion for future editions of Homeword. I think an "afterword" for more advanced users might be helpful.

For example, Ctrl X allows you to insert Ascii characters into the text. This can be an essential facility with some printers, but might confuse the novice.

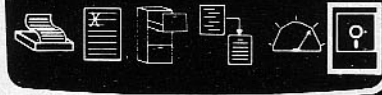
The pursuit of simplicity is admirable, but room should be found for an upgrade path. One can dispense with the icons and use Ctrl Shift commands, but this is only revealed on the reference cards.

In Screenwriter II, Sierra On-Line take care of this problem by introducing users gradually to subsets of the

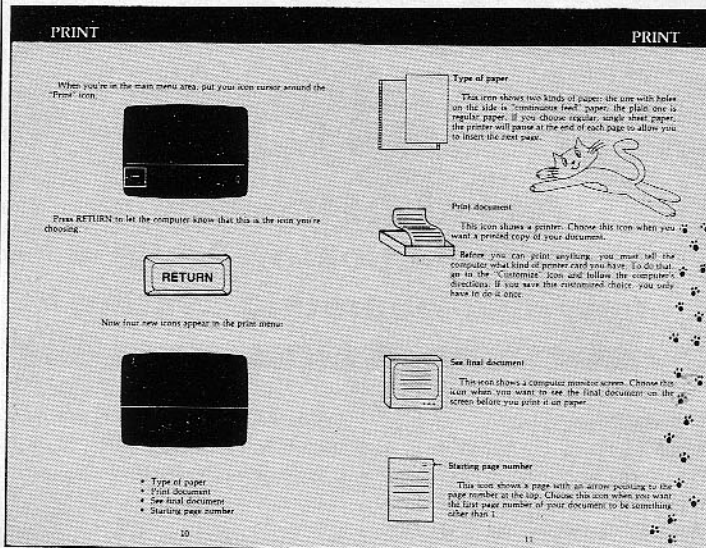
system.

Anyone who wants to produce a word processing package for British schools on the BBC or RML micros should certainly study Homeword.

It is a model of what can be done with a lot of thought in a small system.



Homeword: Some of the icons available in this package



A page from the instruction book

Product: Homeword
Type: Wordprocessor
Price: £35.95
Publisher: Sierra On-Line
Distributor: Softsel, Central Way, North Feltham Trading Estate, Feltham, Middlesex TW14 0XQ. Tel: 01-844 2040.

Product: Pen-Pal
Type: Wordprocessor
Price: £42.50
Publisher: Howard W. Sams, 4300 W. 62nd Street, Indianapolis, Indiana 46268, USA. Tel: (0101) 317-298 5566.

NDS BIG NAME DISCOUNTS

APPLE II

		R.R.P.	N.D.S. PRICE
BUSINESS			
Bank Street Writer	Disk	56.29	43.95
Home Accountant	Disk	60.32	47.10
dBase II	Disk	503.12	402.50
Wordstar	Disk	339.25	227.31
Multiplan	Disk	221.38	172.62
Sensible Speller	Disk	100.62	78.49
Filing, Cataloguing, and Mailing Friday!	Disk	88.44	62.79
EDUCATIONAL			
Type Attack	Disk	218.50	177.90
Story Machine	Disk	33.06	25.79
Math Blaster	Disk	28.12	21.98
Facemaker	Disk	40.19	31.40
Typing Tutor II	Disk	28.12	21.98
Kindercomp	Disk	20.07	15.70
ENTERTAINMENT			
Zaxxon	Disk	24.09	18.84
Lode Runner	Disk	33.06	25.78
Zork I	Disk	24.78	19.36
Sargon II	Disk	33.06	25.78
Deadline	Disk	28.92	22.57
Jumpman	Disk	41.34	32.29
Choplifter	Disk	33.12	25.86
	Disk	28.92	22.57

Plus 5% Further Discount and monthly bargains for club members only! Membership Fee £5 per year

NDS Nationwide Discount Software

To Nationwide Discount Software, P.O. Box No 28, Tunbridge Wells, Kent TN1 2YU. Tel: 0892 36832/3. Please send me the following items:

DESCRIPTION	MACHINE	PRICE
MEMBERSHIP FEE OF £5 A YEAR		
POSTAGE AND PACKING		0.80
TOTAL		

ENCLOSE CHEQUE/POSTAL ORDER MADE PAYABLE TO NATIONWIDE DISCOUNT SOFTWARE READERS ACCOUNT. REMITTANCES SHALL BE HELD ON YOUR BEHALF UNTIL GOODS ARE DESPATCHED. ALL ITEMS SUBJECT TO AVAILABILITY. DELIVERY 21/28 DAYS.

NAME _____ MEM No. _____
 ADDRESS _____
 TEL. _____

ALL PRICES INCLUSIVE OF VAT

Full catalogue available upon request. Phone 0892 36832/3.

THE new Apple II operating system, ProDOS, is a significant improvement over the old DOS 3.3.

I have reviewed ProDOS 1.0D – the software developer’s version. The end user issue – version 1.01 – should be available from Apple dealers shortly for around £35.

It will soon be shipped as standard with all new Apple IIs in place of the DOS 3.3 System Master disc.

Those who intend developing applications using Assembler will require the ProDOS Assembler toolkit which contains an upgraded version of the DOS Editor/Assembler plus a superb machine code debugger called Bugbyter. Not to mention a useful little thing called Exerciser that I’ll talk about later.

Documentation consists of two manuals, the Assembler Tools manual and the ProDOS Technical Reference manual.

The latter is indispensable for anyone intending to do serious work using the system.

I shall describe in more detail some of the things you must know and do if you want to develop a ProDOS application program using Assembler.

The software developer can view ProDOS as host for an application in one of two ways:

- He can produce a program which the user has to RUN or BRUN himself, having first

Developing ProDOS programs

KEITH LANDER concludes his examination of Apple’s new disc operating system

loaded ProDOS.

- He can build a self-booting turnkey application on top of ProDOS, thereby regarding ProDOS as a vital software tool of which the end-user is unaware.

Editors, compilers and other utilities fall into the first category, since these are simply tools that enhance ProDOS as a development base.

Accounting packages, word processors, widget sorters and similar beasts will most likely fall into the second category.

Whichever category the application falls into it will be expected to conform to a set of

system program rules defined in the reference manual. Listed briefly, these state that a system program must have:

- Code to move the program from its load position – always \$2000 – to its final execution location, if necessary.
- A version number in the system global page.
- The ability to switch to another system program.

In other words, any system program must be capable of CHAINING to another system program.

This is already a feature of

system programs such as the FILER and BASIC.SYSTEM.

When the user quits the FILER it asks him to supply the name of a system program to be entered (default BASIC.SYSTEM) and any system program can be called from BASIC.SYSTEM by typing its name prefixed by a minus sign, for example – FILER.

The reference manual contains a whole chapter on the subject of writing system programs. This describes how the chaining mechanism works and how system memory resources should be managed.

In particular it contains a detailed memory map (see Figure 1) and a detailed description of the fields in the system global page.

This latter is situated in the last page of the 48k bank and contains all the global variables.

Some of these contain useful information such as details of the machine configuration.

If you examine the system memory map you will notice that ProDOS occupies the language card RAM, which probably explains why ProDOS doesn’t support Integer Basic.

Actually, the part of the system held in the language card consists of the MLI which stands for Machine Language Interface, and contains:

Command Dispatcher

This accepts and dispatches calls from a machine language program.

It validates each call’s parameters, updates the system global page and jumps to the appropriate service routine.

The interface supports a total of 25 functions as listed in Figure 11.

Block File Manager keeps track of all mounted discs, manages the condition of all open files, and does some simple memory management. It supports the ProDOS multi-level file scheme.

Disc Driver Routines

These perform the reading and writing of data and support all disc drives manufactured by Apple for the Apple II.

Interrupt Handler

This allows up to four interrupt handling routines to be attached to ProDOS.

The handler maintains a set of addresses of interrupt routines. When an interrupt

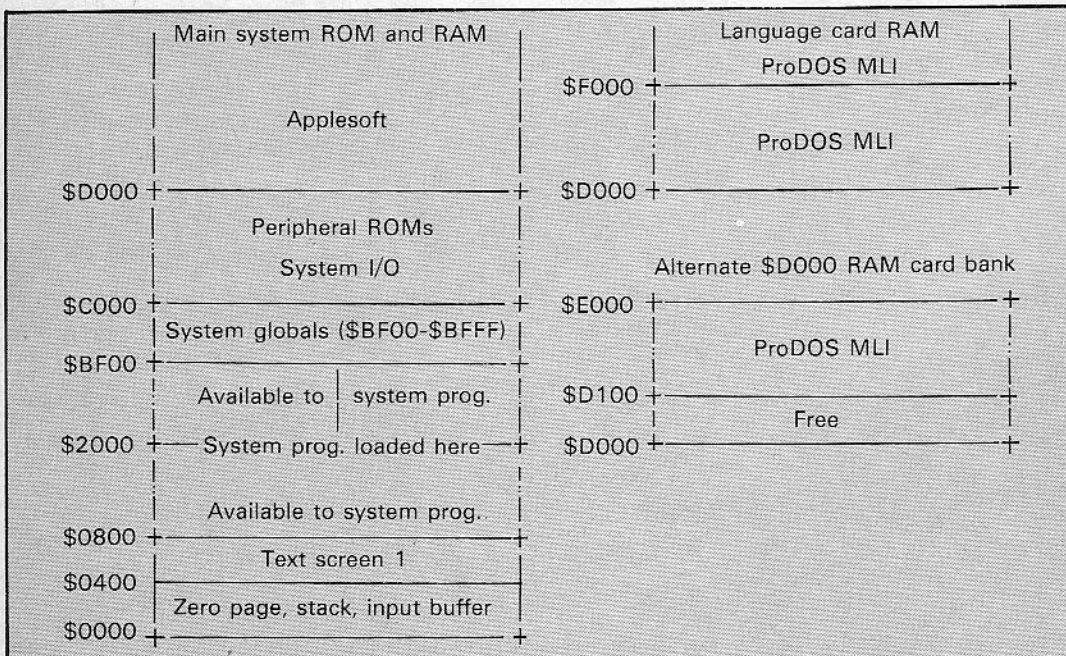


Figure 1: The memory map according to ProDOS

meeting the system for the first time.

A new, and very welcome, addition to the Assembler Toolkit is a goodie called the Bugbyter.

This is a powerful display-oriented debugging tool that can save the programmer hours of testing time.

Using Bugbyter enables me to observe precisely how my programs execute and to locate bugs in a tenth of the time I would normally have taken.

Bugbyter can be used to debug almost any assembly language program as long as there is enough room in memory for both the program and Bugbyter.

Because it is only about 7k long, this should not normally pose a problem.

Furthermore, it is relocatable, and so you can put it almost anywhere you like.

When Bugbyter is loaded and entered the user is shown the Master Display.

This is a masterpiece of design in the use of a 40 column video display screen (see Figure V). Almost everything you need to know about your program can be made to appear somewhere on this display.

You are given the choice of single stepping through your code from a given address, or allowing Bugbyter to trace automatically at a rate which you can control.

Either way the stream of instructions scrolls up the master display as they are executed with the next instruction to be executed shown in inverse video.



Figure V: The Bugbyter master display – masterful use of the 40 column display

As this happens the contents of the 6502 registers are displayed at the top of the screen.

On the left side of the screen the contents of the hardware stack are given with the top of stack indicated in inverse video.

If you wish you can select a small number of memory locations, containing crucial variables say, to be displayed and updated on the master display as you go along.

If you are in trace mode you can stop the program at any time by pressing the Escape key.

Alternatively, you can choose to plant breakpoints at selected places in your code and allow execution to continue until one of these has been reached.

The current set of breakpoints is shown on the master display.

When breakpoints are active the Bugbyter keeps track of the number of times it reaches each one and only stops execution when the trigger count is reached for one of them.

This is a value set by the user when he plants the breakpoint, and allows multiple passes of instruction sequences such as loops to be executed under breakpoint control.

At any point the user can display and alter any register or memory location he wishes, and can even request control to be passed to the monitor.

If you want to debug a program that contains real-time code, it probably won't function correctly if you try testing it using Bugbyter simulation mode.

To test such areas of code you must use what are known as real breakpoints.

These are set in the same way as ordinary breakpoints, but implemented differently.

Instead of tracing each instruction, Bugbyter plants special instructions at the breakpoints and allows the code under test to execute at full speed until the special code at a breakpoint is reached, and control is returned to Bugbyter.

Obviously there will be times when your programs want to generate screen output or request keyboard input.

Bugbyter takes care of the former by allowing you to turn off the master display, and the latter by making available a set of switches that limit the set of keys that Bugbyter is allowed to intercept.

The section of the tools manual devoted to the Bugbyter is quite comprehensive and well written, again with a fair proportion of tutorial material.

Probably anyone considering using the system will already have a fair amount of programming experience, and such people should experience little difficulty.

My own opinion, which I think is shared by other software developers who have had a chance to use ProDOS, is that it is a vast improvement on DOS 3.3.

It is a much more comprehensive and professionally produced product, and it is clear that it has been designed with the serious software developer very much in mind.

The fact that it opens up a viable upgrade path to the Apple III should have some beneficial effects when ProDOS products begin to appear.

No doubt there will be those who feel that the choice of Basic or Assembler as a development language is no choice at all.

This situation is sure to be remedied by the appearance of other languages, such as a real-time version of Forth with full ProDOS file handling capabilities.

Old software can still be used

THERE is a lot to excite Apple programmers in ProDOS. However from the average user's point of view the change from DOS 3.3 to ProDOS won't make very much difference.

Its release certainly won't cut off existing users from future ProDOS-based software releases, as some people have feared.

And, just as important, new owners who start with ProDOS will still be able to access the vast library of software programs that is

one of the hallmarks of the Apple II.

You will be able to run favourite programs such as Visicalc and Applewriter on any II, regardless of whether it is supplied with ProDOS or DOS 3.3, because most applications programs are self-booting – they don't need you to first boot the System Master disc.

Furthermore ProDOS allows you to convert DOS 3.3 text files to ProDOS text files and vice versa, so that it will be possible to incorporate an Applewriter file into

Appleworks, Apple's new three-in-one ProDOS-based program.

Product: ProDOS

Description: New operating system for Apple II range

Price: Version 1.0D under licence to software developers. Version 1.01 around £35.

Availability: Apple dealers from May.

IN the second of this series of articles we met one of the ways in which Pascal programmers can handle loops—the FOR statement. Remember how it went:

```
FOR I := 10 TO 20 DO WRITE
    LN(I);
```

The FOR statement has two serious limitations.

The first is that the controlled variable — I in the example — can only be increased, or decreased by using DOWNTO, in steps of one. There is no equivalent of the Basic STEP size.

The other limitation is that the programmer must know how many times the loop is to be executed.

It is to get over these limitations that Pascal includes two other forms of loop instruction.

The most commonly used is the REPEAT-UNTIL loop.

The body of the loop — that is, all the statements which are to be done more than once — is placed between the keywords REPEAT and UNTIL, and the finishing condition written, in the same way as an IF-statement conditional, after the UNTIL.

Here is an example, printing the squares of numbers 1 to 10 in steps of 0.1:

```
X := 1;
REPEAT
    Y := X*X;
    WRITELN(X, Y);
    X := X + 0.1;
UNTIL X > 10;
```

You can see that the value of X starts at one, and is increased by steps of 0.1 by the assignment statement.

At the end of each pass through the loop the value of X is tested, and if it is less than or equal to 10, the loop is REPEATed UNTIL the value IS greater than 10, just as the loop reads.

No BEGIN-END brackets are needed — the REPEAT and UNTIL act as statement brackets as well.

It is worth indenting the body of the loop, as I have done, to make the bracketing more obvious.

The other main form of loop is the WHILE loop. This differs from the REPEAT-UNTIL in that the test to see if the loop is completed is made at the top of the loop, rather than at the

Loop a WHILE and consider Boolean variables

bottom. The form is just like that of the FOR loop:

WHILE something DO statement;

Again it appears that the loop can be only one statement, but remember that any number of statements may be turned into one statement by using BEGIN — END. Here is an example:

```
WHILE X > 1 DO
    BEGIN
        WRITELN(X, X*X);
        X := X/2;
    END;
```

This isn't very exciting, but it prints out a table of X and X squared, halving the value of X each time, stopping when the value of X falls below 1. Of course, X must have been given a value before entering the loop.

The main difference between a WHILE loop and a REPEAT-

the "::::" is executed, the condition ("1 = 1" is tested, and is, of course, true — the loop finishes.

```
REPEAT
    ::::
UNTIL 1 > 2;
```

This will repeat "forever" — the condition 1 > 2 will never be true.

The three types of loops we have covered — FOR, WHILE and REPEAT — are not completely interchangeable. Only experience can help to decide which is the appropriate one to use in any given situation, but here are some guidelines:

- Use FOR only when you have a counter. For example, in processing all the elements of an array or string.

Counters are variables which increase steadily by one, or

the data in some way.

Most often this will be done by giving some signal data, such as a value of zero, or a negative value.

Here is a little program to convert temperatures from Fahrenheit to Celsius, stopping when a value of 0 is given.

```
PROGRAM CONVERT1;
VAR FAR, CEL: REAL;
BEGIN
    READLN(FAR);
    REPEAT
        CEL := (FAR - 32) * 5/9;
        WRITELN(FAR, CEL);
        READLN(FAR)
    UNTIL FAR = 0
END.
```

The equivalent using a WHILE loop looks like this:

```
PROGRAM CONVERT;
VAR FAR, CEL: REAL;
BEGIN
    READLN(FAR);
    WHILE FAR <> 0 DO
        BEGIN
            CEL := (FAR - 32) * 5/9;
            WRITELN(FAR, CEL);
            READLN(FAR)
        END
    END.
```

In both cases two READLNs are required. The first is to get the process started with the first temperature to convert.

The second reads the next temperature each time, and in such a way that the test can be made.

There is a minor difference between the effects of the two programs. The version with REPEAT will convert a temperature of zero if that happens to be the first value, and then who knows what might happen!

All this testing of conditions in both loops and in IF statements can be made much neater by using a new type of variable.

These are Boolean variables, named after the English logician George Boole who first systematically used letters and

By GORDON FINDLAY

UNTIL loop is that the test for completion of a WHILE loop is at the beginning, rather than the end.

A side effect of this is that a WHILE loop may perhaps never be executed, whereas a REPEAT loop must be executed at least once.

Look at these examples:

```
WHILE 1 > 2 DO
    BEGIN
        ::::
    END;
```

In this, the instructions represented by the "::::" are never executed, as 1 is never bigger than 2, and it certainly isn't at the start.

```
REPEAT
    ::::
UNTIL 1 = 1;
```

Here, after the first time that

decrease steadily, but this is much less common.

- Use WHILE loops when the test to terminate the loop can be placed at the beginning, and when BEFORE reaching the loop the program has given values to all the variables involved.

- Use REPEAT if the test is needed at the end of the loop.

You will probably find that you use REPEAT most commonly to begin with, but tend to change things around to use WHILEs as you gain experience.

Very often a loop is used to take information, one item at a time, from a disc file or from the keyboard, and to perform some processing on each item.

Writing a loop for this isn't quite as straightforward as it may appear, because of the necessity to indicate the end of

symbols in logic. Boolean variables are called logical variables in other languages.

A Boolean variable is one which can take only two values – TRUE or FALSE. Here are some examples, and their values:

```
X := 1 = 2;      FALSE
Y := 1+1=2;     TRUE
Z := HEIGHT > 6; depends on
                 the value of HEIGHT.
```

Naturally we need to declare these, just like any other variables we use. The declaration would look like this:

```
VAR X,Y,Z:BOOLEAN;
     HEIGHT:REAL;
```

Boolean variables can be used in IF, WHILE and UNTIL. Sometimes they make programs more readable:

```
REPEAT
  READLN(X);
  FINISHED := X = 0;
  Y := SQR(X);
  WRITELN(X,Y)
UNTIL FINISHED;
```

Here the Boolean FINISHED

is used only to make the program more easily changed – if at some later stage we find that the stopping condition should be $X < 1$, it is easy to see what to change.

Booleans can be used in IF statements too:

```
OLDENOUGH := AGE > 17;
IF OLDENOUGH THEN WRITELN(
'CAN VOTE') ELSE WRITELN(
'CANNOT VOTE');
```

Perhaps the most important use of Booleans is to “remember” the result of a test.

Suppose that a payroll program needed to know, at several places, whether or not overtime had been worked by a particular employee.

This information might be needed early in the program to calculate pay correctly, and later to calculate deductions such as tax and welfare payments.

Rather than repeat the test the program can save the result of the test as a Boolean OVERTIMEWRKED:

```
OVERTIMEWRKED := HOURSWRKED
> 40;
```

This test has been made, and the program can use the result of the test like this:

```
IF OVERTIMEWRKED THEN ....
```

Boolean variables cannot be added or subtracted for they do not represent numbers at all. But there are operations on Booleans, the most common being AND, OR and NOT.

AND is straightforward. The result of an AND operation is TRUE only when both the operands are TRUE. If either or both is FALSE the result is FALSE.

The word “or” has two different meanings in English. In the sentence, “I will take the bus or go by car”, the possibility of doing both, going by bus and by car, is not allowed.

But in the sentence, “I will write a games program or a utility”, it is possible that both get written, one after the other.

This second sense of the word “or” is known as the “inclusive or”.

It is this meaning that Pascal uses when the word OR is used

with Booleans. An expression with OR is TRUE if one or both of the operands is TRUE.

NOT is straightforward – it just reverses the values TRUE and FALSE.

AND, OR and NOT can be used with expressions too, but be careful! Brackets must often be used. Here is an apparently simple expression:

```
IF HEIGHT > 1.65 AND HEIGHT
< 1.75 THEN ...
```

which will be misinterpreted by the Pascal compiler as:

```
IF HEIGHT > (1.65 AND HEIGHT)
< 1.75 THEN,
```

which is totally meaningless! Using brackets can sort this out:

```
IF (HEIGHT > 1.65) AND (HEIGHT
< 1.75) THEN ...
```

Boolean variables, like so many other features of Pascal, do not seem terribly important until you actually use them.

It is surprising how much easier they make the planning and thinking about a program. Try them – you’ll like them!

And a tip from JONATHAN LEWIS on . . .

Changing boots in mid-program

IN my article on POKEing in Pascal (*Apple User* January 1984) I happened to mention that one of the tricks you could do with a judicious POKE was to fool Pascal into thinking that it booted from some volume other than the actual boot volume.

This article prompted a letter from a reader in Ireland asking how it was done, and where he could find some books which contained useful technical information on Apple Pascal.

The answer to the second part is: “I don’t know, I’ve not yet found any good books on Apple Pascal”. The answer to the first part is the program

printed here.

The first DEEK (double peek) is used to find out where in memory the SYSCOM record (see page 225 of the Operating System manual) is kept.

The following DEEKs and DOKEs change all references to volume four into the new volume which you want considered as the boot volume.

Note that Pascal also keeps note of the NAMES of volumes, so the new boot volume should have the same name.

It is most necessary that SYSTEM.PASCAL on the new volume should be in the same place as it was on the original.

```
program reboot;
uses peeklib;
const
  NEWVOL = 9;
var
  p,q:integer;
begin
  p:=deek(248);
  dake (p+4,NEWVOL);
  p:=p+96;
  for q:=0 to 31 do
    if (deek(p + 6*q) = 4) then
      dake(p + 6*q , NEWVOL);
end.
```

Catching the computer bug – how the Apple rot set in...

By JOHN VANN

ARE you old-fashioned, reluctant to change, and not "with it"? Or, more to the point, were you all of the above?

I'm a freelance writer and all the above descriptions fitted me perfectly until recently. I've been earning my living churning out pearls of wisdom for magazine articles about insurance and pensions – always using only a simple typewriter.

And for a long time I thought Apples were for eating and keeping the doctor at bay.

My re-education began with a visit to a business exhibition in Manchester. The computer section attracted me and it soon dawned how twisted my thinking had been with regard to Apples.

To be honest my non-mechanical mind finds little of interest in any kind of machinery or special equipment. I hardly know the back end of my car from the front.

Computers were somehow in a worse category. They actually put me off. I just didn't want to know.

Yet despite this some months ago I decided to concentrate on producing articles about computer insurance. It seemed pleasant enough, and far enough removed from the actual machines to trouble me.

Eventually it occurred to me

that the articles themselves would improve if I learned something, if even a smattering, about computers.

I can now bandy about phrases such as daisywheel, link disc, megabyte, business graphics and so on. The learning process continued and it was helping me to write with greater confidence about insuring computers.

However the rot had set in. Within a few weeks my ancient mind became filled with a positive urge, image or message: "Why not buy a micro?"

Sudden excitement. But ME get a micro? Where would the money come from? Could I really afford a computer. Could I really afford NOT to buy one?

I was hooked. Now it was down to asking knowledgeable friends what they thought about the idea and the points I should watch for. Was it, in fact, a feasible proposition?

To sum up, it was. Provided, that is, that I trod carefully, tried to avoid pitfalls such as choosing the wrong type of printer to go with my machine, or choosing inappropriate software, or buying from a firm which couldn't offer a sound and reliable support service should I need repairs or advice.

Still more excitement, but followed by last minute doubts. Could I really justify the outlay, even if I managed to find the money?

My courage computer-wise was definitely waning. Eventually I decided to take a train to London to ask the advice of an editor friend of mine. However by the time Euston loomed up I was more negative than ever.

"David", I said dispiritedly to my friend, "I've been toying with the idea of buying a computer".

He grinned and said: "I bought one three weeks ago. It's an Apple IIe and I like it. You should get one".

He went on to explain that he writes the occasional article

among his editing and other activities and he realised the advantages of having his own personal micro, seeing that he works from home quite a lot.

"It's going to be of tremendous help to me", he said, "especially when I get used to it – but I'm learning fast".

That clinched it for me, as this wasn't just a salesman trying to sell his wares (with all due respect to salesmen). It was somebody – and a friend – actually using an Apple and telling me to buy one.

"Seeing you're a full-time freelance writer", David went on, "an Apple IIe would open up all sorts of possibilities for you".

My doubts cleared away as I visualised my output increasing by as much as 50 per cent or even more – and with less effort, giving me more time into the bargain.

As soon as I returned home to Cheshire I phoned my local Apple dealer, Fairhurst Instruments and poured out my story, not disguising my keen interest in an Apple IIe.

The company immediately arranged for me to participate in a two-hour familiarisation session with Keyboard Training in Manchester.

On the course I was carefully and painstakingly guided through introductory programs and what-not. Being able to type helped considerably and I managed to understand most of what went on.

Suddenly computers in general, and Apples in particular, were *not* so strange and intimidating.

Going home rejoicing, I telephoned the dealer and arranged the purchase for the next day.

Now that I have set up my system at home I am all ready for action with all guns firing.

I'm having fun with the mechanics of writing using a word processing program, and my productivity and efficiency has improved.

I reckon to be able to boost my income significantly in 1984 and aim to have paid for my Apple IIe out of my extra income within the next six months.

Can't be bad, can it?



My ancient mind said 'why not buy a micro?'



There's a-code for it

CLIFF McKNIGHT shows how adventure games can be compressed into less memory space by writing them in a special code

I'D like to capitalise this month on a development brought about by the proliferation of home micros.

As you may know, many rely on a cassette recorder for data storage and retrieval. With arcade games, this isn't too bad because they'll fit in memory. Loading is slow, but once it is done there's no problem.

With adventure games it's a different story. The size of most adventures and the frequent need to read new information into the program means that hi-res adventures, like we're used to on the Apple, are out of the question.

In order to maximise on the amount that could be held in memory, and therefore produce a reasonable adventure game, Mike Austin of Level 9 Computing developed what he calls "a-code". The "a" stands for adventure.

Used in conjunction with

some of Level 9's compression programs, a-code apparently makes for a much better game — on a Sinclair Spectrum.

Now the Spectrum uses the Z80 chip and the Apple uses the 6502. The two have different architectures and instruction sets, so a-code isn't transportable across the two machines.

However the principles used by Mike Austin to develop a-code can be applied to almost any architecture, with the possible exception of St. Pauls. Also, there is an awful lot of text already held in various ROMs which can be drawn on.

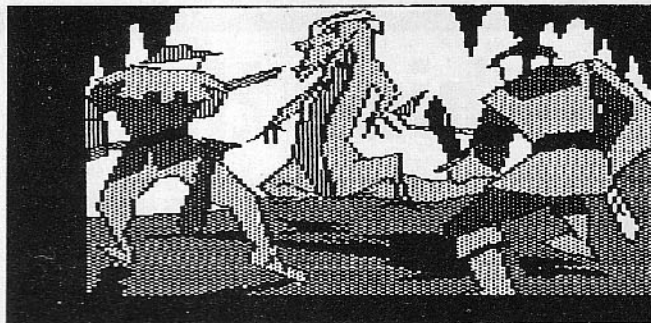
In order to demonstrate this, I've written the following small adventure game.

The advantage of such code should be immediately obvious — the text is coded too, so typing in the program doesn't spoil the enjoyment of the game.

Have fun — and there's a prize for the first person to locate the dragon's cave.

```

O3  DRAGON'S CAVE
5 HOME: INPUT "WHICH
  DIRECTION?";D$
10 CALL -936
20 FOR I = 1 TO 19: READ Z:
  POKE 1201 + I,Z: NEXT
30 FOR I = 1 TO 11: READ X:
  POKE 1460 + I,X: NEXT
90 DATA 66,65,78,71,97,160,
  217,207,213,160,193,210,
  197,160,196,197,193,196,
  161
95 DATA 1,16,18,9,12,160,6,
  15,15,12,33
  
```



Fighting L'kbreth in the hi-res introduction

If ever a wiz there was then this is it

ARE you a Wizardry adventurer? If not, it's time for you to look through past issues for the reviews of Wizardry I and II. Otherwise, here's the latest Wizardry news.

Well, there I was in Gigamesh's Tavern sipping real ale when a thief came up and asked: "How's about a pint?"

What could I say? I bought him a drink and we both got talking.

It turned out that he'd been waiting for Boltac's parcel post to deliver a new adventure for quite a while now.

It was true though, it had been months since the Staff of Gnilda had been recovered, and my sword was gathering rust.

So I asked him if he needed a fighter to come along on the adventure. Before you could say "Bleeb calls for help" we had made a deal. Who could have guessed that he could pay me in Creeping Coins? Never trust a thief!

Seven months and dozens of enquiring phone calls later the thief's adventure arrived — Wizardry III, "The Legacy of Lylgamyn".

Although there are no new spells, and the castle offers no new facilities, Wizardry has taken on quite a different look.

All the text is in chunky white hi-res, and a system called "Windo-Wizardry" has been introduced.

This display system involves framed "windows" of text which may be laid over each other and peeled off again — something like Lisa's display.

Thanks to Windo-Wizardry the maze now occupies the entire screen. Informational

windows of text may be toggled on and off as desired.

The new format is very nice and as Sir-Tech say, helps the player become more engrossed in the maze.

The other obvious changes are listed within the scenario package:

- You can make an additional scenario disc with a single disc drive.

- You must play on an additional scenario disc.

- The *ROSTER command has been shortened to *.

- Pressing * in Gigamesh's Tavern will list names eligible to join the party.

- A new (P)ool gold option transfers all party gold to one character.

- The Quickplot option now remembers its status over the entire duration of the playing session.

- The combat option (B)ack has been changed to (T)ake back.

- "Malor" has taken on a new format.

- The option (R)ite of passage has been added to the Training Grounds inspect options menu.

The (R)ite of passage option is selected to conduct the new "Rite of Passage" ceremony. The ceremony has been introduced so that your adventurers may be rejuvenated.

It involves transferring the characters to the scenario disc where "their spirits are joined with apprentice adventurers from the training grounds".

A new adventurer so created is level one and inherits the name, titles and depleted characteristics of an original character. Only newly-created

characters may be used.

After the ceremony it's off to Boltac's new look trading post. I say new look because all the items are new. Weapons like broadswords and battle axes are available, and you need a dictionary to find out which items are the helmets.

Down in the six level maze the monsters and their pictures have been changed. The graphics are superb, and are now placed in the centre of the screen.

Pictures are displayed not only for monsters, but for some rooms that the first Wizardry manual termed "specials". As

claim a mystical orb from a great dragon. Much to my delight, the storyline is displayed in clever hi-res pictures when the disc is booted.

The challenge is the hardest yet, and reaching the orb should take at least a month of hard adventuring. Most people should take longer, especially with the latest emphasis on riddles.

There are at least five riddles, four of which may not be bypassed by the "Malor" spell.

There seem to be no errors in this scenario, and the "Malikto" spell has been corrected. Identifying in the castle is still not

If it were, the authors would stand to rake in £10,000 without even selling a single copy. For this is the prize the British Go Association hopes to offer for any program capable of beating the British champion.

Clearly a Go-playing program of any sophistication would be welcomed with loud acclaim in some quarters. Equally clearly, no one familiar with the game imagines it would be a walkover to write such a thing.

Sad to relate, "The War of the Samurai" is not only not a version of Go but it does not even play against the user.

It is strategically complex only in the sense that players may find that between two and four of them they can develop complex strategies.

All the program does is plot the players' moves, occasionally perform a simple calculation, and do some nifty footwork with its RAND function.

All this could be accomplished with a board, counters and a pair of dice.

The game is played on a 16 x 18 grid by placing pieces on the intersections and by moving some from one intersection to another.

This is where the computer is at its greatest disadvantage compared with a board and counters.

To make your moves, you have to key in pairs of co-ordinates, but to do this you have to squint along the lines to read the numbers and letters at the side of the screen.

Though the program is generous enough not to penalise you for getting coordinates wrong, it is tiresome having to re-enter them whenever you make an error. Definitely not a game for astigmatics.

It would be nice to report that the program comes into its own when it performs its calculations, but sadly this is not so.

Calculations are required whenever opposing pieces are placed on adjacent intersections, for this is the setting for a "combat".

The outcome of such a situation, whether the attacker or defender loses a piece, is decided by the result of the calculation plus chance.

Admittedly it would be taxing to do the calculations in one's head and the computer saves a

bit of effort here.

However to play with real strategy, you need to perform the calculations yourself before you make your moves or else you will blunder into situations you haven't evaluated.

The War of the Samurai may be a good game in itself. It is likely to be most interesting when there are three or four players, for then there are opportunities for alliance building, negotiation, and double cross.

But it is the sort of game you have to play many times before deciding if it is strategically interesting or not.

If you are addicted to games of this sort my advice is to get hold of the rules and try it out on a Go-board.

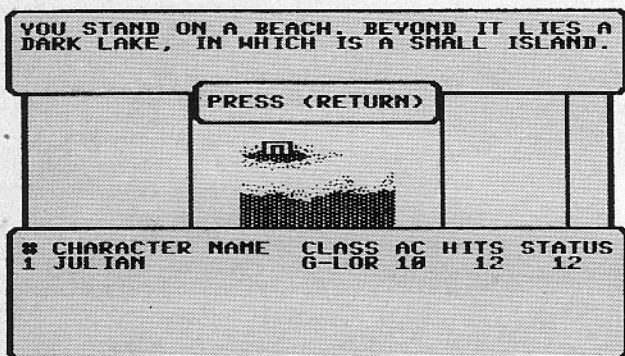
The calculations need only be done once and then copied onto a handy reference card. The operations of the random number generator could easily be approximated with dice.

Why pay the equivalent of \$39.95 to save yourself so little trouble and then ruin your eyesight?

Then, if you don't like it, you can return to the Japanese master game with no regrets. You will have saved all that money towards that beautiful Kaya wood Go-board you've always wanted.

Mike Startup

Title: The War of the Samurai.
Author: None credited.
Publisher: Krell Software.
Requirements: Apple II.



An example of Windo-Wizardry

well as this, complexes have been included in the maze — such as a fort with a moat.

These areas become apparent when mapping out the maze, and along with the "specials" help create a comprehensive world.

The power of the monsters is disappointing, this being a scenario for players with some experience. Most monsters in the first three levels will succumb to a "Katino" spell, and few in the whole game cast spells higher than level three.

On the good side though, many monsters are straight from "Dungeons & Dragons" and most behave in a traditional way.

Surprisingly, encounters have become rarer than in past scenarios. The treasures found afterwards are more fantastic than ever and challenge even the best player to guess their nature. I'd love to know what a "necromancy rod" and countless other items do.

The scenario theme is strong. Basically, your characters must work their way up a volcano to

possible though, an addition that I thought might have been made.

From what I hear, the future Wizardry scenarios are going to be made much more complex.

May I suggest that jumping straight from Wizardry I to Wizardry III is a good idea? Many people, myself included, felt that "The Knight of Diamonds" was disappointing.

Julian Brewer

Title: Wizardry III.
Authors: Andrew Greenberg and Robert Woodhead.
Publisher: Sir-Tech.
Requirements: Apple II, II+, IIe, and III.

No-GO situation

THE blurb that comes with this game promises a "strategically complex version of GO" — the ancient oriental strategy board game. But alas, it is no such thing.

Spy is in with a chance

FAR from being dead, that lovable little spy from Spy's Demise is back to give you another chance to win a prize from Penguin.

In The Spy Strikes Back you stand to win \$100 of software if you crack the code. However, you've got to find the clues first.

They are hidden away in Dr Xavier Tortion's (geddit?) German fortress. The screen displays a "section" of 16

rooms, with 24 sections on each of the five floors.

Hence there are 120 sections in which the nine clues can be hidden.

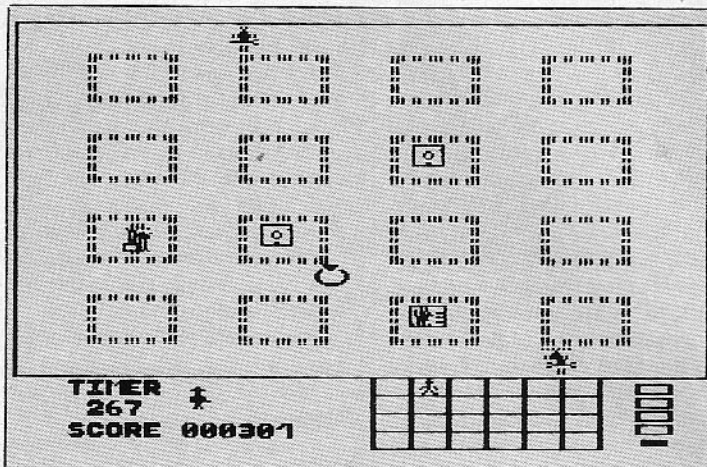
Of course it's not simply a question of exploring. You must avoid the guards, too.

When you first enter a section there is no visible door, although there may be an elevator leading to another floor. In order for the door to appear, you must get a flashing ring from one of the corridors.

If there is a spy shape instead of the ring you've found one of the clues. Also some of the rooms contain bonus objects which bestow additional points.

In addition to the main section display of 16 rooms, the position of the section is shown on a 6 x 4 floor map and a level "map" shows you which floor you are on.

A counter bottom left of the screen shows how many points



The spy strikes back

you'll score for getting through the section. However if a guard sees you an alarm sounds and your points are halved.

Dodging into a room will stop the alarm, but if you are seen again your points will be halved again. As in Spy's Demise, the

spy is without weapons, so must resort to stealth.

Control is by keyboard or joystick, with the usual convenience controls to pause, toggle sound effects off, restart the game and view the high scores, which are saved to disc.

If you've a Mockingboard speech card you can route the sounds through it, but the program defaults to the Apple speaker.

Once again Penguin has produced a game with arcade-like action and the added interest of the code.

If you are to find all the clues you will need to map the sections and elevators, which means you have to make extensive use of the pause facility so that your points don't disappear while you draw.

The code has to be seen and heard to be believed. I haven't a clue what it means. Mind you, I never cracked the code in Spy's Demise, either.

Cliff McKnight

*Title: The Spy Strikes Back.
Authors: Robert Hardy and Mark Pelczarski.
Publisher: Penguin Software.
Requirements: Apple II.*

MACINTOSH

TVS have already established themselves as a leading UK Dealer and Software Development Centre for Lisa.

This expertise has now been brought to bear on the remarkable new low cost Macintosh.

Phone or write for details of availability and price.

We will be pleased to include you on our mailing list to receive full details of all available software and hardware as it becomes available.

Access and Visa card orders accepted by phone



THAMES VALLEY SYSTEMS
GREYS HOUSE,
7 GREYFRIARS ROAD,
READING. RG1 1NU.
Tel: (0734) 581829

Best Buy Bargain Bundles

End of Season Sale of Complete Apple Systems
All genuine Apple Branded products Full pre-sale training and post-sale support available

APPLE IIe 64K BUNDLES	Vega Price	Apple List	You Save
Professional Home Pack Apple IIe 64K, Disk Drive & Controller, TV Modulator and assorted goodies	£749	£1220	£471
Business Starter + Apple Printer Apple IIe 64K, Disk Drive & Controller, Apple Monitor, Apple Printer & Super Serial Card	£1249	£1870	£621
Full Business System As above, plus second Disk Drive and 80-column card	£1499	£2195	£696

APPLE III 256K BUNDLES	Vega Price	Apple List	You Save
Basic System Apple III with high-res Monitor, in-built Disk Drive and SOS Operating System	£1975	£2395	£420
Apple III + Word Processing System As Basic System plus Apple Daisy Printer and Word Processing software	£2795	£3898	£1103
Apple III Profile System As Basic System plus 5MB Profile Hard Disk	£2795	£4190	£1395

Prepayment only and subject to availability.
Special Terms and Prices for Educational, Government, Public Sectors and PLC's. Call us for IBM PC prices.

10 Essex House
George Street
Croydon CRO 1PH
Tel: 01-656 8431, 01-680 4484
Telex 943763

Vega

Computers Limited

Save your secrets with a code that can't be cracked

IF you've ever fancied aggravating the people who "borrow" your discs and play with your programs why don't you make all your programs unusable? (To everyone else but you that is.) All you have to do is encode everything on your disc with a code that can't be cracked.

The machine code program here will ask you to specify a portion of the Apple's memory or pick out the current Basic program for you, and then request an encoding key with which, in about seven tenths of a second, will convert your data into a fairly random-looking list of garbage ready for you to save to disc.

When you reload this garbage and pass it through the encoder again - using the same key - your original data will reappear just as quickly.

The reason why the encoding and decoding use the same key is that the program "Exclusive ORs" each byte of your data with a byte from the key (see Figure 1). EORing twice takes you back to what you started with.

Since the text is rather heavily commented it needs very little explanation, but it is worth mentioning the extra problems caused by requiring the encoder to handle Applesoft programs so that they can be saved to disc, reloaded and still decode properly.

If you are a regular reader of *Apple User*

you will probably know how a line of an Applesoft program is actually stored in the machine:

2 bytes which make a link to the next line

2 bytes giving the current line number

Up to about 250 bytes of actual line

A byte containing 0 to mark the end of line (EOL).

When the Apple loads a Basic program from disc it uses the EOL zeros to calculate the values of all the two-byte links, so these zeros must NOT be changed by the encoder. Furthermore, Applesoft will be misled if the encoder produces any extra zeros.

Given that an encoded program has all its links recalculated and inserted on loading, consider what happens when you try decoding. All the links are encoded, so Applesoft will lose track of all the lines after the first one.

The solution to these problems is quite simple - just don't bother to encode the zeros or the link bytes which follow them. Similarly, if a byte is going to code to a zero, don't encode it.

Finally, since the two-byte representation of a line-number could contain a zero (which the encoder would misinterpret as an EOL) it is necessary to skip over these as well. Although these gaps in the encoding make it theoretically possible for someone

```

if      A   =  1.1.0.0.0.1.1.0
and     B   =  1.0.1.0.1.0.1.0
-----
A EOR B =  0.1.1.0.1.1.0.0

A bit in the result is set to 1 if either but NOT both of the
corresponding bits in A or B is a 1.

EORing the result with B again gives the following:

RESULT =  0.1.1.0.1.1.0.0
B       =  1.0.1.0.1.0.1.0
-----
RESULT EOR B =  1.1.0.0.0.1.1.0

which is the original byte A.

```

Figure 1

*	E3	E1	F2	F2
*6000.61D3				
6000-	20	39	FB	A9
6008-	BE	61	20	8A
6010-	20	63	6F	64
6018-	65	63	6F	64
6020-	8D	00	20	8D
6028-	65	20	6D	65
6030-	20	73	74	61
6038-	64	64	72	65
6040-	6F	72	20	3A
6048-	20	74	6F	20
6050-	64	65	20	79
6058-	42	41	53	49
6060-	6F	67	72	61
6068-	51	27	20	74
6070-	61	76	65	20
6078-	65	6E	63	6F
6080-	00	20	67	FD
6088-	AD	00	02	C9
6090-	6C	61	4C	D7
6098-	03	41	38	61
60A0-	A7	FF	A5	3E
60A8-	85	01	20	8D
60B0-	76	65	20	6D
60B8-	65	20	65	6E
60C0-	64	72	65	73
60C8-	FD	20	C7	FF
60D0-	3E	85	02	A5
60D8-	AE	61	B0	26
60E0-	87	53	74	61
60E8-	75	73	74	20
60F0-	65	66	6F	72
60F8-	6E	69	73	68
6100-	22	60	20	8A
6108-	65	72	20	79
6110-	63	6F	64	65
6118-	20	8A	61	45
6120-	69	6E	67	87
6128-	61	20	8A	61
6130-	73	68	65	64
6138-	20	C9	61	60
6140-	B1	00	D0	0B
6148-	05	85	00	B0
6150-	00	02	F0	02
6158-	D0	02	E6	01
6160-	0A	E8	BD	00
6168-	D7	F0	D3	60
6170-	A5	67	85	00
6178-	C6	00	A5	6A
6180-	38	E9	04	85
6188-	03	60	20	58
6190-	68	85	05	A0
6198-	80	20	ED	FD
61A0-	F6	98	18	65
61A8-	69	00	48	98
61B0-	02	E5	00	A5
61B8-	4C	3C	D4	4C
61C0-	B5	00	9D	B8
61C8-	60	A2	05	BD
61D0-	CA	10	FB	60

Figure 1

to find out what your original key was, the effort required would be pretty horrible.

The program is located at \$6000 and can be called from Basic by CALL 24576. However, if you want to see the encoder in action, why not switch to the hi-res screen, fill it with some junk, then call the encoder from \$6003 (CALL 24579) to "encode" the hi-res screen?

If you can type in the correct answers

without seeing the program prompts, you will be able to watch the EORing happening all over the HGR screen.

For those who don't have an assembler, *Figure 11* is a hex dump of the program, which can be copied in directly from the monitor. If you don't have lower case on your Apple you will need to change some of the data to convert it for upper case output.

The groups of bytes enclosed in brackets are text, and any of these bytes starting with a 6 or 7 should be altered to start with a 4 or 5.

There are two bugs. Sometimes after encoding the message ?SYNTAX ERROR appears, but this doesn't seem to have any bad effects. More seriously, if you try encoding while the printer is on line, the Apple can crash.

```

; Exclusive Or encoder/decoder
;
; J.P.Lewis
;
.OPT NOS,NOG,NOL
;
; Monitor routines used.
SETTXT  =#FB39
HOME    =#FC58
GETLNZ  =#FD67 ;Get input line to $200.
COUT    =#FDED
ZMODE   =#FFC7 ;Clears Monitor status flag, and Y reg.
GETNUM  =#FFA7 ;Tries to get a two-byte HEX number from
;the input buffer ($200), and store it in
;this zero-page location.
VALUE   =#3E
INPUT   =#200
CODE    =#200 ;Input buffer.
PROMPT  =#BE ;'>' is the encoders prompt.
PRMTLC  =#33 ;Contains the current prompt
STAR    =#AA ;ASCII '*'
Q       =#D1 ;ASCII 'Q'
PRGST   =#67 ;Pointer to Applesoft start of program.
PRGEND  =#69 ;Pointer to Applesoft start of data.
;
; Storage addresses used by the encoder.
START   =#0
END     =#2
POINTR  =#4 ;Used in printing.
*       =#6000
;
; Initialisation.
JSR SETTXT
LDA IPROMPT ;Set up the encoders prompt char.
STA PRMTLC
JSR SAVEZP ;Save the contents of $0-$5
;
; Get the start address.
JSR PRINTH
.BYT 'XOR coder/decoder ',#8D,#8D,0
RANGE JSR PRINT
.BYT 'Give me the start address, or !',#8D
.BYT '*!*' to encode your BASIC program.',#8D
.BYT '*Q' to leave the encoder.',0
JSR GETLNZ ;NB Line-length is in X-reg.
CPX #1 ;Is it a one-character input?
BNE NUMBER ;No, so assume it is a number.
LDA INPUT ;Otherwise see if it a '*'.
CMP #STAR
BNE ISQUIT ;If not, do next check.
JSR BASCOD ;Else point to the BASIC program.
JMP CHECK
;
ISQUIT CMP #Q ;Is it a 'Q' ?
BNE NUMBER ;If not, assume it is a number,
JMP EXIT ;otherwise quit.
;
NUMBER JSR ZMODE
JSR GETNUM
LDA VALUE
STA START
LDA VALUE+1
STA START+1
;
; Get the ending address
JSR PRINT
.BYT #8D,'Give me the end address',0
JSR GETLNZ
JSR ZMODE
JSR GETNUM
LDA VALUE
STA END
LDA VALUE+1

```

```

STA END+1
;
; Check that the end is after the start.
CHECK JSR SUBTRT ;Compare (START) and (END).
BCS GETKEY ;(Start) < (End)
JSR PRINTH
.BYT #87,#87
.BYT 'Start must be before finish'
.BYT #8D,#8D,0
JMP RANGE
;
; Get the encryption key, into the normal input buffer.
GETKEY JSR PRINTH
.BYT 'Enter your code',0
JSR GETLNZ
;
; Do the job.
JSR PRINTH
.BYT 'Encoding',#87,#87,0
JSR ENCODE
JSR PRINTH
.BYT 'Finished',#87,#87,#8D,0
EXIT JSR RESTZP ;Restore the zero-page data.
RTS
;
; Encoding routine. Using simple XOR encryption.
;The normal input buffer contains the key, which may be
;anything that can be typed as a standard input line.
; To code or decode, keep a pointer moving along the
;memory range to be encoded, and a second pointer rotating
;around the key (NB The end of key is marked by the CR,
;i.e. #8D which is the end-of-line marker from GETLNZ.)
; Because the coding is by XOR, the key to decode will
;be the same as the original encoding key.
;
ENCODE LDY #0
LPCODE LDX #0
LPCHAR LDA (START),Y
BNE NOTEDL
LDA START ;On hitting a zero, skip it, and the
CLC ;next four bytes, which should be the
ADC #5 ;pointers to the next BASIC line,
STA START ;and the current line-number.
BCS PAGEXV ;NB This increment of 5 in the text
BCC SKIP1 ;increments the key by 1 only.
NOTEDL EOR CODE,X
BEQ DONT0 ;We don't want to produce any zeroes.
DONT0 STA (START),Y
INC START ;Move memory pointer along.
BNE SKIP1
PAGEXV INC START+1
SKIP1 JSR SUBTRT ;Check if (START) has reached (END)
BCC DONE ;If so, it's all over.
INX ;if not, move code pointer on
LDA CODE,X ;and go round again if this is
CMP #8D ;the end of the code-key.
BNE LPCHAR
BEQ LPCODE
DONE RTS
;
; Encoding the current Applesoft program: Use the normal
;Applesoft pointers to find the start and end, then set up
;the encoder's pointers to the zero marking its start, and
;the zero marking the last EDL.
; As Applesoft loads a program, it sets up all its
;'Next Line' pointers by looking for the EDL zeroes, so
;it will be necessary to :-
;1) Avoid encoding these zeroes.
;2) Avoid encoding anything to produce a zero.
;3) Avoid encoding Applesoft line numbers, as they may
; contain zeros which would be mistaken for EDLs.
;4) Avoid decoding the pointers that Applesoft sets up.

```

ENCODING

```

BASCOD LDA PROGST+1
        STA START+1
        LDA PROGST
        STA START      ;Point at the zero just before
                        ;the program. The funny method is
                        ;because the DEC instruction does
                        ;not affect the carry flag.
NOTXVR  DEC START+1
        DEC START
        LDA PRGEND+1
        STA END+1
        LDA PRGEND     ;Change the pointer from the
                        ;Start-of-data to the
                        ;End-of-program. i.e. back 4 bytes.
        SEC
        SBC #4
        STA END
        BCS LOADED
        DEC END+1
LOADED  RTS
    
```

; The output routine to print 'in-line' text.
 ;The text to be printed follows immediately after a JSR
 ;to this routine, so the top-of-stack contains the address
 ;of the last byte before the text. Pulling the t.o.s thus
 ;provides an indirect pointer to the text. To mark the end
 ;of text, a zero byte is used; at which point, it is
 ;necessary to add the length of the text string (found
 ;in the Y reg.) to the pointer, push this back on to the
 ;stack, then return.

```

PRINTH JSR HOME
PRINT  PLA
        STA POINTR
        PLA
        STA POINTR+1
        LDY #1
        LDA (POINTR),Y
PRNCHR ORA #80      ;COUL expects the top bits set,
        JSR COUL    ;but ASM/65 leaves them clear.
        INY
        LDA (POINTR),Y
        BNE PRNCHR ;End of string yet ?
        TYA
    
```

```

CLC
ADC POINTR
TAY
LDA POINTR+1
ADC #0      ;This handles crossing a page.
PHA
TYA
PHA
RTS
    
```

; Simple 16-bit subtraction routine. On exit, the carry
 ;flag is clear if (START) is greater than (END).

```

SUBTRT SEC
        LDA END
        SBC START
        LDA END+1
        SBC START+1
        RTS
    
```

; Routines to save and restore the contents of \$0-\$5
 ;which are used for storing indirect pointers.

```

STORE  .BYT 0,0,0,0,0,0      ;In-line storage space.
    
```

```

SAVEZP LDX #5
NEXTZP  LDA $0,X
        STA STORE,X
        DEX
        BPL NEXTZP
        RTS
    
```

```

RESTZP LDX #5
RESZP1 LDA STORE,X
        STA $0,X
        DEX
        BPL RESZP1
        RTS
    
```

.END

NEW FROM CIRTECH CACHECARD16 AND CACHECARD64

THE ULTIMATE IN PRINTER CARDS AT UNBELIEVABLE PRICES!

A full function parallel printer interface with unmatched features AND a MASSIVE printer buffer on one card at amazing prices.
 ONLY £90.00 for CACHECARD16 and £130.00 for CACHECARD64

Why wait for your SLOW... printer when you can buy a UK designed CACHECARD for less than most unbuffered printer cards.

FEATURES:

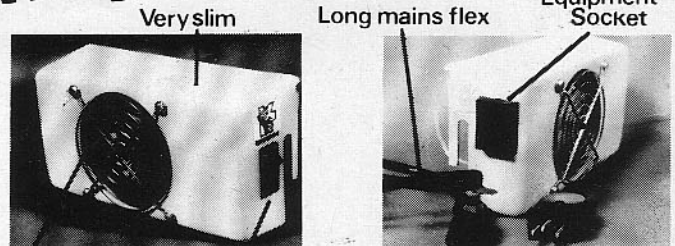
- * 16,000 or 64,000 character buffer controlled by on-card Microprocessor.
- * Full CP/M, PASCAL and APPLESOFT compatible.
- * Screen text echo (can be turned off).
- * 40 or 80 column instant screen dump (80 col. on IIe with 80 column text card and II+ with Videoterm type card).
- * Full function bit-image screen dumps: - Pages 1 and/or 2, double width, inverse, rotate, AND, OR, EX-OR, IIe double-density and full bit density support for Epson FX and Star Gemini printers.
- * Bit-image pictures can be positioned anywhere on the page.
- * Can send ASCII 128 to 255 codes.
- * All features controlled by one interpreter control code giving maximum printer and software compatibility.
- * Control function interpreter can be turned off allowing straight-through text printing guaranteeing compatibility with any software or printer.

Place your order NOW with your dealer or direct with
 CIRTECH on 0383 729770

A stand-alone buffer will be available soon as an add-on if you already have a printer card.

Cirtech also manufacture a range of cards for your APPLE IIe or II+, for example a printer card with the same specification as above but without the buffer for only £32.00. Contact your dealer or CIRTECH for more information.

FROM KEYZONE SLIMFAN £42.50



Very slim Long mains flex Equipment Socket
 Metal finger guard Illuminated switch Apple connector Free plug

- Smallest clip-on fan
- Silent mains motor
- Simple clip-on installation
- Illuminated front facing switch
- Powers up fan/Apple and external socket
- Equipment socket fitted
- Plug supplied free
- Mains suppressors fitted

OTHER KEYZONE PRODUCTS

PRINTERSHARER-PARALLEL (THREE MICROS TO ONE PRINTER) £65	
PRINTERSHARER-SERIAL (THREE MICROS TO ONE PRINTER) £75	
PRINTERCHANGER-PARALLEL (ONE MICRO TO THREE PRINTER/PLOTTER) £70	
RANGE OF CABLES FOR THE ABOVE ARE AVAILABLE	
80 COLUMN DISPLAY CARD For Apple II plus & //e	£120
80 COLUMN DISPLAY INTERFACE For Apple //e (only)	£50
80 COLUMN EXTENDABLE DISPLAY CARD For Apple //e (only)	£60
80 COLUMN-64K EXTENDED CARD For Apple //e (only)	£89
SERIAL COMMUNICATION CARD For Apple II plus & //e	£90
SERIAL PRINTER CARD For Apple II plus & //e	£63
PARALLEL GRAPHIC PRINTER CARD For Apple II plus 7 //e	£60
SPECTAGRAM RGB COLOUR CARD For Apple II plus 7 //e	£120
4 CHANNEL, 8 BIT A/D CONVERTER For Apple II plus 7 //e	£100
4 CHANNEL, 12 BIT A/D CONVERTER For Apple II plus 7 //e	£120

ACCESSORIES

64K Upgrade kit for 80 column extendable card for Apple //e £40
 ALL PRODUCTS CARRY A FULL 12 MONTH GUARANTEE-
 ALL PRICES ARE EXCLUSIVE OF POSTAGE & VAT

KEYZONE LTD, Regeneration House,
 School Road, Park Royal, London NW10 6TD
 Tel: 01-965 1684/1804 Telex: 8813271

ALL KEYZONE PRODUCTS ARE DESIGNED AND MANUFACTURED IN THE U.K. AND ARE SUPPORTED BY THE HIGHEST STANDARDS

A useful guide to Visicalc

Visicalc For The Apple II Plus Computer, by Edouard J. Desautels. (Wm. C. Brown)

THIS 164 page book is part of the Micro Power series. Large pages, multi-ring binder and clear type give an easy-to-read guide which will lay flat on the desk.

I bought the book as a complete beginner and found it an excellent tutor. But it is much more than an introduction and has remained my reference manual.

After a brief introduction to the program and the type of problems which can be tackled, concise instructions are given on the loading and saving of

Visicalc sheets, using DOS, initialising discs and the use of some commands.

At the end of every chapter is a summary of the commands used and techniques covered, making for easy revision.

Next is a simple example of multiplying columns and adding the rows to give an accounting-type worksheet.

Later, the examples are expanded to bring in more advanced replicating techniques and all the mathematical functions. Further DOS techniques and DIF and PF files are explained in detail.

Samples from the worksheets are produced in the same clear type – for once you don't have to squint to decipher

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1 Bank Rate (1)														
2 + For Borrowing	2.50		12.50		12.50		12.50		12.50		12.50		12.50	12.50
3 - For Deposit	4.50													
4 (2) Interest on Borrowing	15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00	15.00
5 Interest Received / Paid	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00	8.00
6 CASH BUDGET	36.22	7.65	46.51	82.07	74.14	-70.29	-245	-175	-151	-128.80	-59.82			
7														
8														
9														
10 OPEN BALANCE / OVERDRAFT	10875	385	1993	12464	13046	10423	-22623	-19129	-10805	-14930	-7059			
11														
12 RECEIPTS:														
13 Credit Sales														
14 Bank Interest	18500	18500	19000	19000	19000	19000	20000	20000	13000	20500	21000	21000	21000	21000
15 Other Income	36	8	47	82	76	0	0	0	0	0	0	0	0	0
16 Sales of Assets														
17 Loans Receivable		825												
18			1000											
19 TOTAL RECEIPTS	18536	18508	35872	19082	21076	20825	20000	20000	13825	20500	21000	21000	21000	21000
20														
21 PAYMENTS:														
22 Goods and Services														
23 Wages and Salaries	7000	7000	7500	8000	8000	8500	9000	9000	9000	9000	6000	6500	6500	6500
24 Taxation	4900	4900	5300	5300	5300	5300	5500	5500	2500	6500	6500	6500	6500	6500
25 Dividend (net)	17126													
26 Capital Expenditure														
27 Loan Repayments		5000	10000	5000	10000	40000	1742							
28 Bank Interest			1500											
29 Loan Interest														
30			900											
31 TOTAL PAYMENTS	29026	16900	25400	18500	23500	34076	14507	11675	17951	12429	17040	17735	28095	28095
32														
33 CLOSING BALANCE	385	1993	12464	13046	10423	-22623	-19129	-10805	-14930	-7059				
34														

printouts.

I found the pace of the book very friendly. Each chapter is about 15 pages long including a short summary, and the commands are introduced gradually to expand the worked examples.

This means you can break off learning at a convenient point and pick it up easily later.

After illustrating all the commands, five case studies are given as examples of real worksheets. Further advice is

given on disc files, scaling for graphs and other techniques.

There is a useful chapter on "When not to use Visicalc" which should save many people wasting time.

The commands are all tabulated for easy reference, and there is a comprehensive index. An optional disc is available which includes all the worksheets. **R.V. Tann**

The Elementary Apple by William B. Sanders
Datamost/Prentice-Hall

DATAMOST, the publishers of the much-acclaimed Kids and the Apple, have now added an Apple title to their Elementary series. As the title implies, it is aimed at beginners.

The book starts right at the beginning with instructions on how to hook up a disc drive or cassette, monitor/TV and printer. Included in this section is some guidance on types of TVs and printers, which suggests that the initial chapter should be read *before* you've bought your complete system.

However, the first little program which you're encouraged to enter switches the printer on and prints a message.

Some other strange advice in this chapter concerns accidentally finding yourself in the monitor. You are advised to press Ctrl-C or Ctrl-B, with no mention of Return.

While the former is not a bad

Alternative to the manual approach

idea, there is no mention of the fact that the latter not only returns you to Basic but zaps your program in the process.

Much of the rest of the book is devoted to teaching a mixture of Basic and DOS commands. (For some strange reason, the author says that DOS is pronounced "DAS", which is news to me and my American friends.)

There are also chapters on advanced topics (Ascii and CHR\$, PEEK and POKE, CALL, etc.), hi and low-res graphics, text files, and printers.

The final chapter concerns programs, hints and helps. Here there is mention of user groups, magazines, and various com-

mercial software packages like word processors, database packages, graphics packages, and a few of the hardware add-ons.

The book aims for a friendly, easygoing style and so the language is chatty and there are lots of cartoon-type illustrations. There are also lots of boxes embedded in the text which contain a variety of asides.

For example under the heading of "How to impress your friends" in Chapter 1 is the information that CALL -1401 has the same effect as PR#6 (which is not quite true – try both with no disc in the drive).

In general, the information in

The Elementary Apple could be gleaned from the various Apple manuals – the DOS manual, the Basic reference manual and the Applesoft Tutorial.

However, many people say they are put off by the manuals, so a book like this may suit them better.

I find the Apple manuals acceptable, and they are certainly better indexed than The Elementary Apple. Points like the Ctrl-B and CALL -1401 mentioned above irritated me, as did statements like "it is necessary to enter at least four POKES to access a hi-res screen".

Using the table immediately above this quote from page 7-15, you can get from text to hi-res in only three POKES.

If you've bought an Apple and hate the manuals, The Elementary Apple might be your kind of book. However, with the price tag of £12.70, I'd be tempted to persevere with the manuals.

Cliff McKnight

I WAS most interested in the article "The Economics of Using Electronic Worksheets" (Windfall, April 1983). This was an interesting demonstration of the use of datagrammes.

However after working through, I concluded that it was rather a case of using a sledgehammer to crack a nut. The desired result can be more easily achieved using normal Visicalc facilities.

A quick tot of the basic datagramme shows that it takes some 1800 key strokes to set up and save. It is then necessary to set up another 11 datagrammes, each of which takes over 550 key strokes to edit and save.

This takes a tidy time to do (it took me nearly two hours) being over 7,850 keystrokes, without allowing for errors and corrections.

Also it is prudent to test every datagramme independently to ensure that no errors have crept in. This takes another two hours or so.

My suggested sequence of actions makes extensive use of the replicate facility, and is as follows:

Set up columns A101 - 155 to K101 - 155 in the same way as the example (Exhibit III page 40) for January. Obviously, the formulae for columns H and I would need to be @SUM(D5... D5) and @SUM(D55... D55) respectively.

To follow the example exactly, to amend the data from February calls for the following sequence:

>D102 and enter "MAR", repeat for E102, H102, I102 (or

use /R).

>D106 and enter "+F5"
>E106 and enter "+F55".

Replicate these two values all the way down the two columns.

/R D106... E106; D107... D134 Rtn then R(egative) twice. This puts values in all the intermediate boxes.

Go to columns H and I and repeat the process.

>H106 /E and amend to read @SUM(D5... F5).

>I106 /E and amend to read @SUM(D55... F55).

Replicate these two formulae all the way down the two columns to row 134, making all values R(egative).

The next job is to tidy up the presentation by getting rid of the excess formulae and replacing the deleted lines.

>D105. This is a row of blank boxes. Replicate to row 112. /R from D105... I105; D112. This writes the blanks over the unwanted data.

>D113 replace the _____ by /-- and replicate them across to column I113 by /R D113... D113; E113... I113.

Replicate these two rows into rows D122 and D123, followed by rows D127 and D128.

The total of key strokes for this procedure, which needs repeating each month is just over 200, depending on how

many mistakes are made.

Against the datagramme process, this would need to be used for 36 months to equal the work involved.

Also the data can be checked more easily by using the Visicalc Boolean features to produce a TRUE/FALSE statement at the bottom of each of the replicated columns, for example:

For March - F41=F142, F91=G142, and @SUM(D41... F41)=H142 and @SUM(D91... F91)=I42.

These also need editing each month, but they ensure the correctness of the amendments to the model and help to counteract possible human error.

A point I missed in the explanation of the example model is that the mode of calculation needs to be set to R(ow) otherwise it needs to be calculated through twice to pick up the correct figures if set to C(olumn).

This is often overlooked, so I prefer to set the summary columns to the right of the base data, from column S onwards.

Provided careful thought is given to the layout of the model so that formulae are not overwritten, this use of the replicate procedure is one of the most useful devices in Visicalc, especially in setting up models. - J.C.R. Hewgill, Chinley, Derbyshire.

● First of all, let me say how much I admire your patience and perseverance having worked through the long example.

While I appreciate the excellent alternative approach which you offer in your letter, especially as it includes the Boolean TRUE/FALSE safeguard, let me assure you that writing the datagramme does not have to be such a formidable task as your letter makes it out to be.

For example, referring to Exhibit IV in the article, it is not necessary to type every line of the datagramme. Having typed in row 7: >D105:+F5, this label

is then replicated from row 8 to row 65.

Each of the entries are then /Edited, so that row 8 reads: >D107:+F6, row 9 reads: >D108:+F7, etc. Next type the datagramme shown in row 66, replicate it from row 67 to 113 and /Edit each line till the rest of the datagramme is completed.

Having completed and saved (with /PF or /PD) the datagramme shown as Exhibit IV, it is now possible to use a word processor, such as Applewriter IIe, to automatically change all the Fs in the datagramme to Gs.

After a little cosmetics we have a ready made datagramme for the second month.

Note that datagrammes written and saved with Applewriter can be executed by VC when loaded with /SL.

Next we have to use the word processor to change all the Fs to Hs and in no time we have the datagramme for the third month, and so on. Each datagramme created with the aid of the word processor must of course be saved under a different file name.

So using a word processor for writing 11 out of 12 datagrammes for each month of the year, life can be made much easier.

Nick Levy

Patience problems

I REFER to the Patience listing in Apple User (January 1984, pages 56 to 59).

On running through the program, the computer is displaying the following message: 3090 BAD SUBSCRIPT ERROR. I have checked the listing, which is exactly as printed on page 57. Can you please help? - O.H. Lake, London N19.

★★★

MR G.R. Gilbertson is either a sadist or I have goofed up

Adding to Jack

THE 1984 March issue of The Apple User carried a review of Incredible Jack under my name which was significantly editorialised before printing. I wish to "add back" two points cut from the published version.

First, the data is stored on a protected disc, which in my view is terrible.

Second, on longish documents a text-formatting bug makes itself evident. The text is saved a screenful at a time. Reformatting on the screen can cause the invention of up to 40 spaces on one of the later

screens.

This fault happened many times while I was testing the program. To make it worse, on two occasions these spaces could not be deleted.

I consider it significant and regrettable that these two points were removed from the review.

I should add that the bugs found in the program were not one-off. They appeared on two different Apples on two different documents. I do not know precisely what caused them. - M.J. Parrott, Stockport.

somewhere (again).

After typing in the *Patience* listing, I spent my perfunctory hour sifting through my typing errors and eventually got it to work.

My query is – are the red kings, queens, jacks and tens all meant to be represented by an inverse @?

And is this supposed to make the game more impossible than it already is?

If not – and I presume this is so as I have read the listing 700 times and can find no reason for it – could you please tell me why they occur and how I can banish them forever from my monitor screen?

If they are just another obstacle to my victory over computers, can you:

Give me Mr Gilbertson's address so that I may throw large stones at his front door at 5am?

Tell me if there's any way to get rid of the little darlings for a while?

Is there any chance of you printing a list of rules that even a dimwit like myself can understand? – **Kevin Hardy, London W12.**

P.S. Love the magazine. (No, I'm not just saying that!)

● It is difficult to guess from afar where programs are going wrong, but Mr Lake's description of line 3090 tells us that TI or TC (the subscript variables) have the wrong values.

If I had the program in front of me, I would ask the computer to print the values of TI and TC. As it happens, both are loop controlling variables (lines 3040 and 3050) and my guess is that there has been a typing (reading) error – namely that one of the three lines has T1 in place of TI.

Kevin's problem is harder to track down, but inverse @ corresponds to Ascii code 0, which suggests that in place of K,Q,J,T zeroes have been put into the array P.

The most likely source of this error is in line 5750 where the Ascii values are adjusted. Again the most likely error is that T1 has been typed in place of TI.

Please, both of you, try these suggestions and if they don't work or lead you on to other ideas send us a listing (preferably with disc) and we will try our best.

Max Parrott

Widen your knowledge

MAY I draw the attention of Apple User readers to the monthly meetings of the Croydon Apple User Group, which aims to widen users' knowledge about their machine, its diverse application and other users.

We have a pool of expertise in many fields and are always looking for new members to broaden our horizons.

For details please phone our secretary, Roger Laming, on 01-681 6842 any time. – **P.C.V. Ramage, South Croydon.**

Loading from disc

I HAVE an ITT 2020 and all was fine until recently when I obtained the Tasc compiler, which only runs with Applesoft. I also have a language card and part-time access to an Apple II.

After saving Applesoft on disc I have tried loading it in various ways, including fooling the intro on the DOS 3.3 master into thinking it was INT, but to no avail.

Can you offer any suggestion? – **D. Poirier, Bishop's Stortford, Herts.**

● A note from Graham Auty in the June 1983 issue of *Windfall* explained a way of accomplishing what you have set out to do.

I am repeating it here

Death isn't terminal

THANKS a lot for many great adventure game reviews. I wish to comment on Denise McKnight's review of *Ultima II* in the October issue.

In it, she mentions that getting killed has a terminal effect on the player disc. I have been playing *U-II* for nine months, but I never experienced this.

I got killed at least 100 times, but when it told me, "Drake is dead", I simply rebooted the game and Drake would be where I entered a castle the last time, or where I saved the game.

because it doesn't appear to enable the language card. To achieve this I have added lines 10 and 80:

Max Parrott

```
10 POKE 49281,1: POKE
49281,1
40 HGR:HGR2:TEXT:HOME:D$=
CHR$(4)
50 VTAB 10: PRINT :LOADING
APPLESOFT INTO LANGUAGE
CARD:
60 PRINT D$"BLOAD FPBASIC,
A$D000"
70 POKE 49246,0: POKE
-16256,0
80 POKE 49282,1: POKE
49283,3
```

Giant strides in Visicalc

WE have recently become the proud owners of an Apple II together with DOS and Visicalc. After reading the Visicalc manual and hands on experience, we managed to formulate personalised worksheets, although very simple at this stage.

However I managed to come across a couple of back copies of *Windfall* and have made relatively giant strides after only reading two of the Visicalc articles.

The October, 1982 article, "Technique with a touch of magic", solved a problem for me which I had spent a day trying to formulate with much disap-

pointment because I could not get my formulations to work the way I wanted them. Now they do.

The beauty of these articles is that they are written in such a manner that even a new boy such as myself can follow them without too much difficulty. – **G.R. Kelleher, Zimbabwe.**

A pleasant change

THERE has been another rash of letters in the computing press complaining of disinterest shown by British firms in selling their products and of poor or non-existent back up service.

I have myself recently been chagrined to receive an answer from a firm in Taiwan – before that from a British firm – in response to a similar question.

It is very pleasant therefore to have experienced quite a different response from Dark Star Systems when my Apple would not perform with one of their Copykit cards.

First I was listened to carefully and given a great deal of advice about what could be wrong. Then, when things still did not work out, I was invited to return the card and was sent another.

Unfortunately the same fault was displayed and it now began to be fairly obvious that the problem lay in my machine.

Despite this, however, I was cheerfully invited to send both my RAM card and the Copykit back to be tested together. Both worked correctly on all Dark Star's Apples and this seemed to be the end of the line since it was now apparent that my Apple had some obscure fault.

I was far from deserted however, and in fact Dark Star sent diagnostic material for me to run. They even eventually rewrote their software to accommodate one of my non-standard chips which should have been innocuous.

All this was cheerfully and efficiently done and I was never allowed to feel for a moment that I would ever be abandoned as a hopeless case.

Three hearty cheers for Dark Star. May other firms take note and behave likewise. – **A.C. Game, Felixstowe.**

Interface cards for the Apple II used to be expensive – anything between £100 to £200 and sometimes more just for the privilege of being able to link up peripherals such as printers (vital), plotters and modems, or to give yourself an 80 column display.

However alongside Apple's own moves to take its Apple II and III family into the Lisa age, there is now a budding market in lower-cost traditional interface cards.

Some of the most successful low-cost card makers are in Britain. Their main problem has been trying to persuade the marketplace that low price does not necessarily mean poor quality.

They may be right, judging from test *Apple*

User has carried out on a variety of cards from Scottish-based Cirtech. The cards are all designed and manufactured in the UK.

The company's Z80 card works perfectly with a variety of programs, including Wordstar, and peripherals. It costs £45.

A printer interface with a graphics dump capability, which Cirtech claims equals the best products currently available, costs only £32. It is also remarkably small.

And we couldn't fault the £50 80 column card (upgradable to give an extra 64k memory) for the IIe.

The latest additions to this range are two parallel printer cards with a built-in buffer and full graphics dump facility.

CIRTECH has extended its range of peripheral cards to include two for printers, the Cache16 and Cache64.

These are parallel printer cards with a built-in 16k or 64k buffer and full graphics dump facility.

They can be used in both the Apple II and IIe and are automatically recognised by DOS CP/M and Pascal operating systems.

They share the same built-in features as Cirtech's unbuffered printer card, so that much of the following is applicable to all three.

In keeping with Cirtech's

other products, the Cache cards are small, compact and neatly constructed.

Cirtech says it uses state-of-the-art techniques to produce cards of this size and quality, the advantages of which are increased reliability, better cooling and, of course, lower cost.

The Cache card contains either 16k or 64k of RAM controlled by an on-board microprocessor.

In use, such as printing a document from a word processor, data is sent to the Cache card at very high speeds – up to 19,000 characters a second – where it is stored in

RAM.

As soon as the data has been transferred to the buffer the Cache card signals to the computer to say that it has received the data and you are then able to continue with your program.

At the same time the data stored in RAM continues to be fed out to the printer at the slower rate until the buffer is empty.

If you decide you want to stop the printer you can do this by pressing the Reset key, which causes the printer buffer to be cleared.

The Cache card therefore speeds up printing considerably because you don't have to wait for the printer to finish printing before you carry on with your program.

The printer card can be used in any Apple slot except 0, although CP/M and Pascal expect to find it in slot 1. It can be activated in Basic by the usual PR#n command, where n equals the slot number.

The cards behave much the same as any other printer card.

However there are certain built-in features, including full graphics facility, disable/enable line feed, echo and non-echo

characters to the screen, dump the 40 column screen and dump the 80 column screen.

If you require any of these features, they can be enabled by typing Ctrl-Z followed by one of the following codes:

- C** deactivates the card, allowing straight through printing.
- F** disables line feeds after carriage return.
- P** does not echo characters to the screen during printing.
- S** turns screen echo back on.
- T** produces copy of 40 column text screen.
- BT** produces copy of 80 column text screen.

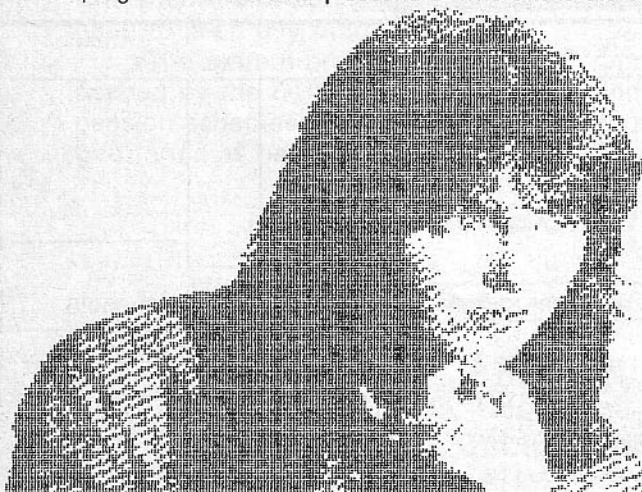
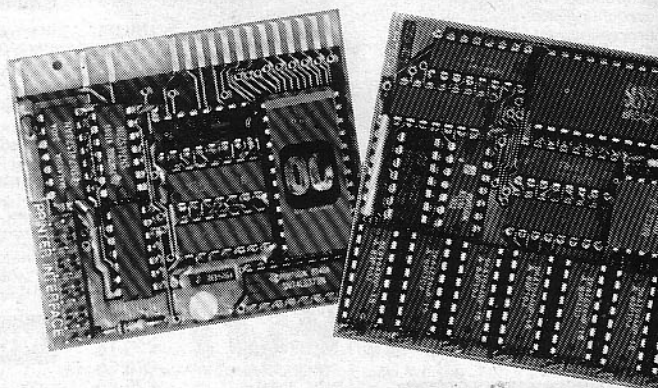
Ctrl-Z was chosen by Cirtech because it is rarely used by software houses. If this is a problem the card can be completely disabled by typing Ctrl-Z C, in which case everything sent to the card is in turn sent to the printer.

All three Cirtech printer cards have built on board software which enables a bit image copy of the Apple's hi-res screen to be dumped onto certain dot matrix printers.

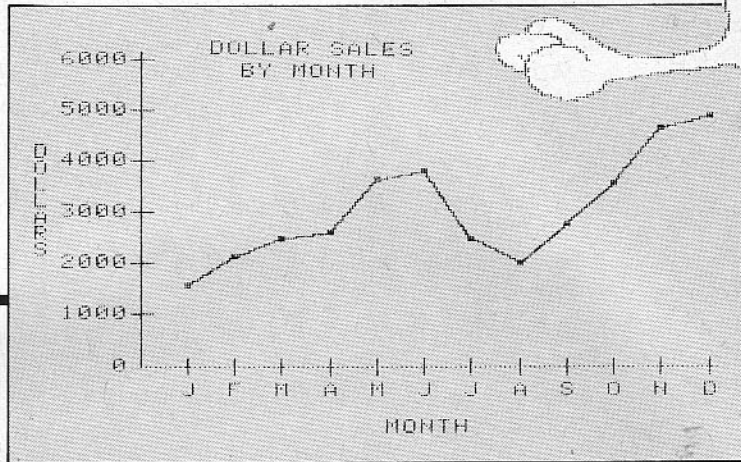
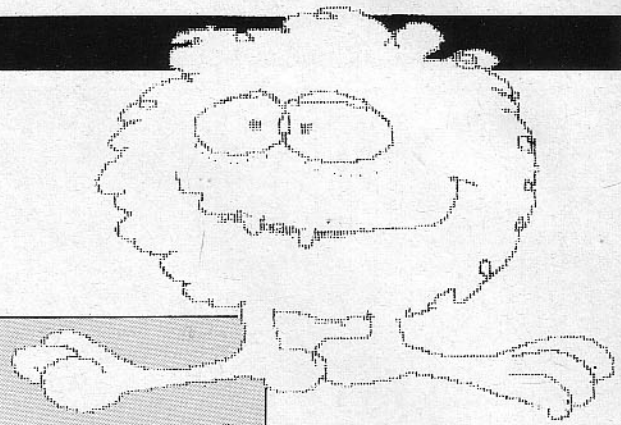
To use the software, you first need to load a picture into the hi-res memory, such as page 1 HGR.

Pep - up your slotting in a

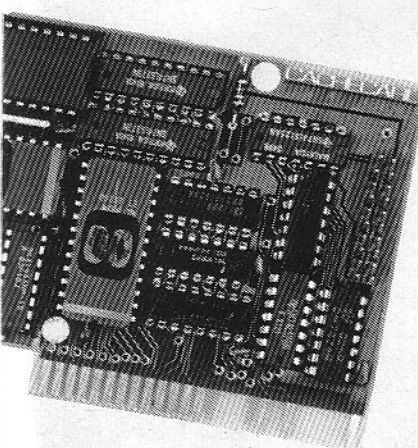
Parallel printer cards to speed up production of printouts are reviewed here by PETER ASHFORD



printouts by card



All printouts shown here were produced with the aid of the Cache cards



The printer card is then activated by typing PR# 1 followed by Ctrl-Z Return and the printer will then print according to the card's default setting which are page 1, non inverse, not rotated, standard density.

It is possible to manipulate the image in a number of ways by combining page 1 and page 2 in various "logical" ways.

In addition the image density, position and orientation can be changed by using the following simple commands:

Density

- L double density
- Y double density, double speed (Epson FX only)
- Z quad density, double speed
- y double density, double speed (Star Delta only)
- z quad density (Star Delta only)

Page selection

- 2 HGR page 2
- B Both pages 1 and page 2
- / Special Ile dump of 560 x 192 (need 64k 80-col card)

Mixed page selection

- O logic 'or' of page 1 and page 2

A logic 'and' of page 1 and page 2

E logic 'EX-OR' of page 1 and page 2

In practice, the picture dumps were satisfactory and compared well with dumps made using different software.

The Cache16 and Cache64 cards were of course considerably faster than the standard non-buffered card (see bench marks).

The following tests were made using an Apple II with a Grappler plus printer card, standard Cirtech parallel card and Cache64 card, connected to an Epson MX100 printer. The times quoted are in seconds.

Listing three pages of text to printer:

	Control returned to Apple	Print out completed
Grappler+	157	155
STD Cirtech	152	150
Cache64	24	150

Print three pages of text using Wordstar:

	Control returned to Apple	Print out completed
Grappler+	182	180
STD Cirtech	147	145
Cache64	60	130

Dump one page of high-res screen:

	Control returned to Apple	Print out completed
Grappler+	45	44
STD Cirtech	39	38
Cache64	4	35

Conclusions

In keeping with other Cirtech products, the new Cache16 and Cache64 represent very good value for money and indeed can reduce printing times quite significantly.

It is fashionable, and often necessary, to criticise aspects of review products. However the cards performed faultlessly during the few weeks of this review.

The non technical user should be reassured that the

cards are easy to use. In most cases, they are simply plugged into slot 1, connected to the printer and forgotten about.

At the same time the three printer cards have some very powerful features, such as comprehensive graphics and text dump facility, in addition to the generous buffer capacity of the two Cache cards.

Who would benefit from such a card? You, whenever you say "I wish the printer would hurry up and finish!"



Product: Cache16 and Cache64 interface cards
Description: Parallel printer cards with built in buffer and full graphics dump facility.
Manufacturer: Cirtech, PO Box 29, Dunfermline, Fife
Price: £90 (Cache16); £130 (Cache64)

From Page 26

```

) = - 50:ZM(4) = 50: REM U
SER VALUES
140 ZM(5) = 0:ZM(6) = 260:ZM(7) =
155:ZM(8) = 0: REM SCREEN V
ALUES
150 GOSUB 40200: REM SET MAPPI
NGS
160 ZG(1) = 8:ZG(2) = 5: GOSUB 40
400: REM BORDER AND TICK MA
RKS
170 ZN = 20: REM NUMBER OF DATA
POINTS
180 FOR J = 1 TO 3:ZT = J
190 FOR I = 1 TO ZN
200 ZX(I) = - 50 + I * 5:ZY(I) =
(20 + ZX(I) ^ 3 / (2000)) *
(J - 2)
210 NEXT
220 GOSUB 42600: REM PLOT POINT
S
230 NEXT
240 ZC(10) = 2: GOSUB 41600: REM
CURSOR 2 X,Y VALUES
250 END :
41600 REM

```

CURSOR ROUTINE

```

41610 REM CURSOR VALUES IN ZC(1
)-ZC(10)
41620 REM 1-4 USER X,Y COORDS
RETURNED
41630 REM 6-9 SCREEN X,Y COORD
S
41640 REM 5,10 GAME BUTTON VALU
E,COUNT
41650 IF ZP < > 1 THEN RETURN
: REM ONLY WORKS FOR PAGE 1
41660 Z1 = PEEK (231):Z2 = PEEK
(249): REM SAVE CURRENT SCA
LE AND ROT
41670 SCALE= 10: ROT= 0:Z3 = 279
.01 / 255:Z2 = 191 / 255:Z4 =
191 / 255: REM PADDLE SCALE
FACTORS
41680 ZT(0) = 1: GOSUB 42200: REM
SET POINTERS TO SHAPE TABLE
41690 IF ZF = 1 THEN POKE - 16
301,0: REM TEXT+GRAPHICS
41700 Z5 = 0: HOME : IF ZC(10) >
2 OR ZC(10) = 0 THEN ZC(10) =
2
41710 ZC(1) = INT ( PDL (0) * Z3
)

```

```

41720 ZC(2) = INT ( PDL (1) * Z4
)
41730 HTAB 5: VTAB 22: CALL - 8
68
41740 PRINT "X = "; INT (( FN UX
CN(ZC(1))) * 1000) / 1000: TAB(
20): " Y = "; INT (( FN UYCN(
ZC(2))) * 1000) / 1000
41750 XDRAW 9 AT 0,ZC(2)
41760 XDRAW 10 AT ZC(1),0
41770 ZC(3) = INT ( PDL (0) * Z3
)
41780 ZC(4) = INT ( PDL (1) * Z4
)
41790 ZC(5) = PEEK ( - 16287): REM
BUTTON
41800 IF ZC(5) > 127 THEN 41850
41810 IF ZC(1) = ZC(3) AND ZC(2)
= ZC(4) THEN 41770: REM NO
CHANGE
41820 XDRAW 9 AT 0,ZC(2)
41830 XDRAW 10 AT ZC(1),0
41840 GOTO 41710
41850 REM STORE NUMBER
41860 CALL - 198
41870 Z5 = Z5 + 1
41880 ZC(6 + 2 * (Z5 - 1)) = ZC(1
)
41890 ZC(7 + 2 * (Z5 - 1)) = ZC(2
)
41900 IF Z5 = 2 THEN 41930
41910 FOR ZL = 1 TO 100: NEXT ZL
: REM DELAY
41920 GOTO 41820
41930 REM EXIT CURSOR ROUTINE
41940 HOME : IF ZF = 1 THEN POKE
- 16302,0: REM FULL GRAPHI
CS
41950 XDRAW 9 AT 0,ZC(2)
41960 XDRAW 10 AT ZC(1),0
41970 SCALE= Z1: ROT= Z2: REM R
ESTORE SCALE AND ROT
41980 ZC(1) = FN UXCN(ZC(6)):ZC(
2) = FN UYCN(ZC(7)): IF ZC(
10) = 1 THEN 42000
41990 ZC(3) = FN UXCN(ZC(8)):ZC(
4) = FN UYCN(ZC(9))
42000 RETURN :
42200 REM

```

SHAPE TABLE POINTER ROUTINE

```

42210 REM SETS 232 AND 233 TO P
OINT TO REQUIRED SHAPE TABLE
42220 REM ZT(0) = 1-9 REQUIRED
TABLE

```

```

42230 REM ADDRESSES SET BY LOAD
ER ROUTINE (42400)
42235 IF ZT(0) < 1 OR ZT(0) > ZT
(10) THEN TEXT : PRINT "NO
SUCH SHAPE TABLE ZT(0) = "ZT
(0): CALL - 198: STOP : RETURN
42240 ZA = INT (ZT(ZT(0)) / 256)
: REM HIGH BYTE
42250 ZB = INT (ZT(ZT(0)) - ZA *
256): REM LOW BYTE
42260 POKE 232,ZB: POKE 233,ZA
42270 RETURN :

```

42400 REM

SHAPE TABLE LOADER

```

42410 REM ZT(1)-ZT(9) TABLE AD
DRESSES (DECIMAL)
42420 REM ZT(10) NUMBER OF TAB
LES TO BE LOADED
42430 REM ZT*(1)-ZT*(9) TABLE
NAMES
42440 ZT(10) = 1: REM ONLY ONE T
ABLE SO FAR
42450 ZT(1) = 2048:ZT*(1) = "PLOT
TABLE"
42460 REM ROOM FOR MORE TABLES
42540 FOR ZI = 1 TO ZT(10)
42550 PRINT CHR$(4)"BLOAD ";ZT
$(ZI);",A"; STR$(ZT(ZI))
42560 NEXT
42570 RETURN :
42600 REM

```

POINT PLOTTER

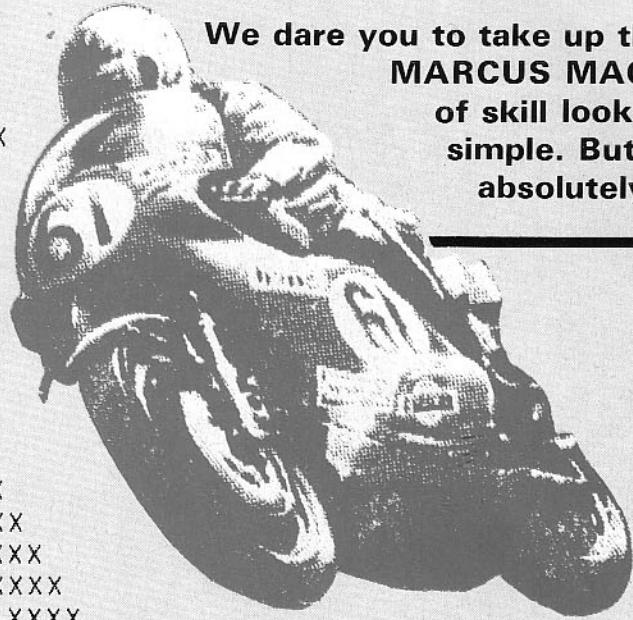
```

42610 REM PLOTS POINTS IN ZX(I)
AND ZY(I)
42620 REM ZN NUMBER OF POINTS
42630 REM ZT = 1-8 PLOTTING SY
MBOL
42640 ZT(0) = 1: GOSUB 42200: REM
SET SHAPE TABLE
42650 FOR ZI = 1 TO ZN
42660 XP = FN XCN(ZX(ZI)):YP = FN
YCN(ZY(ZI))
42670 XP = INT (XP + .5):YP = INT
(YP + .5)
42680 IF XP < ZM(5) OR XP > ZM(6
) THEN 42710
42690 IF YP < ZM(8) OR YP > ZM(7
) THEN 42710
42700 DRAW ZT AT XP,YP
42710 NEXT
42720 RETURN :

```


SPEEDWAY!

We dare you to take up the challenge!
MARCUS MACRAE's game
of skill looks deceptively
simple. But you'll find it
absolutely compelling!



THIS challenging racing
game is quite speedy, des-
pite the fact that it is only
written in Basic. In it the
player must avoid cars, oil
slicks and grenades while
driving down a road.

The oil does not necessarily
represent a fatal hazard. It just

makes you skid in unexpected
directions.

If you collide with another
vehicle or the edge of the road
you lose a life. A grenade, not
surprisingly, eliminates two of
your allotted three lives.

Steering is achieved using
the left and right arrow keys.
You move in a straight line until
an arrow is pressed. There are
five possible directions -
straight down, one press of an
arrow (moves you right or left)
and two arrow presses (right or
left).

An interesting feature of the
program is the Hall of Fame
subroutine (lines 1050 to the
end) which keeps a record of the
top 10 scores. It can be added to
other scoring-type programs
provided you use SC to repre-
sent the score.

```

YOU ARE A BIKER, LOOKING LIKE 'O'.
YOU MUST AVOID THE OBJECTS COMING AT YOU

'...' IS OIL - MAKES YOU SKID
'<O>' & '<=>' ARE CARS
'+' IS A GRENADE - YOU LOSE TWO LIVES

USE THE ARROWS TO CHANGE ANGLE OF MOTION

'RETURN' HALTS

HIT ANY KEY TO CONTINUE
    
```

```

XXXXX  O          XXXXX
XXXXX  O<=>       XXXXX
XXXXX  O          XXXXX
XXXXX  + O        XXXXX
XXXXX  + O        XXXXX
XXXXX          <=> XXXXX
XXXXX          +   XXXXX
XXXXX          XXXXX
XXXXX          XXXXX
XXXXX  ... <O>   XXXXX
XXXXX          <O> XXXXX
XXXXX          XXXXX
XXXXX          ..  XXXXX
XXXXX          ..  XXXXX
XXXXX          XXXXX
XXXXX+         XXXXX
XXXXX          XXXXX
XXXXX          XXXXX
XXXXX          XXXXX
XXXXX          XXXXX
XXXXX  ... O     XXXXX
XXXXX          O   XXXXX
XXXXX          O<O> XXXXX
XXXXX          O   +   XXXXX
XXXXX          O   XXXXX
XXXXX          O   XXXXX
XXXXX          O   XXXXX
XXXXX          +O   XXXXX
XXXXX          )   XXXXX
XXXXX          XXXXX
XXXXX          +   XXXXX
XXXXX          XXXXX
XXXXX          XXXXX
XXXXX          <=> XXXXX
XXXXX          XXXXX
XXXXX          XXXXX
XXXXX          XXXXX
XXXXX          XXXXX
XXXXX          <O> XXXXX
XXXXX          XXXXX
XXXXX          <O> XXXXX
XXXXX          +   XXXXX
XXXXX          <=> XXXXX
XXXXX          O   XXXXX
XXXXX          O   +   XXXXX
XXXXX          O   .   XXXXX
XXXXX          O   XXXXX
XXXXX  ... O     XXXXX
XXXXX          O   XXXXX
XXXXX          O   XXXXX
    
```

```

100 REM      SPEEDWAY          XXXX"
110 REM      BY MARCUS MACRAE  170 REM 14 SPACES
120 DIM B$(20):B$(0) =      180 HOME :A = 8:A1 = 0:X1 =
    "<=>":B$(1) = "...":      0:X = 19:B = 13: POKE -
130 B$(2) = "...":B$(3) =    16368,0
    "+":B$(4) = "+":B$(5) =  190 FOR I = 1 TO 23: VTAB
    "<O>"                      I: HTAB A: PRINT A$: NEXT
140 TEXT : HOME : GOSUB      200 REM NEXT LINE PRINTS
    530: HOME                  WALL
150 SC = 0:CR = 0           210 VTAB 23: HTAB A: PRINT
160 A$ = "XXXX"
    A$;: VTAB 23: HTAB B:      - 2
    PRINT B$;: CALL - 912      280 IF X1 > 2 THEN X1 = 2
220 Q = PEEK (- 16384): IF    290 REM CHECK FOR CRASH
    Q < 128 THEN 300          300 S = SCRN( X + X1 -
230 IF Q = 141 THEN 220      1,16) + 16 * SCRN( X +
240 POKE - 16368,0          X1 - 1,17)
250 IF Q = 136 THEN X1 = X1  310 REM NO CRASH
    - 1                       320 IF S = 160 THEN X = X +
260 IF Q = 149 THEN X1 = X1  X1: GOTO 390
    + 1                       330 REM OIL
270 IF X1 < - 2 THEN X1 =    340 IF S = 174 THEN X1 =
    
```

```

XXXXX  +          XXXXX
XXXXX  <=>       XXXXX
XXXXX          ..  XXXXX
    
```

```

INT (5 * RND (1)) - 2:
GOTO 390
350 REM S=171 => GRENADE
360 IF S = 171 THEN CR = CR
+ 1
370 CR = CR + 1: FOR I = 161
TO 170: VTAB 9: HTAB X +
X1: PRINT CHR$(I);: FOR
J = 1 TO 30:P = PEEK (-
16336): NEXT : NEXT : IF
CR > 2 THEN 520
380 GOTO 180
390 :
400 VTAB 9: HTAB X: PRINT
"0";
410 REM CHOOSE OBSTACLE
420 R = INT (15 * RND
(1)):B$ = B$(R): IF B$ <
> ** THEN SC = SC + 100
430 REM ROAD MOVEMENT
440 REM A1 IS X-COORD
OFFSET
450 A2 = ( INT (10 * RND
(1)) - 2): IF A2 < 3 THEN
A1 = A2
460 IF A + A1 < 1 THEN A1 =
0:A = 1
470 IF A + A1 > 17 THEN A1
= 0:A = 17
480 A = A + A1
490 REM POSITION OF
OBSTACLE
500 B = INT (11 * RND (1))
+ 4 + A
510 SC = SC + 10: GOTO 210
520 HOME : VTAB 7: PRINT
"YOU SCORED ";SC: GOSUB
1050: GOTO 130
530 POKE - 16368,0
540 PRINT "YOU ARE A BIKER,
LOOKING LIKE '0'.": PRINT
550 PRINT "YOU MUST AVOID
THE OBJECTS COMING AT
YOU": PRINT
560 PRINT "... ' IS OIL -
MAKES YOU SKID": PRINT
570 PRINT "'<0>' & '<=>'
ARE CARS": PRINT
580 PRINT "'+' IS A GRENADE
- YOU LOSE TWO LIVES":
PRINT : PRINT
590 PRINT "USE THE ARROWS
TO CHANGE ANGLE OF
MOTION"
600 PRINT : PRINT "'RETURN'
HALTS"
610 PRINT : PRINT : PRINT
"HIT ANY KEY TO CONTINUE"
620 IF PEEK (- 16384) <
128 THEN 620
630 RETURN
1000 REM THIS SECTION CAN
BE USED SEPARATELY,
1020 REM PROVIDED "SC"
REPRESENTS THE SCORE.
1040 REM
1050 P = 0: POKE - 16368,0
1060 IF SC > A(P) THEN X =
P + 1: GOTO 1090
1070 P = P + 1: IF P = 10
THEN FOR I = 1 TO 2000:
NEXT : GOTO 1110
1080 GOTO 1060
1090 VTAB 10: PRINT "YOUR
SCORE IS IN THE TOP
TEN!": PRINT : INPUT
"PLEASE ENTER YOUR NAME
";SC$: IF LEN (SC$) > 25
THEN 1090
1100 FOR I = 9 TO P + 1
STEP - 1:A$(I) = A$(I -
1):A(I) = A(I - 1): NEXT
:A$(P) = SC$:A(P) = SC
1110 HOME : HTAB 14: PRINT
"HALL OF FAME": HTAB 14:
PRINT "=====
"
1120 : FOR I = 0 TO 9: VTAB
I + 2 + 5: HTAB 5: PRINT
".....": NEXT
.....": NEXT
1130 FOR I = 0 TO 9: VTAB I
+ 2 + 5: HTAB 1: PRINT I
+ 1".": HTAB 5: PRINT
A$(I);: HTAB 40 - LEN (
STR$(A(I))): PRINT A(I):
NEXT
1140 IF PEEK (- 16384) <
128 THEN 1140
1150 RETURN
    
```

Lower case text for everyone!



One of the plus points about the Apple IIe is its ability to display upper and lower case characters on the screen – something that has usually not been possible on the Apple II without an expensive modification.

A special offer for Apple User readers is a lower case generator that will enable you to have this valuable enhancement for just £25.

And that price includes a useful pair of chip extraction tongs (to ensure you don't bend any of the 'pins), installation instructions and a small Basic listing, plus copies of helpful articles on the subject.

This product is compatible with most Apple computers. However, some of the earlier versions (ie. Revisions 0-6 Mother Boards) require an adaptor board in addition to the generator chip itself. The cost of this ancillary board is £15.

(Users of the older Applewriter I Word Processing package should note that a

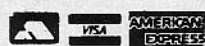
modification is needed before the program can use the generator. We can do this for you if you send a COPY of your program, together with the additional sum of £2.50.)

ORDER FORM

Allow 28 days for delivery

Lower case generator at £25
 Adaptor board at £15
 Total £

Cheque enclosed
 Credit card
 No.



Name
 Address

Send to: Apple User, FREEPOST, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY. No stamp needed if posted in UK.

Ramdrive thinks its a disc

PRICES of peripherals may be plummeting on today's market, but even trusting readers of *Apple User* might find it difficult to believe that you can get an Apple disc drive for under £30.

And you can't. However what you can get is Ramdrive IIe, a software program which uses the IIe 64k or 128k extended memory 80 column card to emulate a fast disc drive.

Author Precision Software claims that Ramdrive IIe runs on all extended memory cards and is compatible with DOS 3.3, Apple Pascal 1.1, the 80 column display and double high resolution graphics.

The software is self-documented – you only get instructions after booting – and is supplied with DOS 3.3 and Pascal versions on the same disc.

The DOS 3.3 version can be set up to auto start which allows, for example, the automatic transfer on booting of a choice of utilities onto the Ramdrive in Slot 3, making them ready for immediate use.

The disc costs £29.95. It can be modified and is copyable.

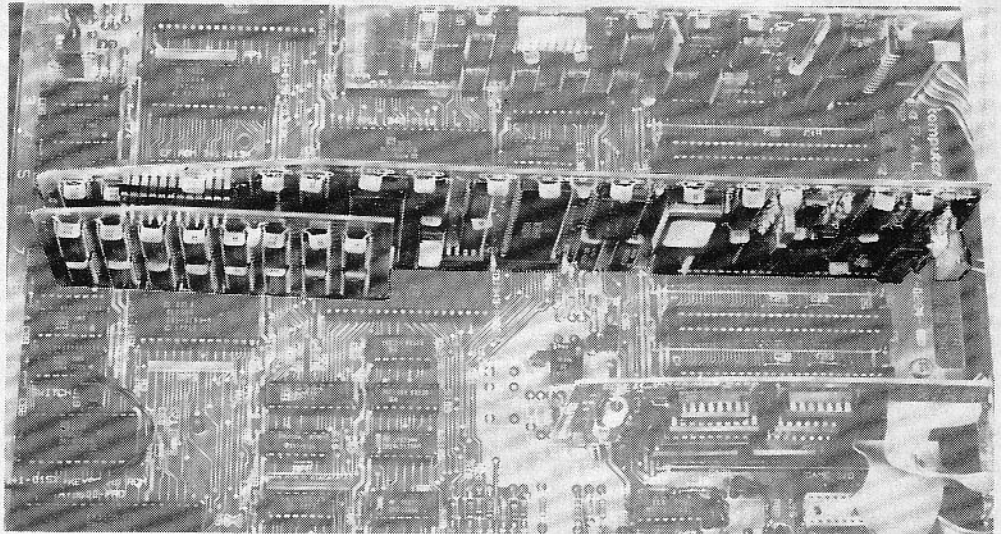
● *Coastal Computing Supplies and Services, 16 Malt Kiln Road, Newbiggin, Ulverson, Cumbria LA12 0RJ. Tel: 0229-88408.*

Lisa joins legal set

PLANS to develop software products for lawyers to run on Apple's Lisa family including the new Lisa 2/5, 2/10 and Macintosh, have been announced by Compu-Law.

They expect to start selling a Lisa Quickport version of the Client Management System 3.1 – a time management, legal billing and report production package for law firms – this month.

"The Lisa range has the



CP/M Gold Card, plus piggyback memory card, installed in an Apple IIe

A monster on the motherboard

IMAGINE a card nearly as long as the Apple II motherboard which can incorporate, piggyback fashion, a 128k memory cache card.

That is the CP/M Gold Card, the first product from the new Digital Research hardware division.

It fits into any slot on any machine in the Apple II series, and the makers claim it allows CP/M applications to run on

these machines up to three times faster than competitive cards.

The CP/M Gold Card uses a 6 MHz Z80 microprocessor with 64k of on-board memory and an optional cache memory of 128k. This, claims DR, virtually eliminates the time required to gain access to and read from a floppy disc.

It also provides an 80 column display on the standard Apple

monitor instead of the usual 40 column arrangement and includes the most recent version of CP/M – CP/M Plus – CBasic, a program assembler and a program debugger.

The Gold Card will sell in two configurations – £399 for basic capabilities and £640 with the 128k cache memory.

● *Digital Research, Oxford House, Oxford Street, Newbury, Berkshire RG13 1JB. Tel: 0635 35304.*

power and future expandability needed to address the demanding legal software market", said Lori McCreary, of Compu-Law. "We will eventually take advantage of Lisa's ability to run multi-user applications".

The Client Management System 3.1 software, priced at \$2,495, is designed to provide small to medium size law firms with data processing capabilities affording control, speed and flexibility on a scale the manufacturers claim was only previously possible on main frame or mini computers.

The program stores all client and case information, time, services, costs and payments online. It also produces billing statements and a variety of time and financial analysis manage-

ment reports.

It handles legal accounting needs, such as trust, retainer and unbilled time accounting, and all input and reports are customised to a firm's needs.

● *Compu-Law, 3520 Wesley Street, Culver City, CA 90230. Tel: (0101) 209-781 2461.*

Magic in the office

THE influence of Lisa on the micro market has been dramatic.

So much so that Lisa-like hallmarks such as cut-and-paste and true software integr-

ation are now becoming the norm on new releases for the Apple II and III.

The Magic Office System, released last month in the US, combines word processing, an electronic spreadsheet and a spelling checker.

Distributor Artsci says the word processor is a "what you see is what you get" system.

The spreadsheet is said to be more powerful than Visicalc and will load Visicalc files. The spelling checker will automatically correct a document.

The products are integrated through a file folder and file cabinet display that simplifies their use.

Each data disc is organised into a single file drawer with multiple discs making up mul-

tiple drawers. A file drawer can contain file folders, stationery, word processing and spreadsheet documents.

Documents and folders can be moved or copied anywhere in the office system and parts of documents can be cut and pasted into other documents.

The program runs on a 64k Apple with 80 columns and two disc drives. It costs \$295.

● *Artsci Inc., 5547 Satsuma Avenue, North Hollywood, California 91601. Tel: (0101) 818-985 2922.*

Graphics galore

THE Fontrix from Data Transforms is a character generator, drawing board, colour painting system, typesetting device and a means of combining the printed word with colour graphics on a screen large enough to fill a printed page.

Its multi-purpose graphics are built around three components: the Font Editor, from which character sets are created, the Graphic Writer, from which the Font Editor sets may be used to write on either single screen graphics or extended screen, and the Graphics Dump, which prints graphics.

It retails at £55.

● *Pete & Pam Computers, New Hall Hey Road, Rossendale, Lancs BB4 6JG. Tel: 0706 212321 & 227011.*

Printer interface

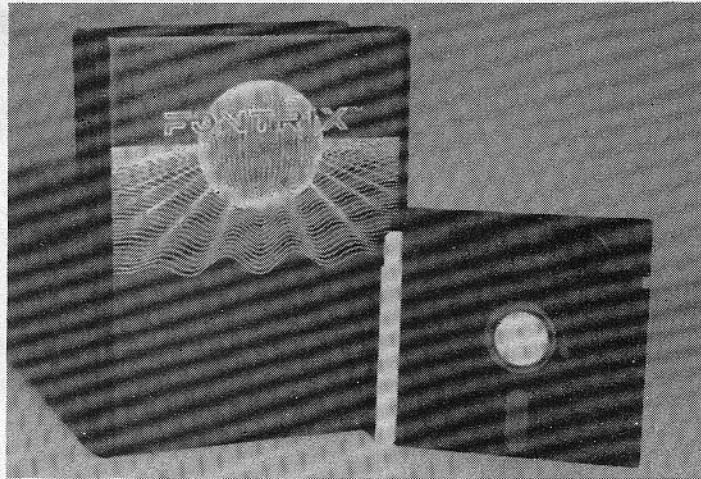
NEW from Videx is the Uniprint Parallel Printer interface for the Apple II range.

It provides graphics transfers of hi-res pages 1 or 2, expands or shrinks the image, or rotates it at right angles.

Colour transfers are also possible on the Prism printer.

The manual provides a detailed step-by-step procedure for installation and includes pre-tested configurations for the popular parallel printers including Epson, C. Itoh, Apple DMP and Annadex.

The price of \$89 includes a



Fontrix character generator

Centronics-compatible cable.

● *Videx, 1105 N.E. Circle Blvd., Corvallis OR 97330. Tel: (0101) 503-758 0521.*

Doctor's assistant

AN updated version of *Medicore*, the software system designed to assist general practitioners with the documentation for their patients, has been announced by Northern Computers.

It runs on an Apple II using a standard printer together with either floppy or hard disc options. Programmed in Pascal, it is suitable for the one-man practice, but is easily expanded to accommodate group practices.

The package provides a full range of documentation including age/sex register of patients, required follow-up treatment prompter and repeat prescriptions.

With the floppy disc version up to 30 additional items of information may be added to each patient.

However with the hard disc option it is possible to accommodate the Royal College of General Practitioners' index list containing up to 2,000 diseases.

Medicore is available as a software package or as a complete system, including all necessary hardware.

● *Medicore Northern Computers, Churchfield Road, Frodsham, Cheshire. Tel: 0928 35110.*

Quark cuts the cost

A SUBSTANTIAL price reduction for the *Word Juggler* word processing programs for the Apple III and IIe has been announced by Quark Incorporated.

The company reduced *Word Juggler IIe* from \$229 to \$189 and *Word Juggler III* from \$295 to \$229 and has incorporated into the package the *Lexicheck* spelling program, which previously retailed separately for \$139.

Word Juggler is written in Assembly and the IIe version is claimed by Quark to be the first commercial program released for Apple's new ProDOS operating system.

The *Lexicheck* option has a 50,000 word dictionary and a new feature called *Word Guess Plus*, which provides the user with possible alternative spellings when using the spelling checker and when preparing a document with *Word Juggler*.

● *Pace Software Supplies, 92 New Cross Street, West Bowling, Bradford BD5 8BS.*

Removeable Winchester

A BRITISH made removable Winchester disc system is claimed to eliminate the Win-

chester backup problem outright.

At the same time the system is said to provide unlimited data storage, reliable interchangeability, flexibility, high quality backup and easy to use selective restore facilities.

The Intec 505 uses closed loop track following and an embedded servo to ensure accurate positioning and overcome problems of locating data on densely packed discs.

A purge cycle eliminates dust problems and "creates the near equivalent of a sealed environment", according to the manufacturers.

Because backup copies are made to a hard disc it is possible to interrogate and back down individual files.

Newbury Data has bought the rights to manufacture the disc used in the Intec 505 system under licence in the UK.

The unit is designed for the Apple and derivatives with full software implementation in DOS, ProDOS, CP/M Plus and Pascal.

"Winchesters offer speed and space, floppies provide portability. Our 505 combines the advantages of both", claims Intec.

● *Intec, 41a Knight's Hill, West Norwood, London SE27 OHS. Tel: 01-761 5999.*

Chasing the Lotus

A SPREADSHEET which is claimed to be twice as fast as Visicalc and just as fast as the Lotus 1-2-3 package – and which costs only £69.95 – is now available on the Apple II range.

Practicalc II has already established a good reputation in the last year on the Commodore 64 and Vic 20 micros.

Features include alphabetic and numerical search and sort, graphics capabilities, automatic and manual recalculation, variable column width in all columns, upper and lower case entry and printing and an on-screen default menu.

● *Marketing Micro Software, Goddard Road, Whitehouse Industrial Estate, Ipswich IP1 5NP. Tel: 0473 462721.*

A modem to speak to the world

ALL major world communications standards are supported by the WS2000 modem, a direct wired, multi-speed unit from Minor Miracles.

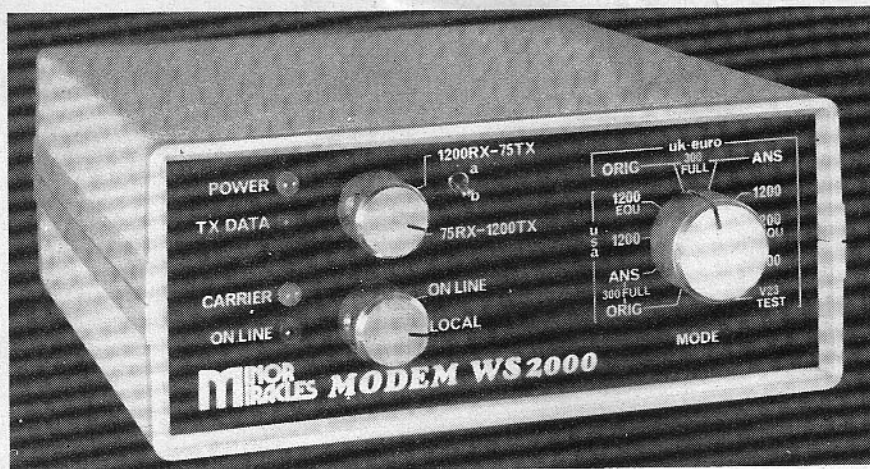
It is switchable to 300 Baud full duplex, 600 half duplex, 1200 half duplex and 1200/75 back-channel (Prestel standard) as well as both Bell (USA) and CCITT (UK-Europe) standards.

Special features include reverse 1200/75 Baud operation which allows the modem to communicate with dedicated Prestel/Micronet/Viewdata terminals. The WS2000 costs £99.95

With the optional plug-in auto-dial/auto-answer board at £39 and a special control lead set (£9.50), the modem can be set up to answer the phone, scan the incoming carrier and set itself to that standard before putting the computer on line.

The WS2000 is supplied with British Telecom modular line cord and plug, and parallel telephone socket on the back panel.

● *Minor Miracles, P.O. Box 48, Ipswich IP4 2AB. Tel: 0473 50304.*



Minor Miracles WS2000 modem ... link to the rest of the world

Applesoft compiler

THE Einstein Compiler is a full-feature Applesoft compiler which translates Applesoft programs into Apple machine language.

It also enables the integrated compilation and execution of multi-module programs, provides extensive debugging tools, and offers mechanisms for protecting program security.

The compiler supports all Applesoft and DOS functions and commands, including hi-res lo-res graphics, shape tables and defined functions.

A code compression algorithm keeps the size of the compiled program to a minimum, so saving memory without sacrificing speed.

Source programs up to 31k

can be compiled and larger programs can be reduced in size with a separate utility program on the compiler disc which removes REMs.

The Einstein Compiler runs on an Apple II and costs £89.

● *Pete and Pam Computers, New Hall Hey Road, Rossendale, Lancashire BB4 6JG. Tel: 0706 212321.*

Chess for all styles

SARGON III, the computer chess game from Hayden Software, is said to be the complete package for beginners and experts alike.

With an opening library of over 68,000 moves it instructs the novice in chess fundamentals.

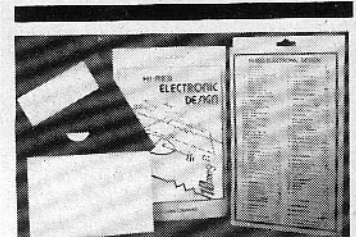
An explanation of the game from the US Chess Federation covers the basics and Sargon III

covers the rest.

It can search many moves ahead at higher levels of play and offers 45 brain-teasing chess problems.

Sargon III's retail price is £33.95.

● *Pete & Pam Computers, New Hall Hey Road, Rossendale, Lancashire BB4 6JG. Tel: 0706 227011.*



A HI-RES utility for creating electronic schematics on an Apple is "Hi-Res Electronic Design".

The program allows the user to create fast electronic schematics and save them as picture files on disc as well as dump them to a printer. Diagrams may be labelled before saving.

Applesoft graphics are easy, with only a single key stroke required to select from 98 different electronic components and for choosing shape rotation.

Authors Avant-Garde Creations claim the picture creation commands are so simple that a child could learn them in five minutes.

Components may be moved about the screen by game paddles or joystick.

The package is complemented by Korsmeyer Electronic Design, which gives a menu-driven array of electronic formulas for speedy calculations.

The program costs £19.95.

● *Pete and Pam Computers, New Hall Hey Road, Rawtenstall, Rossendale, Lancashire BB4 6JG. Tel: 0706 212321.*

The robot house has arrived

AN Apple can be turned into a voice-controlled robot with everything except hands and feet, according to Artra, the American manufacturers of Waldo.

Artra says that Waldo can create a fully-automated house where you don't need a door key, and where heating, bath temperature, cooking and even choice of television program can be programmed.

The basic Waldo interface card consists of a plug-in board which holds a voice recognition circuit, the home control hard-

ware, a clock/calendar with battery back-up, two programmable sound generators each capable of generating three different sound channels, and a stereo amplifier.

A microphone and ultrasonic home-controller driver (both supplied) and stereo speakers connect to the card through a single, removable plug.

Waldo's voice recognition library can cope with 24 different phrases, says Artra. But it has to be trained to recognise a specific voice signature.

The basic Waldo can control 16 domestic appliances and with optional controller devices can handle up to 256 lights and appliances.

Keys to the whole operation are seven software programs including Housemaster, which controls household devices via voice, date-time or keyboard, test and utility programs.

Minimum system requirements are a 48k Apple and a disc drive.

● *Sun Computing Services, Concorde House, St. Anthonys Way, Feltham, Middlesex. Tel: 01-890 1440.*

