





How  
Apples  
boost the  
learning curve

Lisa in action  
— with the NHS

An Esperanto for micros

Apple's growing role in industry

Vol. 3 No. 4 October 1983

*Managing Editor*  
**Derek Meakin**

*Art Editor*  
**Peter Glover**

*News Editor*  
**David Creasey**

*Technical Editors*  
**Cliff McKnight**  
**Max Parrott**

*Advertisement Manager*  
**John Riding**

*Advertising Sales*  
**John Snowden**  
**Peter Nowell**

*Marketing Manager*  
**Linda Dobson**

Tel: 061-456 8383 (Editorial)  
061-456 8500 (Advertising)  
Telex: 667664 SHARET G

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

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)

Trade distribution in UK and Ireland by  
Wells, Gardner, Darton & Co Ltd, Fay-  
gate, Horsham, West Sussex RH12 4SU.  
Tel: Faygate 444.

Writing for Windfall: Articles and pro-  
grams 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  
otherwise agreed, material is accepted on  
an all rights basis.

© 1983 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.

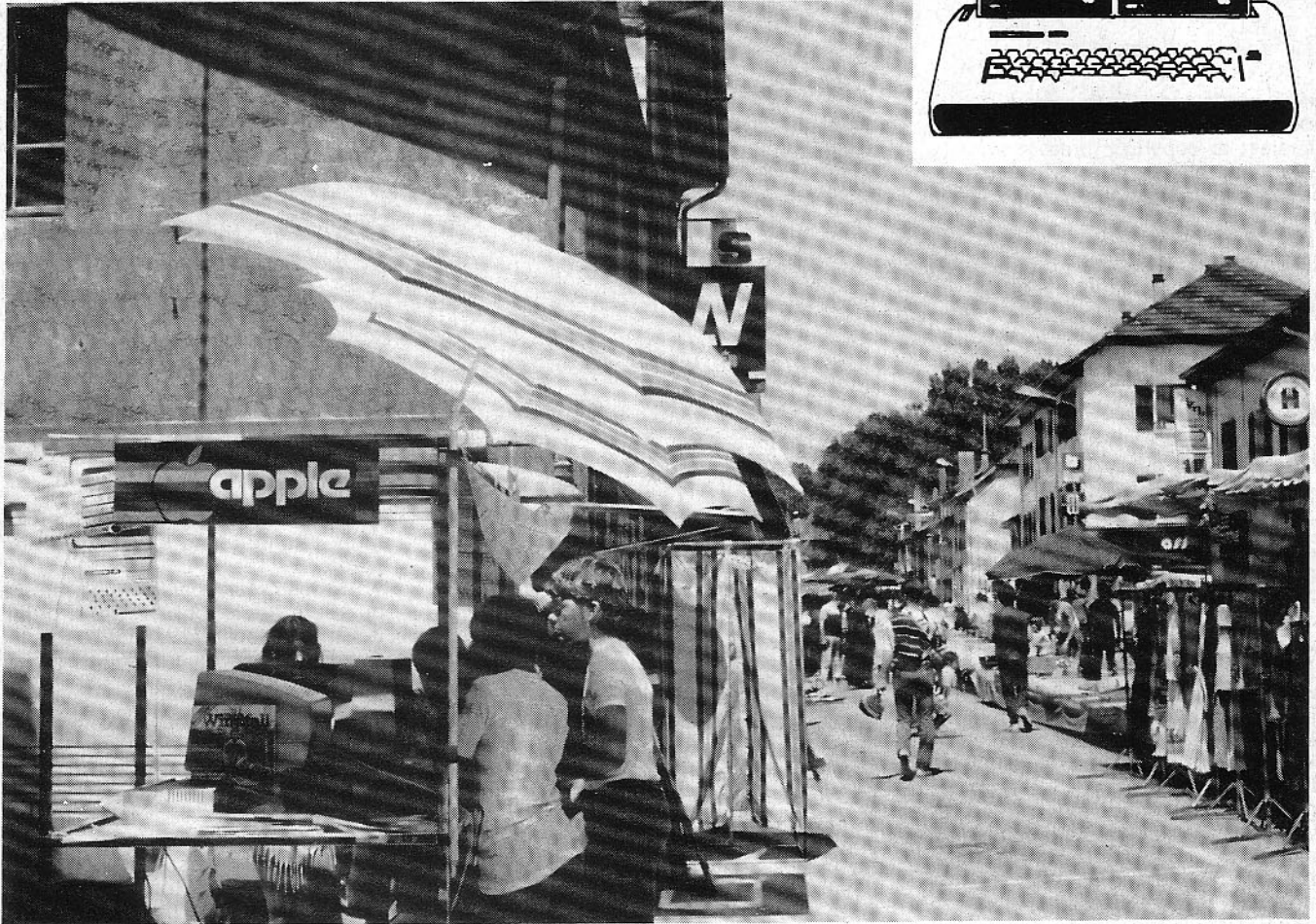
Apple and the Apple symbol are the registered  
trade marks of Apple Computer Inc. Windfall is  
an independent publication and Apple Com-  
puter is not responsible for any of the articles in  
this magazine, nor for any of the opinions  
expressed.

# LISTING

17	<b>WHAT'S NEWS ...</b> Update on the Apple world
23	<b>GAMESMANSHIP</b> Odyssey on the grand scale
28	<b>APPLETIPS</b> Take some toil out of programming
31	<b>THINK TANK</b> Forum for programmers
34	<b>EMULATION</b> Apple's place in industrial control
38	<b>LISA IN ACTION</b> Tackling NHS planning
43	<b>BASICODE</b> All things to all machines
44	<b>DIGISOLVE</b> Professional touch to graphics
48	<b>VISICALC</b> Picking horses for courses
50	<b>FORTH</b> Powerful version from Metacraft
54	<b>TRAINING</b> Six page feature starts here
62	<b>PACKAGE DEAL</b> Thinking man's Space Invaders
64	<b>ABG</b> Pascal-based business graphics
68	<b>FEEDBACK</b> Two pages to air your views
77	<b>COMPUCOPIA</b> The latest in software/hardware
83	<b>AID FOR DISABLED</b> The door to independence



By David Creasey



# In France Apples really get into the market

AMONG the myriad bottles of Beaujolais, Cotes du Rhone and so forth, and with the smells of some of France's most vicious cheeses in the air, yet another Apple dealer entered the market – literally!

The Centre d'Informatique Personnelle (CIP), in Ferney-Voltaire, France, had a unique idea for presenting itself to the public.

No recourse to scantily clad females draped over Apple IIs, trying desperately to promote the idea that Apple IIs, with controller, really do enhance your sex appeal.

Instead, a handful of Windfall T-shirted "Microkids" were engaged to assist, for a day, on CIP's stall at Ferney's weekly open-air market.

On the stall were two demonstration Apple IIs, employed to play anything from Pinball to Beethoven's Fifth. There was also a competition which could be entered either via one of the IIs or, more conventionally, on a sheet of paper.

Everyone who entered was presented with their own Apple to take away – the

*fruit version of course!*

*The Dutch again...*

# Injunction awarded against Franklin

APPLE has scored a major success in its copyright battle against Franklin Computer, manufacturers of the Apple look-alike series of Franklin Ace computers.

It has been awarded a preliminary injunction against Franklin which, it claims, had copied Apple programs in the form of disc and ROM components.

The injunction, awarded by the US Court of Appeals in Philadelphia, reverses a lower court ruling a year ago in favour of Franklin Franklin.

*Franklin Franklin.*



Apple II, particularly those stored in ROM components and the operating system, are subject to copyright protection.

"We think this is a victory for the computer software industry as a whole as well as for Apple," said Albert Eisenstat of Apple in Cupertino.

As a result of the ruling, Franklin will not be able to market their machines at least until the case is resolved at a later hearing.

## Wizard with a window

THERE are unlikely to be any legal wrangles over the latest software offering on the market, Windo-Wizardry, which has been described by some reviewers in the US as a Lisa lookalike.

Windo-Wizardry is a high-resolution display manager for the Apple II which features Lisa-style window graphics.

It is unveiled in the third scenario of Sir-Tech Software's Wizardry fantasy role-playing program, Legacy of Llylgamyn.

"Instead of treating the TV screen as a flat matrix of dots, Windo-Wizardry allows our programs to create and manipulate rectangular windows — areas of any size inside which text and graphics can be placed", said Robert Sirotek, of Sir-Tech.

Each window has several attributes, including the window's size, position and priority. When windows want to use the same area of the screen the window with the higher priority is visible, with the lower priority windows hidden behind it.

When windows are moved, created, removed or changed in priority the hi-res display is automatically regenerated.

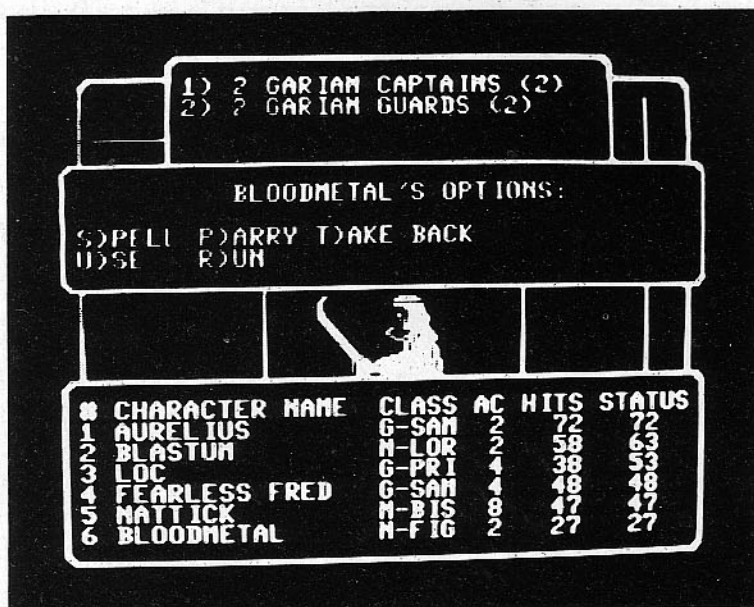
Sir-Tech says the concept fulfills a need to sidestep clutter on the screen and yield a cleaner image with functionality for the consumer.

"Windo-Wizardry propels you into the 3-D world of Wizardry by allowing full screen maze displays with superimposed display windows which help the player become more engrossed in the game," says Sir-Tech.

## New dealer financing

APPLE'S dealer network stands to benefit from two new finance schemes launched by the company last month in conjunction with Borg-Warner Acceptance.

The schemes are designed to help dealers increase their levels of business without undergoing cash-flow crises —



Windo-Wizardry features Lisa-like graphics

one of the prime causes of new business failure.

Applestock, a stock financing programme, is a facility for dealers who apply to Borg-Warner for an extra credit line separate from that provided by Apple itself.

When accepted, dealers also have the option of using Borg-Warner to finance an entire stock purchase from Apple.

If this option is taken the repayments are made through direct debit for a three month period with minimal interest. Stock valued at £3,000, for example, would be repayed as a total of £3,037.50.

The second scheme, a receivables financing programme, is designed to secure dealers against the problems involved in dealing with large companies, who traditionally take longer to pay invoices — especially on substantial amounts.

A dealer can effectively "sell" an outstanding invoice to Borg-Warner for 80 per cent of its face value, thus covering the major proportion of his debt to Apple.

Unlike a factoring scheme, the dealer still collects the invoice payment from the debtor — but he can afford to be less persistent about payment and therefore avoid damaging client relations.

## Fun run

AN Apple IIe took part in Britain's toughest fun run — although it wasn't actually running. It was operated in conjunction with a specially designed software program to ease administrative headaches at the marathon around Canterbury last month.

Apple dealers Emetco wrote the marathon computer program. "With information on more than 1,000 runners we were able to amend details quickly and produce result times and placings immediately after a runner completed the course", said Sheila Buss, of Emetco.

The Apple also supplied lists of competitors, divided into a number of

categories, including sex, finishing time... and where they stayed the night before the race.

With so many cars coming into Canterbury, the Apple took even this problem in its stride, allocating spaces to the entrants to ease the chronic parking problem.

## World of tomorrow

BRITISH businessmen were given a chance to step into "the world of tomorrow" at the first Apple Village at Ashford in Kent last month.

It was a show within a show — the latter being the Kent Industry and Commerce Exhibition — and formed the launching pad for a nationwide computer roadshow tour sponsored by *Windfall*.

"We wanted people to see the real advantages in these marvellous tools and to demonstrate that people need be no more conversant with the actual workings of an Apple than they are with their pocket calculator", said Derek Meakin,

## Prodos goes

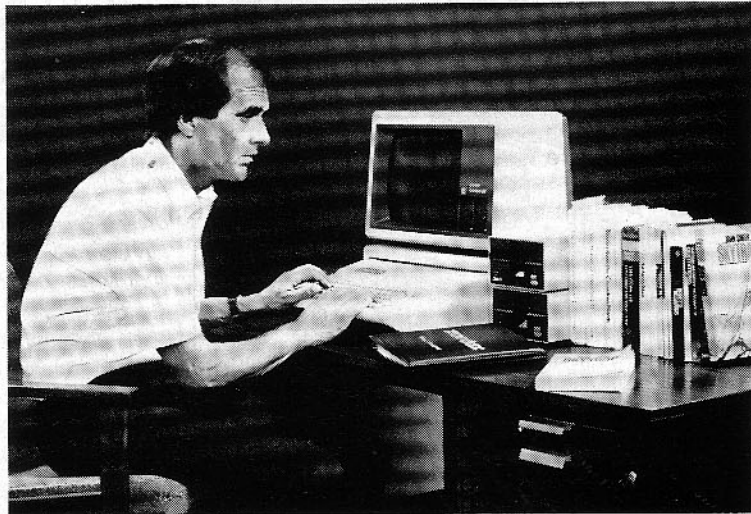
A DISC operating system for the Apple II family is being licensed by Apple Computer to software developers for applications development.

Prodos is said to provide increased compatibility between Apple II and Apple III environments and the higher performance required for more sophisticated Apple II applications.

It uses the hierarchical file structure, file-naming conventions and data formats of the Apple III Sophisticated Operating System (SOS). Prodos data files and data media are interchangeable between the Apple II and Apple III.

Using Prodos, the Apple II can handle larger files such as those often required by





*John Smith ...  
Apple cured  
rewrite blues*

managing editor of *Windfall*.

The show gave dealers and suppliers of hardware and software a chance to show their products to a previously untapped and non-Apple specific market.

"The expert staff at the village were picked for their ability to speak English, not computerese," said Meakin. "We wanted to fascinate the potential first-time buyer, not confuse him."

## Technical aid today

ALSO on the road, but with a slightly different purpose, is the Concerned Technology Travelling Exhibition.

It will be visiting 29 major centres throughout the UK over the next 10 months to demonstrate the latest technological aids, both microprocessor based and electronic, for the disabled.

The first of the three day exhibitions is in Hastings from October 9 to 11, and the show moves from there to Southampton, Exeter and Plymouth.

It will be in London on November 12-14 and will reach Birmingham next February, Swansea and Belfast next May and various Scottish cities in June and July.

Entrance to the shows, which are funded by the Department of Trade and Industry, is free. A variety of aids will be demonstrated and can be tested by able bodied and disabled alike.

"We want to encourage and develop the aids and microchip equipment that will take us well into the next century and be a benefit to mankind", said Peter Mahon, one of the organisers.

"Trained staff will be on hand to discuss any problems and thoughts as to the workings of the equipment. Each

venue has catered specifically for the disabled person.

"At the end of the day we hope that the exhibition will have helped people lead a fuller and more independent life".

## The wise investment

"AN investment that paid for itself 20 times over in the first year" ... that is how new thriller writer John Smith describes his Apple.

A professional pilot since 1960, Smith turned to freelance aviation journalism in 1979 and subsequently decided to write a novel "as a hobby".

That was *Skytrap*, released in the UK in August. It was initially "pecked out" on a typewriter - and the amount of work involved brought the author close to abandoning the project on several occasions.

"The main problem was one of wasted time", he said. "The novelty of re-typing a page of 300 words just to restructure one short sentence wears thin when you wind in the 10th sheet of A4 paper and realise you are still on the first page.

"Multiply that by 350 pages and a couple of complete rewrites and you find

yourself slowly but surely wilting.

"Then last year I walked into Guernsey Computers and asked about a word processor. Within 30 minutes they had introduced me to the 21st century!

"That evening I started to put *Skytrap* onto discs. Two weeks later the four newly-edited discs were printed and a few weeks later I began my second novel.

"Within five months it was finished - the result of an Apple II and a Zardax word processor package".

Smith said the significance of those five months with an Apple was staggering when compared with the two and a half years it took to complete his first novel.

"Roughly speaking it equates to a saving of 700 man hours or a time factor of 80 per cent.

"And while my first book consumed about 8,000 sheets of paper - plus endless typewriter ribbons and a small ocean of corrector fluid - before it was right, the second novel used only 350 sheets of A4 when I ran out the first copy".

His two novels, *Skytrap* and *Patterson's Volunteers*, were bought by London publishers Century, with the paperback rights for both books going to Corgi in the UK and W.W. Norton in the United States.

## New training centres

THREE Apple training centres have been established in Manchester, Birmingham and London. They are equipped by Apple UK, but staffed and run by Keyboard Training as a commercial proposition.

These are the first of what Apple hopes will become a national training centre programme.

Two courses will be offered for first time users - a two hour personal computer familiarisation session and a full-day applications course.

As part of a special promotion Apple is offering a £25 discount voucher towards the cost of the courses, as well as a £100 training voucher to buyers of new Apple systems.

## out to software developers

word processing and data base applications, and can recognise any storage device, either floppy or hard disc, that uses Apple's protocols.

Prodos-based applications will never need to be rewritten to recognise new storage devices for the Apple II.

In addition, the hierarchical, Unix-like file structure of Prodos provides the Apple II user with an organised method for managing larger numbers of files on larger storage devices.

Natalie Shuttleworth, operating systems program manager for the Apple II and Apple III in the USA, said: "So far, we have seeded almost 100 developers with Prodos in the US. All have received it

favourably.

"Based on this acceptance, we are committed to the development of Prodos-based applications and are encouraging developers to use Prodos in favour of the DOS 3.3 applications environment".

Steven Holmes, product manager for Apple II in the UK, said seeding is now beginning in Britain.

Prodos supports advanced interrupt-driven processing, which is required for technically advanced applications such as networking and data communications.

Because it shares common data formats with SOS, Prodos supports mixed network environments such as Apple's forthcoming local area network, Applenet.



## Take an Odyssey on the grand scale

ONE of the first non-arcade games I played on the Apple was called Odyssey. The screen showed a map and you navigated a little man around until you had gained enough companions, possessions and "character points" to go on the quest.

Ultima II is similar in many ways, except it is on a *much* bigger scale and is much harder.

The first thing you have to do is read the manual. It's not a bad idea with most games, but in this case it is absolutely essential.

Firstly, until you've read the manual you may have problems working out how to start playing. Secondly, unless you copy side two (the player disc) and only use your copy, you're not going to play more than one game since loss of life is accompanied by loss of disc.

But the best reason for reading the manual is that it is very funny and gives away a few clues.

Once you've read the manual, admired the free tea towel (which serves as a map) and copied side two, you are ready to start creating a character.

You have a choice of race (human, elf, dwarf or hobbit), profession (fighter, cleric, wizard or thief) and sex (only male or female - what happened to androgeny?).

You then get the chance to allocate 90 character points among the six attributes which make up your character: strength, agility, stamina, charisma, wisdom and intelligence.

If you've made it this far you can name your character and you are then ready to play. You begin life in an unknown place with a little gold, no armour or weapons, and not much of a clue about what on earth you're meant to be doing.

You may be lucky to stumble across a town or village where you can buy a variety of useful or attractive possessions. I stocked up on armour and weapons the second time out, only to find that my

character Zappo (well, it's better than Qwerty, which is what Cliff calls all his characters) wasn't strong or agile enough to use them.

I also bought all sorts of spells since Zappo was a cleric and could use some magic, but found I needed a "certain item" in order to cast them.

The key to Ultima II seems to be in obtaining the appropriate "certain items" so that you can do all the wonderful things which the manual promises. Unfortunately, the main way to gain money, items and experience is to go out and fight.

Finding someone to fight with is the easy part. My first character (Lippi, would you believe?) was killed off very quickly while wandering about in a daze.

There are nine varieties of natives and unless you are in an inhabited region they are all extremely violent.

You can either fight or run, but running away invariably leads you to another. Anyway, they follow you so you've got a bunch to deal with when you're cornered.

Fighting is probably the most tedious

aspect of Ultima II, and unfortunately the most common. In order to fight, you have to wait for the cursor to signal your turn (the natives seem to get theirs randomly), press A for attack and then a direction key.

You get the knack after a while but it is tricky, tiring and noisy. You can rest your weary fingers by getting a status update, but the buzz-ping noises which accompany the fighting can't be toggled off.

This is irritating if you're playing - downright maddening if you're trying to do something while someone else plays.

The map you are wandering around is a stylized map of the world. In certain locations a time door occasionally appears and whisks you off to one of the other four time periods.

The tea towel I mentioned earlier is actually a map of where (but not when) the time doors will appear and where they will take you. It took me a few lives to work out what was going on, and I still can't guarantee to get where I want to be.

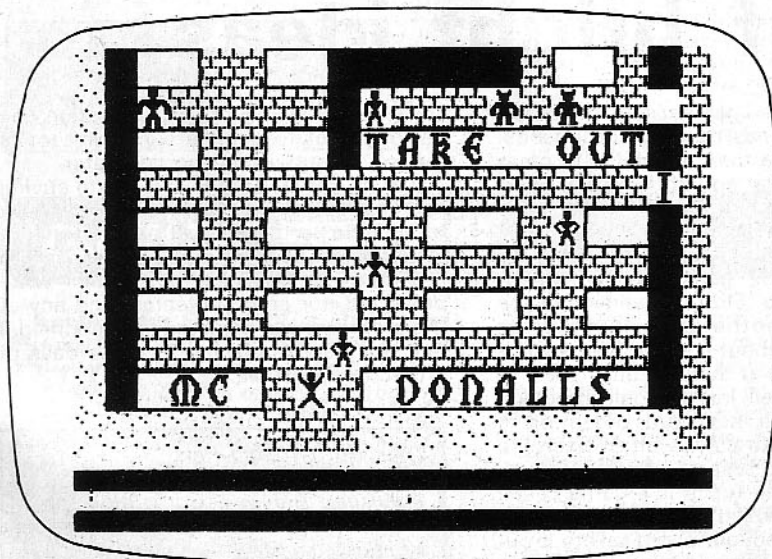
The game can be saved, but it's only a way of suspending play, not of getting back to an advantageous position. Your player disc is constantly updated, and getting killed has a similar terminal effect on the disc.

In order to play again, you have to go through the whole routine of copying the player master, creating a character and so forth. I found this particularly annoying at the beginning when getting killed was a frequent event.

After a while, I realised that if I saved the game and then copied the player disc, I could go back to my backup copy and thus get a second chance to develop an interesting situation. This way, I've managed to get my faithful wizard Wierdo quite far on.

Ultima II is the only game I know which uses all 26 letters of the alphabet as controls, plus a few non-alpha characters for good measure.

However, some of the commands seem to be included purely to fill in gaps in the alphabet. For example, J lets you jump up



How to  
replenish your  
food supply  
while playing  
Ultima II



# GAMESMANSHIP

and down in frustration, and Y for yell lets you type in any kind of invective which is then ignored. Some of the spelling is a bit tortuous too, like Z for ztatus and K for klimb!

There are castles and dungeons scattered about in various locations, and if you have some form of illumination you can enter them at will. Once in, you travel in a 3D maze (like Wizardry) and work your way up (in castles) or down (in dungeons), fighting the ubiquitous nasties and finding treasures or clues – at least that's what the manual says.

Wierdo has entered castles loaded to the hilt with spells and massive hit points, only to be killed off without finding anything out of the ordinary.

Once you've got a strong character and solved enough puzzles to know what your quest is, you can venture into outer space and explore the planets.

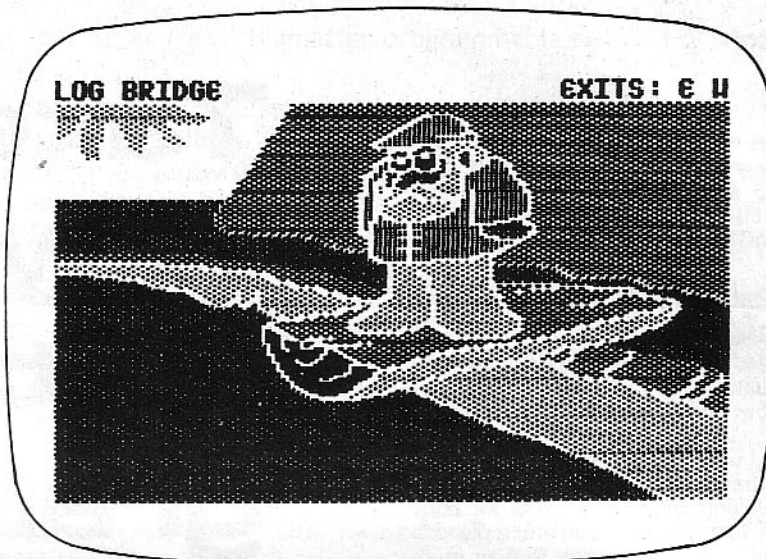
I honestly can't say more than that because Wierdo isn't up to biplane standard yet and looks unlikely to "boldly go" before they start selling day trips to the moon.

Ultima II is a big game and in that respect represents some kind of value for money. The combination of puzzles and arcade-type action makes a change from the normal adventure game.

If you don't mind slaughter on a grand scale and you have good sound-proofing (or wear your personal stereo while you play), you'll get a lot of fun out of the game. I certainly got addicted to it, even though I don't like arcade games as a rule, as a rule.

**Denise McKnight**

*Title: Ultima II  
Author: Lord British  
Publisher: Sierra On-Line  
Requirements: Apple II.*



*A suspicious-looking Little John in Sherwood Forest*

## ***Down in the forest a bridegroom stirred***

IT'S funny how all your friends seem to get married at the same time, isn't it? When my friends were all doing it, I was called on to be best man twice and hated every minute of it. In Sherwood Forest, your duties are approximately those of best man, but playing this game is far more enjoyable.

It seems Robin Hood took a blow on the head while fighting the Sheriff of Nottingham and has forgotten just about everything, including the fact that he is supposed to be marrying Maid Marion today. It is your job to get him to the church on time, and to this end you must solve the various puzzles in the game.

It is not too difficult to find Maid Marion as you wander around, but unfortunately she doesn't seem to recognise

Robin. Funnily enough, the merry men (including Little John) don't seem to recognise Robin either – the blow on the head must have rearranged his features as well as his memory! You must, therefore, help restore him to his former glory, and plastic surgery isn't an available option.

The game is similar in format to many adventure games – hi-res graphics screen with text underneath. Full descriptions appear on your first visit to a frame but not on subsequent visits, so you must pay attention to the text. If you don't you are likely to miss the clues, as I did at least once.

The graphics are very nice, with some use made of animation. Little John looks at you suspiciously, but his fluttering eyelids could give the wrong impression!

## **A pot full of bright ideas**

A RECENT survey showed that the average family in Britain has 1.8 children, lives in a 2.6 bedroomed house, and has a compendium of games in which half of the ludo counters are missing. That's right, the one in the sideboard cupboard that spills out whenever the door is opened.

Rainbow Computing has produced the Apple equivalent of a compendium of games, with the added advantage that bits don't get lost. Pot O' Gold Plus is a double-sided disc containing no less than 46 games using text and lo-res graphics.

There is such a wide range that there is something for everyone... which is exactly what Jim Day (the compiler of the package) was aiming for. Additionally, if your machine is equipped with an Echo II speech synthesizer, Pot O' Gold Plus will talk to you.

Some of the games are extremely simple, and at least one (Bouncing Ball) requires no more than the ability to press a key. At the other end of the scale, offerings like Shooting Stars, Awari, and Dragon Maze require more than a little skill.

The concept of "game" is used widely, too. For example, Eliza is a version of the famous psychotherapist simulation program – talk about your problems without the huge bill at the end; and Tower of Hanoi is the well known logic problem. There is even a keyboard organ or, if you're too tired to play, a bit of Bach "to soothe a savage breast."

One really nice touch is a game called Connect Five, which is quite difficult to lose. Jim Day thought about taking it out of the package because the program plays

so poorly. However on reflection he decided "not to deprive some tiny tot of the joy of outwitting the computer."

I think Pot O' Gold Plus is worth buying just as a games package for all the family. In addition to this it is a treasure trove of ideas for the budding game writer. As Jim has told us, the disc is neither write protected nor copy protected, and any of the programs can be listed and modified. I find this very refreshing in these days of protection paranoia.

*Title: Pot O' Gold Plus  
Author: Jim Day  
Publisher: Rainbow Computing Inc  
Requirements: Apple II. Paddles or joystick*



The colour is quite good, and you may find the game slightly easier at one point if you have a colour set, but it is not essential.

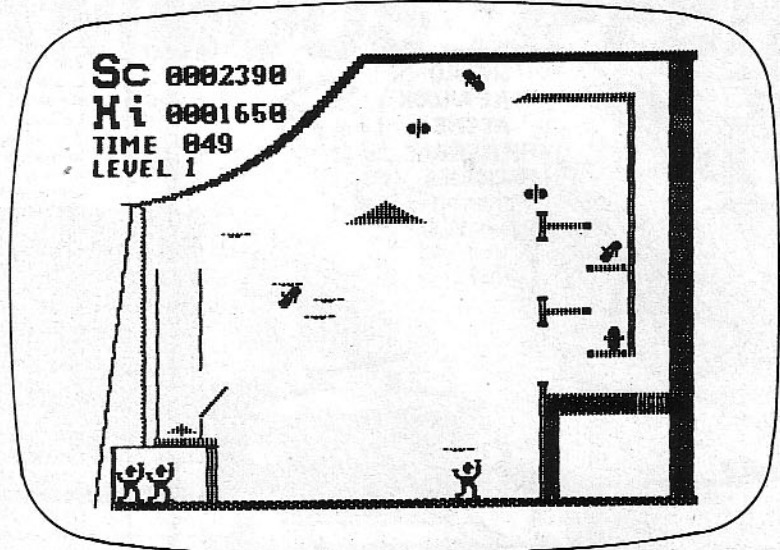
One thing which Phoenix Software have done is to solve the problem of people getting stuck. They will not give hints over the phone (*even if you can afford to phone Illinois*), but they will send a hint sheet on request. The nice thing about the hint sheet is that it doesn't give away anything that you don't want to know. The answers to key questions are written in code, with several levels of hint for many of the questions. Hence, if you are stuck at a certain point and want a hint, you can decode only that particular answer in order to get you moving again.

Another feature which I really appreciated is the save facility. While all adventure games have one Phoenix recognises that you may have two disc drives and therefore gives you the option of having a saved game disc in drive 2. Since you will want to save the game many times, this means that you don't have to keep swapping from one disc to another.

Of course, the reason you need to save the game frequently is that you are quite likely to be killed off in a multitude of ways. One comforting feature of the game is that death is accompanied by its own graphics screen instead of the usual line of text. When I go, I like to go in style.

Phoenix also recognises that players have different degrees of experience in solving adventure games, and have therefore adopted a grading scheme to indicate the level of play required. Sherwood Forest is a Class 3 game for the average player who has completed two or three adventures, with Class 1 being novice level and Class 5 belonging to the experts.

Sherwood Forest is described in the publicity as "Sof-toon 1". I'm not entirely sure what a sof-toon is (software cartoon?), but I look forward to future issues. If they are as much fun as the first one, I'll be quite happy.



Defying the laws of gravity in Juggler

ONE of the many as yet unfulfilled ambitions in my life is to learn to juggle. Honest! If you'd seen the number of flattened oranges in our fruit bowl you'd believe me. It was with trembling hands, therefore, that I booted Juggler from IDS!

Alas, it was not to be. My ambition still remains but I've had some good fun playing the game. As the juggler, your task is to toss various objects into the air in order to prevent other objects reaching the ground.

You move horizontally, trying to keep all six objects from hitting the ground. If you can last for a minute, you move to the next level. If an object hits the ground, the action stops and the next juggler is brought on. There are three jugglers, so you have three "lives".

There are seven levels of play, and the score is 30 times the level per hit. If you stop an object directly with the juggler you get a bonus of 500 points. As the level increases so the size of the juggler's missiles decreases. You start with something resembling flying saucers but they are gradually reduced to flying tea-leaves!

The game also has an auto-toss feature which will produce a constant stream of emanations from the juggler's hand — "nothing up my sleeve". The game then becomes a matter of horizontal positioning of the juggler. This may sound too easy but effectively it means that quite young children can manage to play.

In fact, I think this is a game that will appeal mostly to children. Like Pie Man,

## Game for three hands

which I reviewed a few months ago, Juggler is a non-violent game. This may well be a selling point for some parents who dislike the idea of perpetual slaughter on their monitors. After all, you can see it on the news every day without seeing it emanating from your micro.

The game has all the usual features like sound effects, which can be toggled off, and a pause facility via ESC. It can be played on keyboard or paddle/joystick although keyboard play is a little difficult at first. It also features some interesting variations on the laws of gravity, but then it doesn't pretend to be a physics tutorial.

All in all, I think Juggler is a nice game for the younger members of the family. The combination of the auto-toss option and the seven levels means that it can be enjoyed by both the very young and the more adroit elders of the primary school.

Meanwhile, I still haven't learned how to juggle so it's back to the fruit bowl for me. "Orange squash" has a whole new meaning in our house.

*Title: Sherwood Forest*  
*Authors: Dav Holle and Dale Johnson*  
*Publisher: Phoenix Software*  
*Requirements: Apple II*

*Title: Juggler*  
*Author: None credited*  
*Publisher: Innovative Design Software*  
*Requirements: Apple II*

## QUICK SPINS

**Apple Cider Spider:** *This little spider mistimed its return to his web in the rafters after the nightly excursion round the Cider Mill. Can he make it up from the bottling racks, avoiding apple crushers, conveyor belts and hungry birds among many other hazards. His only way through and up is via the drop lines he left on his way down. So get jumping.* (Sierra Online).

**The Quest:** *As the king's newest*

*adviser, you must accompany his champion, Gorn, on a mission to rid the kingdom of Balema from a terrible dragon. Over 200 locations on a double-sided disc, hi-res graphics and more than one possible solution.* (Penguin Software).

**Plasmania:** *A secret serum shrinks you and a surgically equipped submarine to microscopic size before injecting you into the patient's bloodstream. Your object is to clear a bloodclot near the*

*patient's brain, but to reach it you must fight antibodies and bacteria while avoiding healthy blood cells. A race against time.* (Sirius).

**Theseus and the Minotaur:** *The dreaded Minotaur has captured the princess Ariadne and hidden her somewhere in his labyrinth, a 3-D maze with secret doors, confusing Halls of Mirrors and three floors. Can be played at different levels of difficulty.* (TSR).



# An HGR answer to BLOADing problem

**t** It is possible to switch from text to graphics, pages 1 to 2 etc using a series of POKE commands as outlined in the Applesoft Basic Manual pages 132-134.

However, I recently encountered a problem with this technique which other users may wish to note. My program BLOADed a previous prepared set of axes on the high-resolution graphics and was plotting points on this using shapes from a shape table.

The problem is that using the series of POKE commands to display my BLOADed axes I found that no shapes were being plotted on the screen.

This is due to the fact that high resolution graphics uses a number of

page 0 locations (see Applesoft Basic Manual pages 140-141).

Using the POKE commands does not set up these locations with the correct values for plotting on the screen.

The solution is to have a line in your program (anywhere before the series of POKE commands will do) containing HGR or HGR2. For example the first line of your program could be 10 HGR : TEXT.

It is interesting to note that issuing the commands HGR and TEXT in immediate mode before running the program will suffice, and that even soft booting the system with PR#6 will not affect this "graphics initialisation" once it has been carried out.

**W.H. Thomson**

## Memory map reveals secrets

**t** The routine below calculates the location of a particular byte in the Apple's hi-res memory map.

**W.J. Hill**

```

10 REM LOCATION OF HI-RES BYTES
11 REM X=0 Y=0 IS TOP LEFT CORNER
12 REM ONLY 7 BITS OF BYTE ARE
13 REM DISPLAYED ON THE SCREEN
14 REM SEE REFERENCE MANUAL FOR
15 REM MORE DETAILS
16 REM
17 REM BASED ON INTEGER PGM.
18 REM IN 'MICROCOMPUTING' NOV.
19 REM 90
20 INPUT "HGR1 OR HGR2";P
30 IF P = 1 THEN S = 8192
40 IF P = 2 THEN S = 16384
50 IF P < > 1 AND P < > 2 THEN
60 GOTO 20
60 INPUT "INPUT X ";X
70 IF X > 39 OR X < 0 THEN 60
90 INPUT "INPUT Y ";Y
95 IF Y > 191 OR Y < 0 THEN 90
100 A = INT (Y / 64)
110 Y1 = (Y / 64 - INT (Y / 64))
120 B = INT (Y1 / 8)
130 C = (Y1 / 8 - INT (Y1 / 8)) *
140 L = S + (A * 40) + (B * 128) +
(C * 1024) + X
150 PRINT L
200 INPUT "MORE? ";A#
210 IF LEFT$(A#,1) = "Y" THEN
220 END

```

## Satisfying sounds

**t** Here are two short subroutines which may be useful. Enter the monitor (CALL-151) and key:

\*300,310

0300- A2 00 86 FF A6 FF CA D0

0308- FD AD 30 C0 A5 FF 69 01

0310- 85 FF AD 00 C0 C9 80 30

0318- ER AD 10 C0 60

CALL 768 will produce a very satisfying noise which can be altered by changing the number underlined (location \$30F). The noise stops in a keypress. The routine is relocatable.

This tip follows on from M. Bowyer's offering on page 34 of the September 1983 Windfall.

**Marcus Macrae**

## Sort out the Apples

**t** IF it is necessary for a program to determine whether it is running on an Apple II or IIe, try the following:

X = PEEK (64266)

IF X = 240 it is a IIe

IF X = 208 it is a II plus

ROM locations 64265 to 64272 contains the codes which produce "Apple II" at the top of the screen when the machine is cold started. 64266 to 64269 (pple) are in lower case on the IIe.

**Brian N. Gedge**



## Track down those OUT OF MEMORY errors

**t** Sooner or later everyone sees the OUT OF MEMORY error sign. However there are two ways to run out of memory in Applesoft. The most common is to have a program that is too big or uses too many variables.

The only solution to that problem is to trim down the program, keep the data on a disc, or chain the program in from the disc in segments.

The less common cause is stack overflow. This is easy to spot because after getting the OUT OF MEMORY error, PRINT FRE(0) tells you that there is still free memory.

The problem is that Applesoft uses the 6502 stack to save its recursive subroutine calls and the stack is a limited resource.

Here are some causes:

- Too many FOR-NEXT loops.
- Too many GOSUBs.
- Excessively complex mathematical or string functions.
- GOSUBS with no RETURN.
- Improper recovery in ONERR GOTO routines.
- CALLS or interrupts that don't restore the stack properly.

These effects are cumulative, and you might be affected by more than just one. The first four are inherent in the program structure. If your program is cleanly structured then you probably won't suffer from these.

If you are using ONERR GOTO, Apple UK suggests you carefully read pages 81 and 82 and look at page 136 of the Applesoft Reference manual.

There are two correct ways to leave the ONERR routine. You can use RESUME so that Applesoft can take care of the stack and re-execute the statement that caused the error.

But if you don't use RESUME, you must use the stack recovery routine on pages 82 and 136 (the example on

page 136 is easier to use) before you do a GOTO to any other part of the program.

The routine on page 136 can be replaced with CALL -3288 and no POKES if you can insure that you will be using ROM Applesoft.

When Applesoft executes a CALL, it does a 6502 JSR to the specified address. It's up to you to leave the stack the way you found it.

Likewise any routines to handle interrupts from a peripheral card must remove any bytes pushed on the

stack and maintain the 6502's internal registers.

A very complex program might run into one of the first three problems as a part of normal operation. There is a routine in Applesoft that clears the stack, CALL 54915 will clear the stack without clearing the variables. It wipes out all pending FOR-NEXT loops, GOSUBs, and formulas.

This is one way to implement a program controlled restart. But beware, there is no substitute for a well structured program!

## Power points

**t** We have often been asked about the problems of using a British Apple in the US and vice versa. Here are some hints from Apple UK.

- Use a stepdown transformer to convert from 220V to 110V (or step-up from 110V to 220V if you have a 220V Apple). Buy a 1000 watt transformer from your local hardware store. Cost is about \$15 US and it comes equipped with adaptor plugs to fit most common sockets.

- You must use the monitor or TV that your Apple was designed to work with. Using an NTSC (US) monitor with a European Apple (or vice versa) will not produce satisfactory results.

Check with the manufacturer of the monitor or TV to determine if it will work with both 50 and 60 Hz line frequency.

- Check with the manufacturer of any peripheral that must be plugged into the wall socket (printers, plotters, hard discs) to determine if they will work on both 50 and 60 Hz.

- The Apple power supply will function correctly with voltages between 107V and 132V (214V to 264V on a 220V supply). If the line voltage fluctuates outside of these specifications you will need to use a power conditioner to insure uninterrupted operation of your Apple.

Operation without the conditioner will not result in damage to the Apple hardware. However, the system may "crash".

**t** The following routine, also relocatable, inverts a rectangle of the low resolution graphics screen designated by two pairs of co-ordinator (top left corner, bottom right corner):

\*300,31E

0300- A4 FC A6 FD BA 20 71 F8

0308- 49 FF 20 44 FB BA 20 00

0310- FB E8 E4 FF D0 EE D8 C4

0318- FE D0 E7 50

By invert, I mean replace the colour of each dot by its exclusive OR.

To use from Basic, enter the values as follows:

POKE 252, X (low byte): POKE 253, Y (low byte): POKE 254, X + 1 (high byte): POKE 255 Y + 1 (high byte).

To call the routine type, CALL 768.

Marcus Macrae

## COLOUR SWITCH TRAP

**t** If the colour killer switch on the Ile is in the wrong position it results in a very poor quality picture on a black and white screen.

The switch is located inside the Apple, either in the centre or on the

right hand side of the motherboard.

It is usually in the "colour" position when the machine is supplied, and must be placed towards the keyboard if using a black and white screen.

Steve Holmes





# Reading text file names into Applesoft programs

I HAVE a number of Applesoft programs handling various kinds of scientific and other data which is stored in TEXT files, writes **Max Parrott**. Time and time again it is desirable to let the user see which files are available to him and to select one of them for loading.

The simple way of doing this is to issue the CATALOG command from within the program. The drawbacks are that all files on the disc are listed and the user has to type in the name of the required file in order to load it. I find that most users find this quite difficult and rightly think it stupid to have to type in a name when it is already there on the screen.

Finding that other people were also looking for a utility to overcome these problems, and also not being able to find one in the magazines, has forced me to write my own.

My programs tend to find space at a premium, hence I put the program into the DOS area where the CATALOG command is usually handled. The result is that the issue of this command only lists TEXT files, but I don't find this any problem. If I want to edit the program or the disc I boot up another disc.

There are two ways of utilising the program, either BLOAD it or POKE it from the Basic program or BLOAD it in and INIT a new disc which will carry the Basic program.

The following Basic program both gives the POKEs required to set the routine up and illustrates the use of the utility.

```

LISTING OF PROGRAM TO 'CATALOG
INTO BASIC'
10 DATA 32,220,171,169,255,141,
249,181,32,247,175,32,17,176
,176,76,162,0,142,156,179,18
9,198,180,240,66,48,57,189,2
00,180,41,127,201,0,208,48,1
69,30,160,0,145,155,32,82,22
8,24,173,156,179,105
20 DATA 201,170,169,0,105,180,1
68,169,30,32,226,229,160,1,1
65,111,145,155,200,165,112,1
45,155,200,152,24,101,155,13
3,155,144,2,230,156,32,48,17
8,144,184,176,175,76,127,179
,32,220,3,132,155,133
30 DATA 156,160,0,169,6,145,155
,168,145,155,169,1,136,145,1
55,32,190,222,32,217,247,24,
165,155,105,7,133,155,144,2,
230,156,32,214,3,96
40 FOR I = 44440 TO 44440 + 136:
READ J: POKE I,J: NEXT
50 D$ = CHR$(13) + CHR$(4)
60 DIM A$(30)
70 CALL 44535,A$
80 HOME: FOR I = 0 TO 30: IF A$
(I) = "" THEN GOTO 100

```

```

90 PRINT I;" ":A$(I): NEXT
100 PRINT : PRINT "PRESS NUMBER
REQUIRED"
110 INPUT I
115 ONERR GOTO 200
120 PRINT D$"OPEN":A$(I)
130 PRINT D$"READ":A$(I)
140 GET T$: PRINT
145 PRINT T$:
150 GOTO 140
200 PRINT D$"CLOSE":A$(I)
210 END

```

The program will read a TEXT file from disc and display its contents on the screen. The array into which the TEXT file names will go may have any valid string array name as long as it has been dimensioned.

Sometimes you may not want to read TEXT file names but other files such as Applesoft or binary, etc. This is easily accomplished by putting values other than 0 into location \$ADBA (44474 in decimal). The required values are:

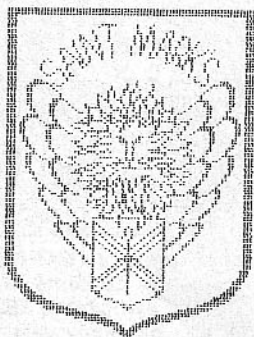
- 0 text
- 1 integer
- 2 Applesoft
- 4 binary
- 8 S-type
- 16 R-type

The assembly listing is as follows:

0800	1	* ROUTINE TO PUT FILES *	ADAA BE 9C B3	23	STX #B39C	
0800	2	* INTO AN ARRAY	ADAD BD C6 B4	24	LDA #B4C6,X	:GET TRACK NO.
006F	3	FRETOP EPZ #6F	ADBO FO 42	25	BEQ EXIT	
007B	4	LOWTR EPZ #9B	ADB2 30 39	26	BMI NEXT	:SKIP IF DELETED
0071	5	FRESPC EPZ #71	ADB4 BD CB B4	27	LDA #B4CB,X	:GET FILE TYPE
03D6	6	DOSCALL EQU #3D6	ADB7 29 7F	28	AND #*7F	:INCLUDE LOCKED
03DC	7	FINDMAN EQU #3DC				
E452	8	GETSPC EQU #E452	ADB9 C9 00	29	CMP #0	:THIS IS FOR TEXT
E5E2	9	MOVSTR EQU #E5E2				
DEBE	10	CHKCOM EQU #DEBE	ADBB D0 30	30	BNE NEXT	
F7D9	11	GETARYPT EQU #F7D9	ADBD A9 1E	31	LDA #30	:NAME LENGTH
AD98	12	ORG #AD98	ADBF A0 00	32	LDY #0	
AD98	13	*****	ADC1 91 9B	33	STA (LOWTR),Y	:LENGTH INTO
AD98 20 DC AB	14	JSR #ABDC				
MANAGER WORKAREA						
AD98 A7 FF	15	LDA #*FF	ADBC D0 30	30		
AD9D BD F9 B5	16	STA #B5F9	ADBD A9 1E	31		
ADA0 20 F7 AF	17	JSR #AFF7	ADBF A0 00	32		
ADA3	18	MAINLP:	ADC1 91 9B	33		
ADA3 20 11 B0	19	JSR #B011				
SECTOR			ADCB 20 52 E4	34	JSR GETSPC	:MAKE ROOM FOR
ADA6 B0 4C	20	BCS EXIT	ADCC 18	35	CLC	
ADAB A2 00	21	LDX #0	ADCD A7 9C B3	36	LDA #B39C	:GET OLD X TO
ADAA	22	INNERLP:	FIND NAME			
			ADCA 69 C9	37	ADC #*C9	:X CARRIES LOW
			ADCC AA	38	TAX	
			BYTE			



ADCD A9 00	39	LDA #0	MANAGER	ADF7	63	START:
ADCF 69 B4	40	ADC #B4		ADFA B4 9B	64	JSR FINDMAN
ADD1 AB	41	TAY	;Y THE HIGH BYTE	ADFC 85 9C	65	STY LOWTR
ADD2 A9 1E	42	LDA #30	;THE LENGTH	ADFE A0 00	66	STA LOWTR+1
ADD4 20 E2 E5	43	JSR MOVSTR		AE00 A9 06	67	LDY #0
ADD7 A0 01	44	LDY #1	;NOW SET UP	AE02 91 9B	68	LDA #6
POINTER				AE04 AB	69	STA (LOWTR),Y
ADD9 A5 6F	45	LDA FRETOP		AE05 91 9B	70	TAY
ADDB 91 9B	46	STA (LOWTR),Y		AE07 A9 01	71	STA (LOWTR),Y
ADDD C8	47	INY		AE09 88	72	LDA #1
ADDE A5 70	48	LDA FRETOP+1		AE0A 91 9B	73	DEY
				AE0C 20 BE DE	74	STA (LOWTR),Y
ADE0 91 9B	49	STA (LOWTR),Y		AE0F 20 D9 F7	75	JSR CHKCOM
ADE2 C8	50	INY			76	JSR GETARYPT
ADE3 98	51	TYA				;FIND THE ARRAY
ADE4 18	52	CLC	;UPDATE LOWTR			
ADE5 65 9B	53	ADC LOWTR		AE12 18	77	CLC
ADE7 85 9B	54	STA LOWTR		AE13 A5 9B	78	LDA LOWTR
ADE9 90 02	55	BCC NEXT		AE15 69 07	79	ADC #7
ADEB E6 9C	56	INC LOWTR+1		AE17 85 9B	80	STA LOWTR
ADED	57	NEXT:		AE19 90 02	81	BCC OKAY
ADED 20 30 B2	58	JSR #B230	;UPDATE X REG	AE1B E6 9C	82	INC LOWTR+1
ADFO 90 B8	59	BCC INNERLP		AE1D	83	OKAY:
ADF2 B0 AF	60	BCS MAINLP		AE1D 20 D6 03	84	JSR DOSCALL
ADF4	61	EXIT:		AE20 60	85	RTS
ADF4 4C 7F B3	62	JMP #B37F	;BACK TO FILE	AE21	86	END



## Bypassing the bit image mode

I FOUND the articles on the Epson printer by Mike Glover and Christopher Roper in the May and June issues of *Windfall* very interesting, and as you can see from the above, the bypass routine proved particularly useful, writes **P. Barlow**.

Rather than attempt to use bit image mode from within a word processing program, I use a simple Basic program which incorporates a routine to place the image in the desired position.

I find it convenient to use DATA state-

ments for the encoded dot pattern, and if this is done in batches of say 10 bytes, editing of the printed image, which is often needed, is easily accomplished.

In encoding, it is important to allow for the dot pitch of 60 dots/inch horizontally and 72 dots/inch vertically to avoid distortion. In the school badge illustrated starting with a master of width 144mm I used 2mm spacing laterally and 1/16in vertically to give the desired size of printed image and acceptable proportions.

Suitable graph paper is made from tracing paper.

For some images, a signature for instance, the distortion is acceptable and a master written on 1mm graph paper can be used.

With 77 x 16 print head positions to encode for the badge, this part of the work can be a chore, but it is achieved surprisingly quickly, and in this particular case, the semi-symmetrical nature of the design eased the work considerably. 🍏

```

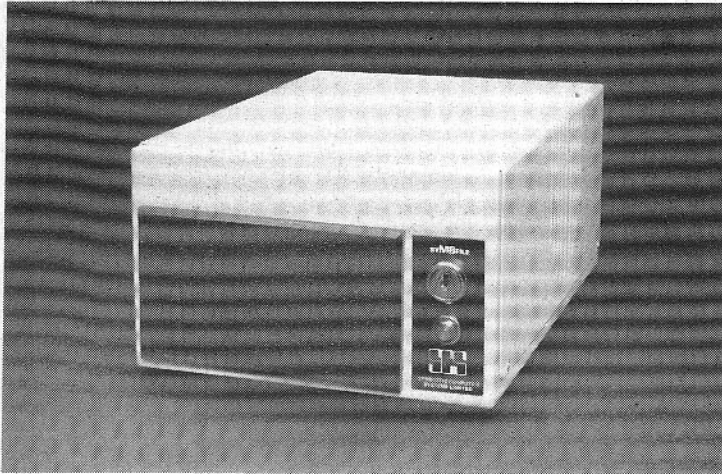
10 REM BIT IMAGE PRINTING
15 L = 77
20 ROW = 16
22 FOR AD = 768 TO 790
24 READ BYTE
26 POKE AD, BYTE
28 NEXT
30 DATA 32,183,0,240,16,32,190,
222,32,248,230,44,193,193,48
,251,142,144,192,16,235,96,0
40 HOME : PRINT TAB(11)"BIT IM
AGE PRINTING"
50 PRINT : PRINT : PRINT "CHECK
PRINTER PLUGGED IN AND SWITC
HED ON"
230 PRINT
235 PRINT "NORMAL DENSITY 60 DOT
5/8 INCH HORIZONTALLY": PRINT
240 INPUT "NUMBER OF DOTS TO STA
RT OF IMAGE?": B
270 H = INT ( (B + L) / 256 )
280 R = (B + L) - (256 * H)
290 CALL 768,27,65,B
300 CALL 768,10
310 FOR T = 1 TO ROW
320 CALL 768,27,75,R,N
325 IF B = 0 THEN 360
330 FOR Z = 1 TO H
340 CALL 768,0
350 NEXT
360 FOR Z = 1 TO L
370 READ PINS
380 CALL 768,PINS
390 NEXT
400 IF T < > ROW THEN CALL 768
,10
410 NEXT
420 CALL 768,27,65,12
430 CALL 768,10,10
490 REM DATA COMES NEXT
10000 END

```



# SYMBIOTIC

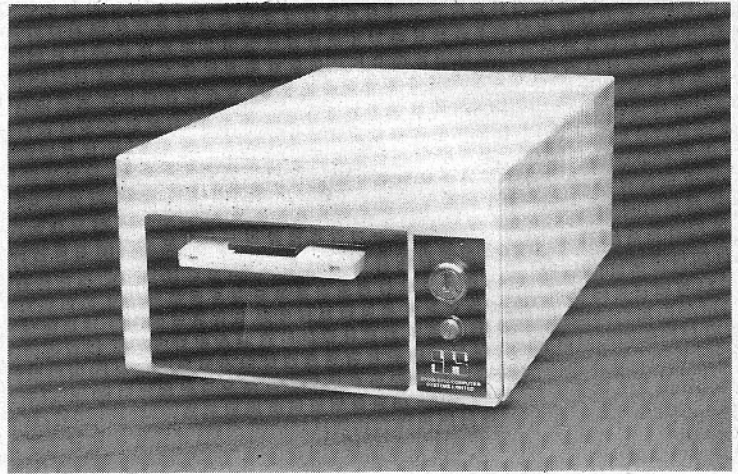
COMPUTER SYSTEMS LIMITED



**SYMBFILE**  
5 1/4" WINCHESTER SUB SYSTEM

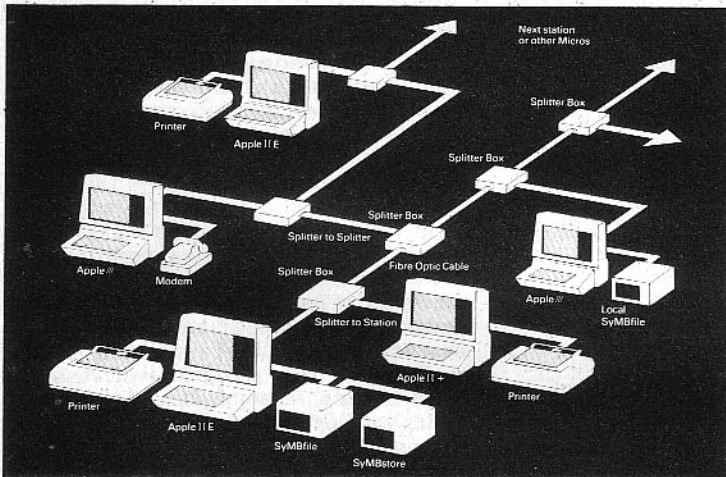
The SYMBFILE hard disk subsystem is a complete add-on mass storage system for the Apple II, II+, IIe, or III microcomputers and is at present being developed for the SIRIUS, IBM PC and the BBC micro. It is compatible with the majority of hardware products currently available for the Apple, including the 16K Language card and 80-column cards. SYMBFILES are available in sizes from 5-21 megabytes.

Full DOS, Pascal, and CP/M support allows any standard application software, including database, word processing, and accounting packages to be used.



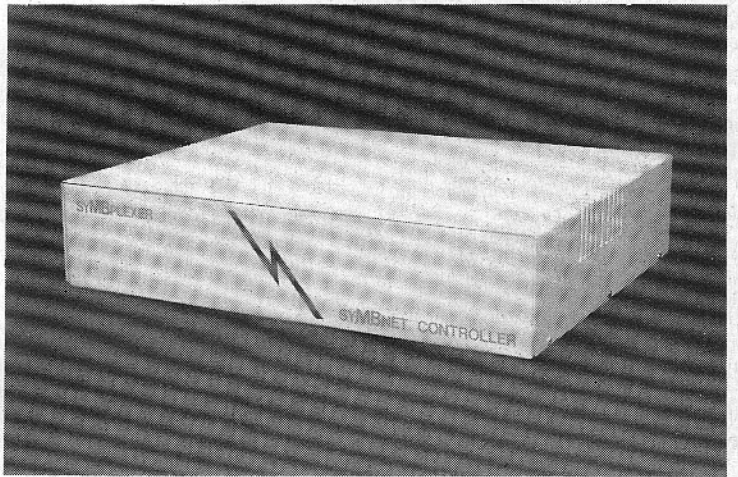
**SYMBSTORE**  
TAPE STREAMING BACK UP

SYMBSTORE is the perfect complement to SYMBFILE giving a high speed, totally reliable back up system. SYMBSTORE will copy the entire contents of a SYMBFILE to a C60 type digital cassette. SYMBSTORE'S unique multiple buffer verification ensures the perfect transfer of data. Software to individually back up volumes and files under present operating systems will soon be available on request.



**SYMBNET**  
LOCAL AREA NETWORK

SYMBNET is a "tree and branch" network system using fibre optic cable to allow several microcomputers to share a common SYMBFILE. SYMBNET is the fastest long range local area network for microcomputers, and can cover a range of 7-9 Kilometres. Fibre optics means that SYMBNET is more cost effective; it uses a high intensity semiconductor laser to transmit data and cables can be laid along the shortest route, whereas other networking systems use flat ribbon or coaxial cable which are sensitive to electrical noise from fluorescent lights, photocopiers, etc. SYMBNET is compatible with DOS, Pascal, SOS, and CP/M running on any microcomputer supported by the SYMBFILE in concurrent operation.



**SYMBPLEXER**  
NETWORK CONTROLLER

The SYMBPLEXER is a network controller which complements SYMBNET. The SYMBPLEXER is connected directly to the SYMBFILE and performs all read-write operations to and from the hard disk. SYMBPLEXER does away with the need for a central machine thus releasing another terminal to run any application you wish, being a dedicated device the read-write operations are performed very efficiently, the support software allows SYMBPLEXER to designate pass codes for each user and to decide which user may access which applications. If you are currently using SYMBNET, accessing your SYMBFILE via a central machine, the addition of a SYMBPLEXER will in no way change the operational capabilities of the network.

*For more details of all SYMBIOTIC products contact*

**SYMBIOTIC**  
COMPUTER SYSTEMS LIMITED

Duroma House, 32, Elmwood Road, Croydon, Surrey CR9 2TX ☎ 01 683 1137 PBX Telex 893815



IN most applications the Apple micro is used as a free-standing data crunching tool. Business applications predominate, with word processing, electronic spreadsheets, database systems and general purpose computation being used by a majority of owners in the areas of both the small business and engineering.

Additionally, in engineering, the Apple can be extended almost without limit to a wide variety of control applications. This is due to the ease with which the basic system can be enhanced by a range of peripheral cards that is unequalled for any other micro, for example, A/D and D/A converters, graphics cards, a full range of I/O cards, and attached processor cards for at least six of the major micro-processors.

With this kind of power, plus excellent documentation and worldwide support, it is not surprising that Apples are found in

many industrial laboratories almost permanently tied up to a wide range of apparatus, from chemical processing plants to nuclear reactors.

However, there is an increasing need in industry for small, dedicated SBCs (Single Board Computers) that can be used in control applications not necessarily requiring a keyboard, full visual display and disc storage.

In other words, these facilities may well be required for the development of the control programs, but are no longer necessary when this development is completed. In such a situation, it is better to regard the Apple as a microprocessor development aid, with the end result being a dedicated SBC.

To this end, let us look at the "conventional" development cycle and discuss each stage in some detail (Figure 1).

#### **Specification and design:**

It will be assumed, for the present, that the 6502 microprocessor has been chosen for the system. In view of its pipelined architecture, powerful addressing modes and the wealth of support literature, this processor is a reasonable choice for all except the most demanding applications.

The overall system is specified in block diagram form, and an assessment made of the number and type of input/output devices required (that is, serial I/O, parallel I/O, counters, clocks and timers, etc). For a 6502-based system, the 6522 versatile interface adapter (VIA) would be a natural choice for many applications.

Also at this stage some estimation would be made of the RAM/eprom memory space required by the final system, so that a suitable SBC may be chosen. The system program task would then be broken down into distinct modules and detailed flowcharts would be prepared.

#### **Code preparation and translation:**

At this stage the designer would actually start writing the system software, and a decision would have to be made as to whether the source code would be written in high level language (such as Basic or Pascal) or assembly language. Although high-level language is often preferred for applications requiring extensive arithmetical operations, the general inefficiency of its compilation into machine code makes assembly language preferable in most cases.

Therefore the designer would very likely prepare assembly language source code, with extensive comments, using a text editor. The object code would then be produced using one of the many assemblers available for the Apple - Applesoft Toolkit assembler, Lisa, S-C, Merlin, etc.

#### **Software testing:**

This first involves a thorough testing of each individual subroutine or program module, then a gradual integration of these until as much as possible of the final system software is tested.

Rarely does any software work correctly at first attempt, and so it is the power of the available debugging aids on the development system that is all-important at this stage.

The facilities that the Apple system monitor has include examining and changing memory locations, examining and changing registers, moving a range of memory, carry out single-line assembly (not available on Apple II Plus), listing (disassembling) a section of memory, single-step and trace (not available in the

# The in-circuit approach to prototype development on the Apple



autostart ROM) and, of course, running the program at full speed.

Various programs and hardware attachments are available to enhance these basic debugging facilities.

As Figure I shows, the software testing may, at least partially, include some hardware testing. For example, similar input/output devices may be attached to the Apple as are intended for the final (SBC) system. The peripheral apparatus may be attached to these and the software testing thus extended further towards the final system test.

**Hardware construction and testing:**

While the preparation and testing of the software is in progress, some hardware construction and testing can be carried out.

The SBC may be designed from scratch using, say, wirewrap techniques initially then subsequent printed circuit board preparation.

More likely, however, is that a commercially available SBC would be used. A wide choice of these exists at a very reasonable cost, with extensive I/O, configurable memory map, built-in D/A and A/D converters, real-time clock and full expansion facilities, thus satisfying the needs of most designers.

If he wishes to fully test the SBC, he may do so by incorporating test procedures into a system monitor ROM and effectively turn the SBC into a "mini development system".

However, this approach would be an unnecessary duplication of the facilities already available on the development system. For the moment, then, let us assume that we have an SBC which has somehow already been tested, maybe by the manufacturer.

**Hardware/software integration:**

Even if things have progressed smoothly so far, we still have, at best, a program which functions correctly on the Apple, and an SBC onto which we must now transfer the system software.

Even if the I/O devices we attached to the Apple are identical to those on the SBC, there is still more work to be done.

The Apple program in RAM (say, starting at address \$800) must be relocated to function correctly at the address of the SBC's eeprom (say, \$F800). This is a simple change to make using an assembler. However, we must also add more code to the program to suitably initialise the stack pointer. Also, it is very likely that all the I/O device addresses must be changed when transferring the program to the SBC.

Finally, we have to provide the means to actually run the program, and this in-

volves including the vectors (particularly the reset vector) at the top of memory.

With all these changes made, an eeprom is blown and fitted to the SBC, and the system should then work correctly.

On a very simple system, it is possible that the integration process may be as straightforward as this. In most cases, however, we are more likely to have a system that does not function correctly, and so we would have liked the full set of debugging tools from our Apple to be available on the SBC, but only until the system works correctly.

It is at this stage of system integration

that things get very difficult, and we would have derived great benefit from having the facilities of in-circuit emulation, as we shall now see.

In-circuit emulation (ICE) consists of circuitry and software integrated with the "host" development system, which allows almost total control of the "target" (for example a single board computer onto which it is intended that an applications program shall eventually be transferred).

This is achieved via an emulator cable terminating in a 40 pin header plug which

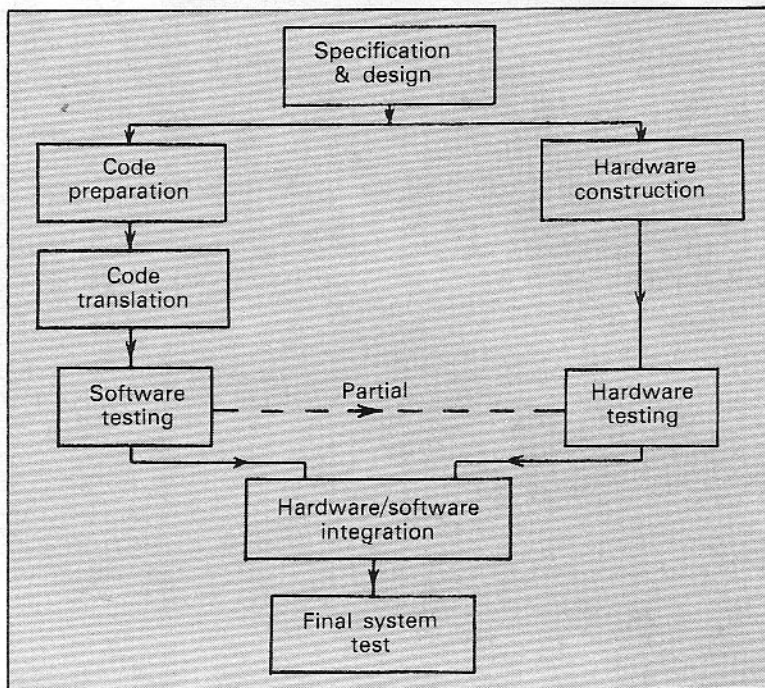


Figure I: Conventional development cycle

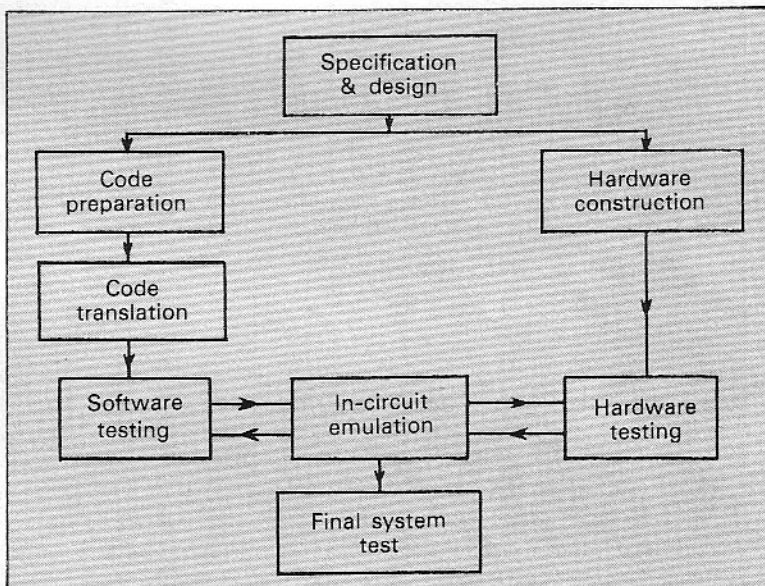


Figure II: Development cycle using in-circuit emulation



# CONTROL

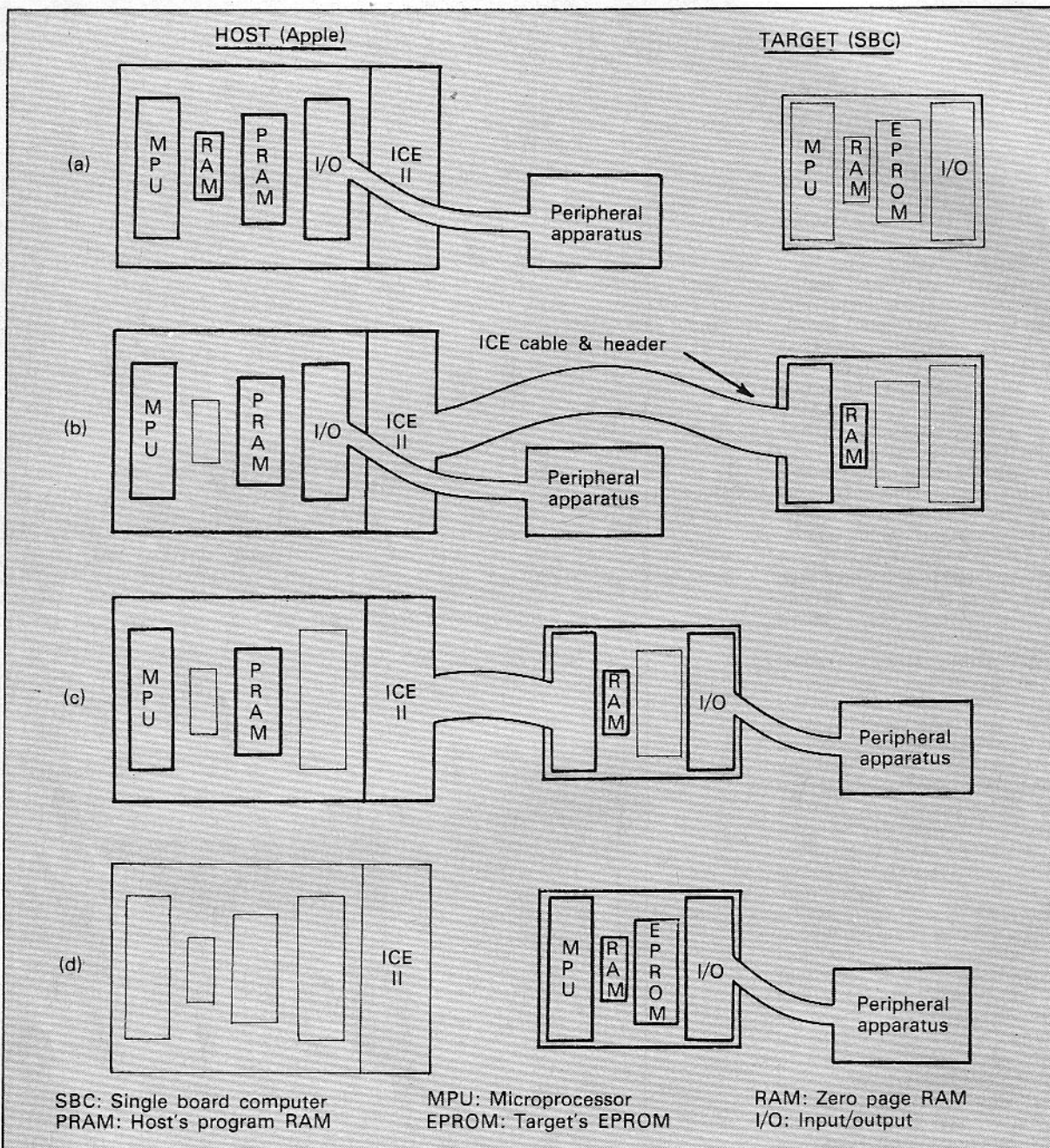


Figure III: Practical application of in-circuit emulation

is inserted into the socket on the target normally occupied by its microprocessor. Thus various sections of the target computer may be tested in turn from the host, thereby avoiding the awkward hardware/software integration phase of development (see Figure II).

The application of ICE is illustrated in Figure III. In Figure IIIa the peripheral apparatus is shown controlled entirely from the host, with the ICE card serving only as a debugging aid for initial program development.

For example, trace, breakpoint and the ability to monitor particular memory locations ("windowing") can greatly reduce the program development time.

In Figure IIIb the ICE header has been

inserted into the target board and its zero page RAM is now being used in preference to that of the host, since the Apple uses zero page extensively for DOS, Applesoft and the machine code monitor. The host's microprocessor and I/O are still being used, and the application program is resident in its RAM.

The next stage would be to start using the target's I/O devices to control the peripheral apparatus, as shown in Figure IIIc.

By now, the system development is almost completed, and it now remains to relocate the host's program so that it will run on the target, add the appropriate vectors and program an eprom.

In some cases, the eprom programmer

itself may have emulation facilities. It may be attached to the eprom socket via a 24 pin header and the program transferred from the host to the programmer's RAM. Only when this program is working correctly do we actually program an eprom.

With the eprom now fitted to the target SBC and the ICE header replaced by a microprocessor, the system should now function exactly as the original version developed on the host, as shown in Figure III d.

● In Part II, the facilities offered by Rovino's ICE II card for the Apple will be examined, and examples given of practical applications of this very powerful technique. ●



AT the beginning of August the Emergency Planning Department of the North Western Regional Health Authority received the first Lisa to be supplied commercially in the UK.

The sheer delight of this superb new equipment compensated for the many hours already spent in developing the maximal potential of the department's previous Apples and now Lisa is ready to take its part in dealing with emergencies, assisting in planning for disasters, including warfare – and in its spare time helping with the many everyday activities of the Health Service.

The department started using micros 18 months ago as part of its planning for emergency headquarters which might at some time involve different personnel working in unfamiliar settings having to recall from records and process new information.

After careful research it was decided to use Apple II Plus computers with Microline 80 printers, dual floppy disc drives and 12in monochrome monitors.

A set of equipment was intended to be used in each control centre if and when required, but otherwise to remain at headquarters in general use for developing the necessary systems.

After initial familiarisation, a graphics tablet was introduced and the printing capability was enhanced by the addition of a cut-sheet NEC Spinwriter. The quantity of data to be processed soon led us to add a Corvus 10Mbyte Winchester hard disc system.

New skills were needed. The whole work pattern of the department needed re-thinking. Immediately, problems arose. We had, in effect, to edit, re-write and tailor to requirements a great deal of our program material. In this, we were not helped by the inadequacy of the manuals.

Early on we prepared training discs to supplement or replace manuals. It is very pleasing to be able to record that our complaints have been heeded – with Lisa the manuals are fully comprehensive and seem excellent.

Material had to be prepared to meet the requirements of varying levels of user. This was tested on volunteers. Deliberate errors were fed into the tests to anticipate the problems that might be met in emergency. Although we started with quite complicated systems, we finished with very easy ones. Our main information database was CP/M based and edited using the Wordstar program.

For absolute beginners, a series of files named "dofirst" were created to provide the elementary instructions on how to work the CP/M database. The next follow on instruction files were named "instrns". These were followed by files named "synonyms" that provided the key code

## Lisa is taking NHS planning in its stride

file names in the form of tables against the alternative common names. Experienced operators could enter the system at whatever level they found appropriate.

Interchangeability of both hardware and software is of great importance in an emergency network. It was our intention to link all our districts to either one or two regional headquarters. Each district would then have the capability of sorting for itself material supplied from the parent computer at control.

The intercommunication to be used would depend upon the facilities available, and provision was made for the eventuality of telephones being destroyed,

By CHARLES F.W.  
FAIRFAX

and even power supplies being lost.

A pedal cycle generator fitted with a car alternator was constructed to be used with an "Applejuice". For this reason, overall power demand was limited to the order of 350 watts, corresponding to an acceptable limit of human pedal power.

A car battery was used to buffer the intermittent input and varying demand.

This also had the advantage of providing an element of EMP protection. Short wave radio communication might need to be used as a last resort.

We soon had two computers talking to one another across the bench using trans-receivers on AF. We have still to achieve direct transmission of discs using higher frequencies. This is very much a matter of available time rather than technical capability.

Until the reason was identified, unexplained minor corruptions occurred in the processing of programs. As work grew, especially using a graphics tablet, it became apparent that static was the source of the difficulties.

Regular anti static treatment of the room and equipment was instituted. In addition, attention was paid to elimination of static sources such as plastic tops and

the improvement of earth discharge pathways from the furnishings.

A rather amusing finding was that there seemed to be sex discrimination in the human induction of static charges. Women wearing synthetic manmade fibre garments seemed to produce more disturbance than men.

Fortunately, the other anti static measures were so successful that it was unnecessary to offer a change of underwear to the ladies before entering the computer room!

The generation of static in the office building led to anxieties that this might occur in the field. However, no static problems were encountered on field tests.

A possible explanation well might be that the increased handling of the equipment was providing earth discharge pathways through the bodies of the handlers.

Although not easy, it would have been possible to match quite a range of alternative makes of equipment. It was soon decided to standardise on that with which we had started. It was however recognised that to increase the RAM by the introduction of expansion cards in order to enhance the 48k memory would be desirable.

Various configurations were tried. The one at present in use employs two Saturn 128k cards, the first in slot 0 and the second in slot 7. The ROM previously on the Apple language card was re-located to the 0 Saturn card, the vacated space on the Appletcard then being re-assigned to additional RAM.

A Microsoft card for CP/M and a Videx 80 column card were used to obtain a fuller potential for the video output and enable us to use Wordstar. The graphics tablet interface card and a printer interface card completed the hardware set-up, filling all the available slots on the main board.

An Apple dot matrix printer was used in place of the Microline 80 to give greater versatility and higher definition in printing both text and graphics. This larger assembly gave rise to overheating problems, which however were very quickly cured by the installation of small internal fans.

The two sets of equipment were not

\*Dr. Fairfax is a consultant in Community Medicine and Emergency Planning Officer to the North Western Regional Health Authority.





*At the touch of a mouse, Lisa helps health workers plan for emergencies*

treated in quite the same way; the non Saturn set was left with the 10Mbyte Corvus back-up and the 48k original memory. Both machines were configured to run on Pascal with Pascal CP/M and Pascal/Applesoft DOS interfaces, imprinted from master floppies. This enabled one to call the systems from the first volume of the Corvus and to use the remainder for data and to re-locate the standard DOS 3.3 on the Saturn.

This arrangement left each machine with multiple language capability. The size of expansion using the Saturn configuration provided the opportunity for a pseudo disc. An additional facility provided by Corvus Omninet enabled one to call systems from the Corvus instead of from floppies and to feed these to more than one computer. To do this, rearrangement of slots is required because the total number available is limited.

In addition to creating a data bank in text form, a useful set of graphics with both graphical diagrams and maps was created. The Apple "disc program" on the graphics tablet software was utilised in a modified form to indicate zones of damage from accident or aerial bombardment.

The area and distance facilities on the graphics tablet enabled calculations to be carried out providing a scale map had been traced on the tablet. Different selections from the colour menu were utilised to give indications of different hazard zones. It would be desirable to indicate the effects of wind drift, but this stage has not yet been reached.

Until enhancement had been carried out, the very large size of the necessary program had proved rather embarrassing, and so the Apple programs NUM1 and NUM2 were adopted to overcome the difficulty. The relatively large sized Saturn enhancement simplified this task.

The accepted standard practices using the Apple II and Apple IIe worked extremely well. The enhanced memory system shortened the recall time for data to a very impressive extent.

The field trials showed that it was possible to set up the necessary computing facility in a control centre from the time of opening the boot of a vehicle to unload the hardware to the stage of being

"in business" in a mere 20 minutes.

Although quite fast to operate, the software handling needed a little more sophistication. The key control of slot selection, booting, formatting, copying, loading, running and saving and interchanging peripherals was a little time-consuming and tedious. This was particularly so to those who had not developed skill in typing.

To interchange with ease between programs, particularly those written in different languages, necessitated the writing of complex software, for which we had neither the time nor the immediate skill.

A further aggravation was that software developers tended to corrupt their programs with copy protection devices that understandably were deterrents to rival commercial interests, but had no place in computer applications to a humanitarian service. It can readily be understood that the computer industry were granted a surfeit of impassioned appeal to make their procedures more friendly.

Many of the users' complaints mentioned fortunately fell on listening ears, and Apple emerged first in the field with the answer to practically all of the difficulties. Fundamentally, Lisa has a large memory - 1.5Mbytes compared to the 300k of the Saturn. This enabled our developments to continue on the lines we had determined.

The system used on the Corvus 10Mbyte unit for feeding systems into the central processor without the use of small floppy discs was transposed to the very much quieter Profile, which although only 5Mbyte size was quite adequate for the purpose.

The internal programming to enable access to a range of systems replaced the tedious key control system calling previously used. There were still two disc drives for use with large capacity 5in floppy discs, but a great deal of work can be done after the initial backing off of the systems on the Profile without the need to use them.

The use of the mouse enabled a great deal to be carried out without key controls, although its use for graphics meant freehand drawing, not tracing, and so calculation on accurate calibration was much more difficult.

The greater menu for graphics helped to overcome the difficulties in producing fallout plumes on the "bomb plot" program.

It would be rash to commit oneself to attempting a considered judgement without an extensive experience of using the new computer. However, the order of improvement and enhancement of this new development are so great that it would be foolish and a little uncharitable not to make early comment.

Lisa is very much ahead in its concept than anything we have yet encountered. It has met most of our demands that arose from the shortcomings of the previous models. Initially the time taken to start work to run through its test procedure and to start up the selected program seemed to be slow.

When measured, what appeared to be slowness was really the absence of activity on the part of the user. The complete absence of keying in, slot codes, system codes, file call codes, loading and editing commands, was overlooked and forgotten while Lisa was quietly doing all this at the press of a button and a touch on the mouse.

Likewise, the putting to bed sequence was not only a great time-saver but a help in avoiding mistakes at the end of a tiring session. Should one wish to pick up where one left off, a press on the starter button and a click on the mouse and this is achieved. This alone would commend Lisa and condemn previous methods to the wastebasket that sits quietly at the bottom of the Profile menu, taking the place of "delete" in previous parlance.

The desk top manager system simulates closely what one would require to do in the emergency headquarters or every day in the manager's office, with, however, consummate ease and lightning speed.

For the more scientifically and technically minded user, the program development toolkit will be impatiently awaited, since the ability to self program and to tailor to suit one's requirements would make this equipment even farther ahead of all others than it already is.

A minor drawback has been met that



IT is rare that a product lives up to its pre-release publicity, but Lisa, Apple's latest brainchild, looks like doing just that — and more. It is already established as the computer darling of the '80s and has set the standard for tomorrow.

For a software house such as ours the most attractive feature of Lisa is the combination of advanced hardware engineering, ergonomic design, and above all the development systems and the challenge of producing total integrated software to Apple's graphic mouse technology.

It's an upmarket product in the forefront of the computer world — and the task of the software house is to design and write systems to match Lisa's potential.

We believe that Busifile, a management information retrieval system does that. Its concept is based on both single and multifile record structures. At the moment Apple has accepted our single file system for Lisa. There will also be a trimmed version for Apple IIe and III.

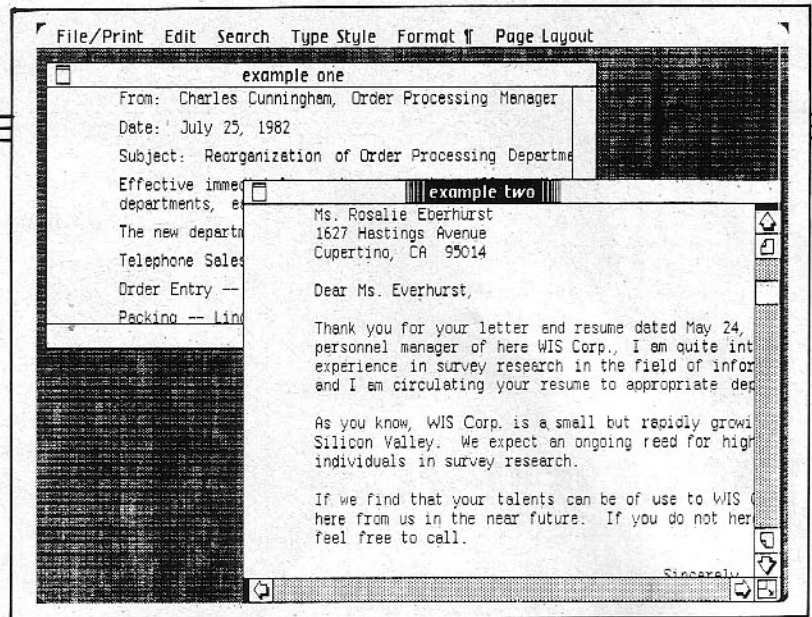
BSI is the offspring of Eurobeta Information Systems, authors of the Ormbeta range of application software. The first of those packages was conceived and written in Apple Pascal on an Apple II three years ago by Steve Ashcroft with a little help from a friendly accountant to specify the financial ledgers.

The design of the Ormbeta systems was to provide a range of application software built and written using a database as the nucleus. Its aim was to provide flexibility in file structures and record layouts, allowing people to customise the software to their requirements and even write their own systems based around the database.

Apple has installed Ormbeta packages at its European distribution HQ in Zeist, Holland, and at offices in Paris, Munich and Australia. All Ormbeta systems for Apple Computer run on Apple III with Profile, with communication facilities to other computers throughout Europe.

So when Apple asked BSI to put a new database system onto Lisa, we had the

\* Ken Helps is managing director of BSI Ltd.



# Designing systems for Lisa

choice of adapting Ormbeta or producing something new — hence Busifile.

Busifile was initially developed writing in Apple Pascal on an Apple III. It has 18,000 lines of source code (about 300 pages of A4) and its implementation on Lisa was achieved in two main stages:

- We used Lisa's development system,

---

By KENNETH HELPS

---

named Toolkit V1.0 BSI, to port the code files to Lisa, and then every line of the source code (18,000 of them) was amended to Lisa's advanced Pascal. At the same time initial debugging and enhancing to Lisa's special features took place. Busifile was up and running within

two weeks of starting the transfer process.

- The second stage consisted of linking the mouse technology to Busifile. This was achieved in two and a half days and involved writing a series of routines and preferences for the way in which we wanted the mouse to work with Busifile, and also to discover the fun of working with the mouse.

Lisa's overlay facilities allow the viewing of multiple files on screen as indicated by Figure 1, resulting in the developer being able to utilise the cut and paste routines within Lisa to transfer text from one file to another, or from within one file.

The development system has a single line command prompt which appears at the top of the screen, and most commands are activated by a single keystroke.

Lisa has three programming phases. The first involves compiling Lisa Pascal into an intermediate code, I-code. In the second the I-code is translated into machine code, and in phase three the

## From Page 39

Lisa has brought upon herself by being so attractive. She has many would-be viewers and has only a small screen. Simple daisy chaining would have helped here. Unfortunately, setting up a parallel monitor requires a little more than this. Coping with the power load and the increased resolution means additional electronics. Simple parallel coupling of another monitor does not suffice.

Some colleagues have suggested colour would be an improvement. They were not, however, colour blind and had overlooked the handicaps of a fair percentage of their fellows.

The ability to handle mathematical calculations in conjunction with pictorial graphics — scaled maps for example — could

be developed. In this respect, a graphics tablet would be a useful peripheral.

Linkage with other computers and communication by the telephone network is already available, but a worthwhile development would be the extension of this to high frequency radio communication, so that large hunks of data could be transmitted easily.

As further advances are made in miniaturisation, no doubt we will be able to look forward to tiny pocket-sized personal computers to link into headquarters. The advantage from the point of view of handling emergency situations speaks for itself.

Among the many spin-offs from this sort of development could well be an

application helpful to busy general practitioners to enable them to call up information while on their rounds.

One cannot get far in this work if one stays in isolation. We have benefited very greatly from innumerable contacts with other computer users and developers. I have been helped in the department by several enthusiastic colleagues, in particular Mr Richard I Popplewell, statistician and operational research adviser, Mr Malcolm P. Hodson, research officer, statistics and computing, University of Manchester, Mr Anthony G. Parsons, senior assistant emergency planning officer and Miss Geraldine Howe, personal secretary and computing assistant. ■



# APPLICATIONS

machine code is linked to Lisa's Pascal run-time support routines using the linker facility in the development system.

Such is the power and the processing speed of the Motorola 68000 16/32 bit chip that it took only 16 minutes to complete all three phases with Busifile's 18,000 lines of code.

The larger memory of the 16/32 bit computer allows the system developer to design and write programs with additional features, enhancing what is already written, and so providing the user with mainframe type application software on a micro, and in most cases an easier interface between the computer, application software and the user.

The end result helps the layman to understand computer software, and allows him to operate the computer/software system with the minimum of fuss, training and expertise.

Busifile forms the nucleus of a new range of products which are based on the simple principle that people should be able to store and retrieve information in the most efficient and effective routines but should also be able to define the modus operandi within the systems.

People should be able to define the structure and call the information from more than one database at a time.

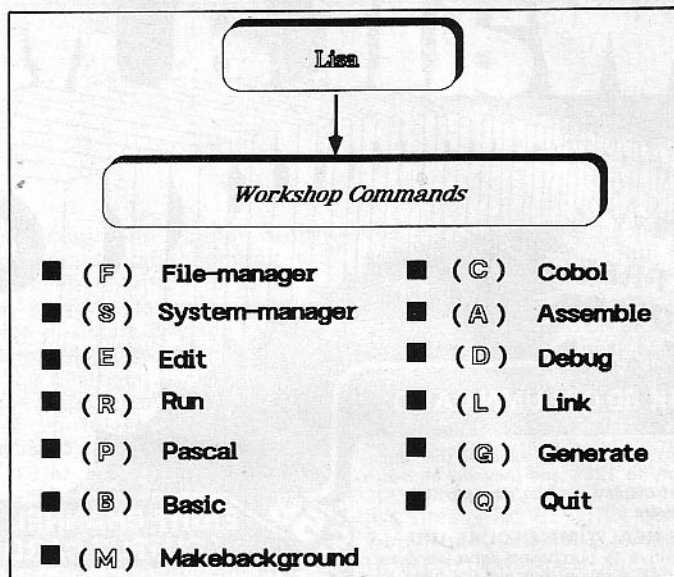


Figure 1: Diagram prepared using LisaDraw

Tomorrow's systems must link bases of information, whether the systems are sales ledger, word processor, critical path analysis or financial modelling programs. They must be totally integrated.

With the multi-file version of Busifile users have three databases at once from which they can update and retrieve infor-

mation.

Because BSI is linked with Lisa, it's not surprising that it takes a bullish view about its market prospects. In our opinion it is not just idle talk to say that Lisa is going to find its way onto the desks of a fair proportion of the world's top executives.



**HELLO!**  
My name is LISA.

I am the latest Apple computer and complement my older brothers the Apple IIe and Apple III.

Naturally, I am very clever but even I could not promise you trouble free operation and expert support and service. So, Apple in their wisdom have appointed Style Systems to help me help you.

They are experts, real professionals when it comes to applying computer technology. Many of you may find my powers to be of little use, then consult Style because my brothers may be able to help you.

I am very lucky to have been born into the Apple family, I am even luckier to have Style Systems Ltd help me (and you) grow.

So if you wish to see me, my brothers, or just talk to Style Systems about technology then call or ring any one of their three premises:-



Hull. Tel: (0482) 25318  
Birmingham. Tel: 021-707 8739

Style House,  
Adelaide House,  
Adelaide Terrace,  
Blackburn BB2 6ET.  
Tel: (0254) 51051



For full sales, service and support in the Thames Valley area talk to Berkshire's only Lisa dealer or come to one of our FREE Lisa introductory seminars and "hands on" sessions - phone Reading (0734) 581829 for details.



**THAMES VALLEY STSTEMS**

**T.V.S. GREYS HOUSE, 7 GREYFRIARS ROAD, READING RG1 1NU. TEL: (0734) 581829**



HOW many times have you watched your friend's computer do something, thought to yourself "I'd like to do that on mine", and then realised it would take a long time and finally be put off by the sheer volume of typing because "we have different machines".

Some people have attempted to remedy this awful predicament. The first approach I came across was a Canadian program which allows an Apple II to read a Pet's tape (Basic only) enabling the Pet program to be put into the Apple with as much translation as possible.

It was a brave attempt and worked very well except, of course, Pet graphics don't match the Apple's, the Pet has a timer and the Apple doesn't, etc. In other words, just about any decent program on the Pet just will not work on the Apple without considerable time and effort being spent on it.

A Dutch radio programme, Hobbyscoop, which looks at electronic developments, hi-fi, space research, communications computers and decided back in 1978 to try transmitting computer programs over the air - a natural progression from their earlier experiments in slow-scan television and viewdata pictures over the radio.

In 1979 regular transmissions of computer programs for Apples, Pets, Tandys and Exidy Sorcerers began to take place, but the low Baud rates (300 for some of the machines) and consequent long transmission times for these began to pall. Also, of course, only one brand of machine at a time could be serviced.

An answer was found by a group of enthusiasts who produced a "standard

Basic" which could be "read" and "written" by various home computers. This has now been rewritten to form a new, relatively powerful, language known formally as Nos-Basicode-2 which can be used by more than 17 different machines.

The language has been used regularly on the air since last January. Hobbyscoop is transmitted on Hilversum 2 on the medium wave 747kHz (401m) on Sundays at 1810 GMT (1710 in the summer) and can be heard in England, especially in the South.

Since the program is also being received in Denmark, Belgium, and West

By MAX PARROTT

Germany and encouraging reports of excellent data capture have been heard from all areas, the computer part of the programme also carries English explanations.

Basicode can help if you have access (maybe via friends) to a variety of machines. I think the situation which will benefit the most is a school or college with two or three different brands of computer. A useful program can then be written which will hopefully work on all of them.

Basicode allows most of the usual Basic commands such as ABS( ), CHR\$( ), GOTO, LOG( ), DATA, RESTORE, TAN( ), TAB( ), etc. all of

which are handled in the usual way. Those commands which tend to be very machine-dependent, such as HOME (which is CLS on many machines) or GET A\$ (which is AS=GET on the Beeb), are handled in subroutines, each specific to each machine but at a standard line number.

Thus, in place of CLS your program would include the statement GOSUB 100; at line 100 the command appropriate to your machine is included (together with RETURN of course). This would be CLS on a Tandy or the Beeb, HOME on the Apple, etc.

These subroutines, of which there are 12 standard ones, are all between lines 0 and 999 and are written out automatically in your machine when you run Basicode.

Having written your program it can be saved to tape in the standard Basicode protocol which can be read by any other of the, at present 17 machines, in the system. The Basicode package includes a manual and a tape which contains the programs for the following machines - Apple II, BBC (Models A and B), Commodore CBM 3008, 3016, 3032 (new ROMs) CBM 4016, 4032, 8032, and 8096, Commodore-64, Pet-2001 (old ROMs) and Vic-20, \*DAI Personal Computer, \*Exidy Sorcerer, \*OSI IP-Computer, \*Philips P 2000, Sharp MZ80A & K, \*SWTPC, \*Tandy TRS-80 Models I and III and the \*Videogenie.

In addition it is possible to buy a disc for CP/M systems with parallel port (Centronics type) and a Basic interpreter such as MBasic.

The manual explains the protocols (and the hardware modifications required for those machines marked with an asterisk).

The description of the language is good, assuming you know Basic, so you will soon be programming quite happily in Basicode. By the way, the tape contains a large number of programs written in Basicode for your delight. If you write any good programs Windfall and Hobbyscoop will be pleased to hear from you.

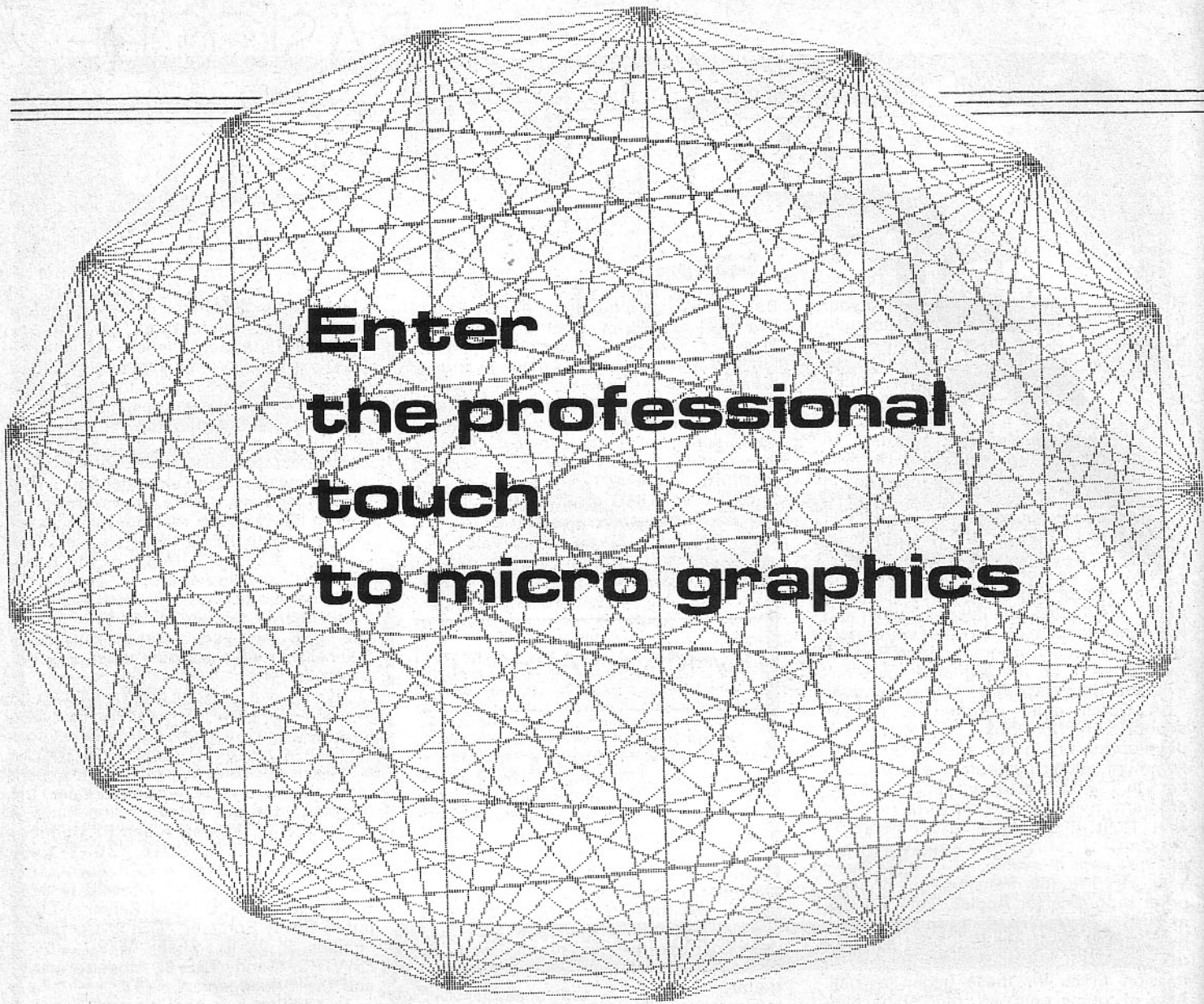
One "improvement" I would like to see someone undertake to write for Basicode is a set of standard high resolution graphics commands at standard subroutine line numbers. This will need a lot of thought and a good knowledge of the capabilities of other machines, so that a clever way of "transmitting" the details of these capabilities is found.

On the text side Basicode presently assumes that a video display consists of 24 lines of 40 characters, but requests that only 16 lines are used because many machines will only handle that number. Clearly this approach to the lowest common denominator cannot be used for good graphics.

● Unfortunately at the present time Basicode 2 can only be obtained from Holland. We hope to announce price and UK availability in our next issue.

# This new language is almost all things to all machines





# Enter the professional touch to micro graphics

FROM the moment I first saw it advertised I began pestering Windfall to obtain a Digisolve 512x512 Vector Graphics Board for review.

The campaign of harassment paid off and a few weeks ago I received a VGP board, monitor and assorted software from Digisolve.

My own love affair with computer graphics has been nurtured over the years by large mainframe machines or reasonably powerful minis utilising hardware graphics features. It has always been a disappointment to me that micros have relied on software to implement their graphics — until now.

Before continuing, it's worth expanding on the difference between software and hardware graphics implementation since most people don't know how the graphics are actually produced at the nitty-gritty level.

Take drawing a line on the Apple hi-res screen as an example. All the user does is issue the appropriate HPLOT command and — voila! — there's the line. However, in Applesoft the HPLOT instruction and its associated routines occupy a large chunk of memory and are quite complex.

These routines must decide which pixel (screen point) to switch on in order to display the line. This is much more difficult

---

---

By PETER GORRY

---

---

than it sounds because, except for horizontal or vertical lines, pixels just don't line up where you need them.

To come up with a good line drawing algorithm which doesn't produce dog-eared or ragged lines is quite demanding, and as a result drawing a line requires a considerable amount of machine code program and is relatively slow.

For a number of years more expensive computers have been utilising progressively more complex hardware chips to perform functions such as line drawing. These devices are extremely fast and can plot millions of pixels a second. The only problem is that hardware graphics features normally have to be designed into the machine from the beginning.

What Digisolve has done is to take such a Vector Graphics Processor (VGP) chip and interface it to the Apple. The VGP then performs the various graphics operations in hardware which previously were done in software — and a lot more besides.

An important point to get clear is that

the Digisolve board does not replace or improve the normal Apple hi-res graphics. It is totally independent of them.

The best way to think of the Digisolve system is as a peripheral like a printer or a plotter. The card plugs into any slot and contains the VGP, associated video circuitry and 64k of memory for the two graphics pages.

The monitor displays the VGP graphics only and does not display the normal Apple screen, thus in normal use one has two monitors in operation.

It's a shame that the system can't be switched between the two, thus eliminating the need for two monitors. In fact, if you already possess a monitor you will probably want another anyway, since the Digisolve system needs a slightly long persistence screen for optimum results.

The image on a normal monitor suffers from flicker due to the interlaced video needed to obtain the 512x512 resolution. This and the cost of the board render the system outside the price range of most home users and the main market will clearly be a professional one.

The VGP is supplied with a disc containing the machine code routines (automatically installed on booting the disc) which are required to drive the board from Basic. This software support offers 24



new commands to control the VGP using the & feature in Applesoft.

Digisolve is to be congratulated in adopting a policy of total software disclosure. The manual and disc contain complete listings and assembler files of the software so that (machine code) programmers can easily add to or alter the software. I wish other manufacturers were so forthcoming.

Many of the commands use familiar Applesoft words but they don't behave in quite the same way. For instance, &HGR, &HGR2 select the page to be displayed but they don't clear the screen. &CLEAR does that. Since the commands just exploit the inbuilt features of the VGP, it's simpler to look at these directly.

The VGP can handle two pages of 512x512 pixel resolution in black and white. As already stated, it takes care of the vector generation itself and can generate them at an incredible maximum rate of 1½ million pixels a second.

In fact, the process of getting the co-ordinates to the VGP is the limiting factor from Basic and the net vector generation rate is much slower.

In many ways the VGP can be treated as a "plotter" and the instructions follow a typical plotter format. One must select PEN mode in order to write to the screen and the "pen" must be set DOWN to write. UP mode produces no visible effect on the screen.

The VGP offers four line types – solid, dashed, dotted and dot-dashed. However, unlike a plotter it also has an ERASE mode which erases lines. The erase mode is equivalent to plotting in black in normal Applesoft graphics.

Although the screen co-ordinates are 0-511, 0-511 (with 0,0 at the bottom left of the screen – sanity at last!) the VGP works with co-ordinates in the 0-4095, 0-4095 range and there are two display modes associated with this.

CYCLIC mode wraps round the screen on a 0-511 basis – that is, any vector which goes past the end of the screen re-enters from the opposite side.

CLIP mode wraps round only on the 0-4095 basis, so unless the vector is extremely long it has the effect of clipping the vector at the screen edges.

This is a very welcome feature but it's a pity that the routine can't handle negative co-ordinates too so one could forget about having to check co-ordinate values entirely.

The VGP can operate in INVERSE mode, in which case drawing a line inverts the colour of the pixels along the line. This is the missing "XPLOT" command in Applesoft, although even here INVERSE mode does not work for ERASE, which is a pity. The whole screen can also be inverted or cleared to white.

Single pixels are set by the appropriate plotting command and &AT returns the status of any individual pixel. Other routines include MOVE, POS and STORE. The latter two set up the position and size of a non-destructive inverted cross hair cursor on the screen.

If this was all the VGP offered it would still be impressive but it also has an internal character generator for putting text on the screen. This is an indispensable feature which is so sadly lacking in normal Applesoft graphics.

The character generator is very flexible, and as well as providing writing along the X and Y directions the size of the writing can be individually scaled in X and Y. This allows one to generate a variety of text – small, large, tall and thin, short and fat – and just to top it all there are two character types normal and italic.

I had no trouble with the Digisolve support software and I was writing programs with impressive graphics in no time at all.

With such high resolution, one's mind soon turns to animations and games and it was here that I found my first problems.

The best animations are produced when one page is updated while displaying the other, followed by a quick display swap and repeating the process. Unfortunately, the ability to display and draw on different pages is not standard and it requires a modification by Digisolve to achieve this. I'd like to see that become standard.

Also presently, the software does not support shape tables, although this should be remedied soon.

Apart from these shortcomings I was very impressed with the facilities on offer, and the speed with which complex and very high resolution graphics could be produced does bring a totally new professional feel to micro graphics.

I think we can soon expect to see computer aided design applications normally reserved to more powerful machines making their way to the Apple.

"What about printing a screen or saving it to disc?" must be the next question.

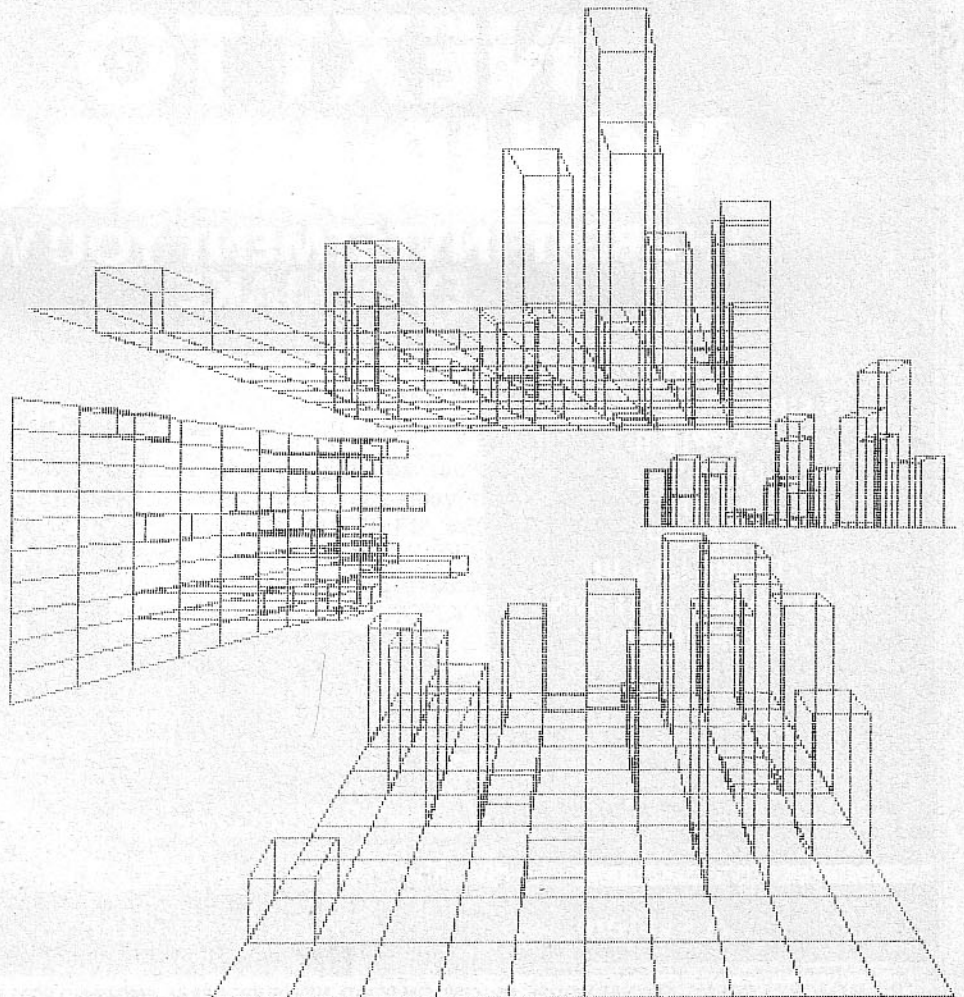
Since the VGP's memory is not part of the Apple's normal memory, the usual printer facilities for dumping the hi-res screens are of no use, neither can BSAVE be used to save a picture to disc.

These two operations are provided for on a utilities disc which contains machine code routines for this purpose.

The disc received for review was configured for an Epson MX80 and Grappler+ interface card. You would have to check with Digisolve to ensure that your particular printer/card combination is supported.

With the MX80 the graphics resolution provides only 480x512 points, not the full 512x512, so a small part of the screen can't be dumped.

Wider printers would, I presume, provide the complete image. The manual





# DIGISOLVE

and disc contain full listings and assembler files as before.

The routine for saving a picture to disc is easy to use but the process is somewhat slow, and it takes a huge 118 sectors to save a picture (there are over 260,000 pixels after all). The reload procedure is also slow and, generally, I found that the picture could be redrawn more quickly than it could be loaded from disc.

When working from Basic the limiting rate in drawing is the speed with which commands can be sent to the VGP, and the obvious way to increase the rate is to use a compiler. However, no compiler can handle the extra & commands that drive the VGP since they do not exist in standard Applesoft.

Digisolve has gone some way to overcoming this problem with a set of support routines for use with the Microsoft TASC compiler. Several of the machine code routines are replaced by Basic sub-routines but machine code is retained for vector generation.

These latter routines use specific variables defined in a common block to enable the machine code to find them. Converting a working Basic program to one that will compile thus requires a reasonable amount of rewriting, and not all commands are supported in the TASC

version.

It's clear that this area could do with more work but at least the foundation for such support is there, fully documented.

If you are a Pascal programmer don't despair, because also supplied for review was a Pascal support disc. This contains a set of procedures similar to the ones described for Basic to perform the VGP commands.

It is a nice feature of Apple Pascal that machine code routines can be treated as if they were normal Pascal procedures, and Digisolve makes full use of this.

Unfortunately, the disc doesn't contain routines to perform printer/disc dumps from Pascal and there are no routines that mimic the Pascal turtle graphics, but I gather that such a system is in preparation.

The Digisolve VGP system is new and, as such, there are not many support packages available for it but that is probably changing rapidly. Since Digisolve has produced interfaces to micros other than the Apple, this wider base should spur on independent software development.

One novel application I have already seen is a Visicalc pre-boot disc which uses the VGP to display the Visicalc model. The text size can be altered from very small (and hence a lot of cells on the screen) to

really very large indeed.

What about the future? The latest generation of vector graphics chips offer a host of hardware features not available here, such as multiple windows, panning, scrolling and zooming, area filling and in-built graphics primitives such as arcs, circles and rectangles.

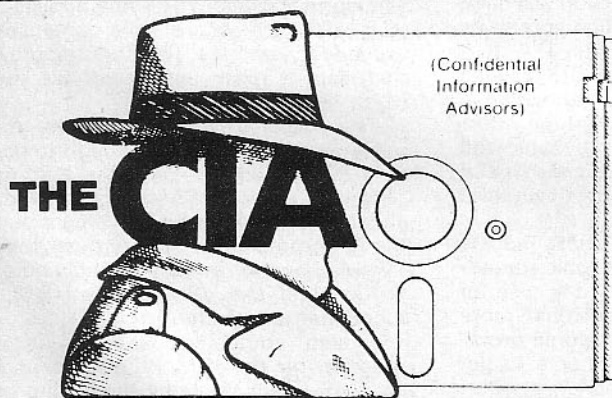
Although getting these into a form that can be used on the Apple may be some way off, I'm sure that such powerful and exciting facilities will become available in due course.

At Apple '83 I spent some time with Digisolve investigating its colour VGP system (there are two versions, each offering 512x512 resolution with either 64 or 4096 different colours). Also unveiled there was a very impressive graphics system from Trisoft which uses the Digisolve colour system.

I spent some time with its author putting it through its paces, and it really is remarkable.

It offers a host of facilities that I've seen only on machines costing £20,000 or more. In fact the globe/route-map sequence at the end of the British Airways Manhattan television advert was done with this package.

Hmm... it would be nice to have the colour system to review. now where did I put Windfall's telephone number?



*Why buy Commercial Software and not know what's going on?*

Meet 'TRICKY DICK' and his accomplices:-

- The LINGUIST
- The CODE BREAKER
- The TRACER
- The TRACKER

Why not buy a kit of 5 powerful disk espionage utilities, all on one disk, that allow you to investigate, edit, locate list, trace, rescue, translate, patch, repair, verify, examine, protect, unprotect, encrypt, decrypt, and analyse programs or textfiles on normal and PROTECTED disks. This kit includes the 'CIA FILES', a 60,000 + word book on all aspects of the Apple ® disk, many of which have never before appeared in print. "The CIA Files" gives you extensive, detailed, easy to read, information on disk repair, file recovery, DOS patches, copy protection, disk formatting, program encryption, and many other vital topics, showing you keystroke by keystroke, EXACTLY how to do it!

CHECKSUMS:  
defeat them, Y/N?

HEADERS:  
change them as  
you like - even  
insert 'wild cards'!

WRITE BACK  
TO ANY DISK!

(You need an Apple II+™ or IIe™,  
48K and 1 drive)

TRAILERS: modify at  
will, wild cards if  
you please!

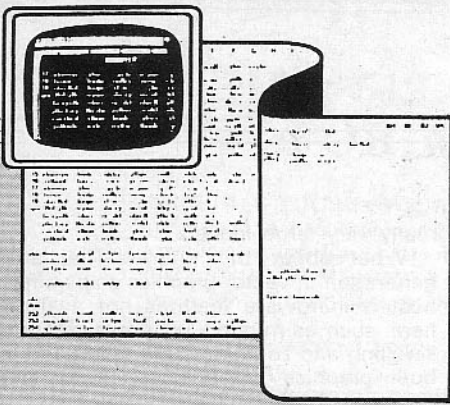
LIST: in APPLESOFT, INTEGER, ASSEMBLY LANGUAGE  
direct from the nibble dump of a protected disk!

Specify ENCODING 6 & 2, 5 & 3, 4 & 4,

TRICKY DICK THE LINGUIST BY T TSE  
 D596AA Y AADE DOS SL=6 T=23 <00> VOL  
 D6ABAE N 0000 3.3 DR=1 S=0F <-> 254  
 D5AAAD 0 DEAAEB PR=0 <AL> <62> <H>

Send £35 to  
GOLDEN DELICIOUS  
SOFTWARE LTD.,  
(Sales Dept. W3),  
The Penthouse,  
7 Sloane Avenue,  
Chelsea,  
London SW3 3JD





# On choosing horses for courses

WHENEVER I have the opportunity I like visiting computer dealers to talk with them about new developments in computing and software. On a recent visit to such an establishment in Reading, owned and run by two attractive and friendly young ladies, one of them asked me: "When are you going to start teaching Beebcalc"?

"Never heard of it", I said.

"Beebcalc is the BBC micro answer to Visicalc, and we have already sold five such packages", came the reply. This was enough to make me immediately interested in Beebcalc, and I decided to try it there and then.

Before proceeding any further, try this little test: Given that Beebcalc has a matrix of 26 columns by 99 rows compared to Visicalc's 63 columns by 254 rows, would you say (without using paper and pencil or a pocket calculator) that Beebcalc is  $\frac{1}{6}$ ,  $\frac{1}{5}$ ,  $\frac{1}{4}$  or  $\frac{1}{3}$  the size of Visicalc?

Did you make your guess? Did you think it was a  $\frac{1}{5}$ , a  $\frac{1}{4}$  or a  $\frac{1}{3}$  of the size of Visicalc? Well in that case you were wrong, because Beebcalc is in effect only  $\frac{1}{6}$  the size of Visicalc. So we are obviously not comparing like with like, but then if you work with a 48k or a 64k Apple you could not use more than  $\frac{1}{5}$  of the Visicalc matrix anyway.

So I decided to proceed — especially when I heard of a prospective electronic spreadsheet user who could not make up his mind on whether to buy an Apple with Visicalc or a BBC computer with Beebcalc).

I started by browsing through the manual and noted that both Beebcalc and Visicalc commands start with the special character /. And there the similarities end.

Besides having a different command structure, Beebcalc commands like /I, /D, /W, /T function differently compared to the same commands used in Visicalc. I then turned by chance to page 20 of the Beebcalc manual and discovered to my amazement that the system does not support brackets, which put me off completely from going any further.

Beebcalc, like Visicalc, calculates strictly from left to right. For example,  $2+3 \times 4$  equals 20 (incidentally the same calculation performed on Supercalc or on Multiplan will produce a different answer, that is 14 not 20, because these programs

obey the rules of algebra and give precedence to multiplications before performing any additions).

If you want to change the order of calculations, using Visicalc you can enter  $2+(3 \times 4)$  and get the answer 14. With Beebcalc, on the other hand, you will have to rearrange the expression and enter it as  $3 \times 4 + 2$  in order to obtain 14.

But how about  $(2+3) \times (4+5)$ ? It looks to me that on the Beebcalc you will have to execute this calculation in three intermediate stages: first use a Beebcalc cell to calculate  $2+3=5$  then use another cell to calculate  $4+5=9$  and finally use a third cell to calculate  $5 \times 9=45$ .

It could be that many business models can be developed on an electronic spreadsheet without resorting to the use of brackets, but as one becomes more experienced with using electronic worksheets one demands and expects to get more out of them. So sooner or later Beebcalc users will get an urge for brackets which the program will not be able to satisfy.

What do you do then? I suspect that Beebcalc users could face the daunting prospect of having to switch to a technically more advanced electronic spreadsheet and start all over again.

Going from Beebcalc to the other extreme I noted this week that the price of the sophisticated Micro Modeller business planning package for 8 bit computers (which includes the Apple) has come down from £595 to £250. What more can I say?

Returning to Visicalc, I had the opportunity recently to conduct an in-house course in VC and for the first time used the program on the IBM Personal Computer. The training skills of using Visicalc on one brand of computer can be easily transferred to other brands but there are nevertheless some significant

and some marginal differences in using it on different machines.

Take for example, the IBM PC. To save a Visicalc file on it you can only use an 8 (yes, eight) character name compared to file names of 30 characters on the Apple version of Visicalc. The same applies also to the TRS 80, where a file name cannot exceed 8 characters. The Commodore is in that respect a little better, with file names of up to 16 characters.

The following are some minor manipulations which I could not perform on the IBM PC. I could not /E (that is, edit) numbers. If I enter 12345678 and want to change it to 12345678, I cannot use the /E command for that purpose, instead I would have to retype the whole number. Otherwise the /E on the IBM PC functioned as on the Apple.

I also could not set the Visicalc program on the IBM PC to accept /GFI or /GF\$ (that is global formatting of all the cells to display either integers only or only two decimal places).

Finally, if you develop a VC model on the IBM without first creating a VC data disc you will not be able to save your model. That's because you can not use the Visicalc program to initialise a Visicalc data disc for use on the IBM PC (as you would if you use Visicalc on an Apple).

You will unfortunately have to abandon your model (unless you have simultaneous access to two IBM PCs), load the IBM DOS, create one or more VC data discs and start all over again developing your model using a blank spreadsheet.

Another habit I developed using the Apple VC version is to type only the first two letters on any function, press any of the arrow keys or the space-bar and let the program complete the entry.

For example, If I type @AV or @LO or @CO or just @S (note, just a single character) followed by any of the arrow



keys or by the space bar, Visicalc will complete the entry and display @AVERAGE( or @LOOKUP( or @COUNT( or @SUM( etc.

Why not try it for yourself – but not if you use the IBM PC or other computers on which you can run Visicalc where these shortcuts cannot be used. On the other hand, one of the things which I liked in particular about the IBM PC was the large + key. On my Apple IIe that can only be used in conjunction with the upper case key. What unergonomical positioning for such a frequently used key!

I forgot who it was who said that poetry should always be read in the language in which it was originally written. I am inclined to suggest that computer programs, like Visicalc, should preferably be used on the computer on which they were first developed. It's almost like selecting horses for courses.

Take the Visicalc versions for the Commodore or the TRS-80 or the HP or Atari-800. Serious VC exponents who use the program on these computers rather than the Apple or the IBM PC may tolerate the inconvenience that their version does not have the useful /E (edit) command. So every time they have to alter or correct a formula they have to retype the entire expression, which very often could be a long and complex entry.

But when these people find out that their version of Visicalc does not have such invaluable functions as @IF, @AND, @OR, @CHOOSE, @TRUE and other similar logical functions, and when they discover the astonishing and spectacular feats that you can do with the aid of these functions, then they get a real feeling of being cheated and deprived of something good which they fully expected to have when they bought Visicalc.

The significance of choosing horses for courses, or which spreadsheet is best suited to which computer, becomes more critical when one considers future developments in spreadsheet technology.

As I mentioned earlier, the more you get to know Visicalc, or any other spreadsheet program, the more you demand from it. This is inevitable because one of the beneficial effects of using electronic spreadsheets is that it improves your working knowledge of your business and industry and encourages you to apply more advanced quantitative manage-

ment techniques which up to now you could not be bothered to use.

This in turn could lead to dissatisfaction and a demand for more sophisticated computer programs for management accounting applications.

So once the bug of using electronic spreadsheet gets hold of you and you hear that your version of the program has been upgraded, or that a new more advanced version has been issued (like the change



By NICK LEVY  
Principal,  
Interface Management

from the 13 sectors to the 16 sectors of the VC program), the temptation to change up to the latest technology is too big.

Even if you are a conservative user of electronic spreadsheets, and for a while feel content with the results that you get on the now superseded version of the program, sooner or later you will want to discard your "old toy" and start using the last word in technology.

And what is that last word in spreadsheet technology? (Not to mention Lisa which from the price point of view is in a class of its own.) One large company I know of, once dedicated users of Visicalc advanced version on Apple III, decided to switch to the IBM PC in order to be able to use the Lotus 1-2-3.

This piece of software combines a spreadsheet, database, word processor and graphic facilities, integrated into a single package.

Is this going to be the trend? That com-

pany had a number of Apple IIIs and trained staff well versed in using VC advanced version, so the switch could not have been taken light heartedly.

Is using integrated software going to be the future, trend. If so, then what should Apple users do? They cannot at present use the 1-2-3 because to the best of my knowledge there is no version compatible with Apple DOS.

Would Apple users be able to use the Visi-On when it arrives to the UK or is it only wishful thinking? Visi-On, considered to be the equivalent of the 1-2-3, is a piece of software produced by the makers of Visicalc, with similarly integrated programs as the 1-2-3 (spreadsheet, database, word processing and graphics).

It is supposed to be capable of functioning almost like the Lisa – complete with a Visi-On mouse – for a small fraction of Lisa's cost.

Lisa, incidently, is an integrated hardware package which contains, in addition to the above four facilities, also a project management program and one for making visual presentations – six integrated programs in all.

So am I right in assuming that the makers of Apple are unlikely to encourage or adapt their machines for use with Visi-On, with which they have to compete. (I understand that Visi-On, being an integrated package, could only be used on computers with a minimum memory configuration of 256k.)

So on behalf of Apple owners who use the computer in their business, may I appeal to whoever is listening: Could we please have, fairly soon, a high quality integrated package like the Lotus 1-2-3, for use on the Apple?

At this point I should have fulfilled some of the promises that I made in last month's article, including the second set of exercises for mastering the skills of replicating Visicalc commands and functions.

I was put off from doing this when I discovered that the article which appeared in the September issue of *Windfall* under "Practice makes perfect" was still the draft version. Let me assure you however that except for the presentation of the replicating exercises which I revised in the final version, the article required no significant changes. 🍏



Forth is a language much applauded by computer buffs – and there is a thriving UK Forth Society (c/o Goldie-Morrison, 15 St Alban's Mansions, Kensington Court Palace, London W8) to prove it.

The language has appeared on the Apple under many guises (see *Windfall* July 1982, Page 43 and August 1982, Page 49) but probably under none better than this British package from Metacraft.

# Go Forth and multiply...

THERE comes a time when most Basic programmers become frustrated with the language and recognise the fact that it was not really intended as a serious programming tool.

Whether this realisation comes about because of the lack of speed, structure, or the fact that you are locked into the set of commands implemented in the version you use is really immaterial, the cure is the same. Either write part of your code in machine code and add it to your Basic program, or change languages.

The first choice is lots of fun if you like a challenge but is not very productive as coding can be quite slow. The neatest way I know of incorporating machine code routines to Basic is to use The Routine Machine (see review *Windfall* June '83).

The Routine Machine enables Basic programmers to add Forth-like extensibility to the language which is nice. However, to use Forth itself to make the Apple really powerful can be even nicer.

I first became interested in Forth (the name comes from a truncated "fourth generation language") when listening to a tape recording of the high guru of the Apple, its designer Steve Wozniak.

He replied to a question about his favourite language by saying it would be a mixture between Forth and Pascal. Well I tried Pascal and found it really didn't suit my programming style (or perhaps lack of it).

The rigid discipline imposed and the suspense of waiting to see if the compiler would let me through just this once became all too much, and took me back to the days of punch cards and programming cycles of several days.

Not that I'm knocking Pascal, I believe it is the best language available on the

Apple for many applications, and feel much more comfortable with commercial software written in it than Basic. However, at the moment it's not for me. What I want is speed of programming, a language that will grow with me, and a compiler that lets me know the moment I make a mistake. These qualities I found in Forth.

Forth was first used to control telescopes and is used at Jodrell Bank. It is also the language that was used for the simulation sequences in "Starwars".

The version of Forth sent to me for review is by Metacraft, a company based in Crewe. It compares very favourably with other versions I have tried and is Forth 79, not a hybrid like Graforth or Transforth.

Metacraft's Forth is not just a language but, like Apple's Pascal, is a language system. That is to say you get an editor, compiler, assembler, decoder, debugger and documenter, plus a copy utility and disc formatter. All of which help towards really productive programming.

Metacraft's Forth is fully compatible with the Forth 79 standard as documented by the Forth standards team. In addition it has several useful extensions including:

**Hi-res Graphics** – HIRRES, HCLR, HCOLOR, HLIN, HFIND, BKGND, ASPECT-RATIO, HLINE, HPLOT, HPOSN.

**Turtle Graphics** – MOVETO, MOVEBY, PEN-UP, PEN-DOWN, HEAD-ING, TURN, TURNTO.

**Lo-res Graphics** – LORES, COLOUR, CLRTOP, PLOT, VLIN, SCRIN.

**Shapes** – DRAW, HSCALE, XDRAW.

**Trancendentals** – SIN, COS.

**Paddle control** – BUTTON PADDLE.

**Strings** – CONSTANT\$, ARRAY\$, !\$, @\$, IN\$, INSERT\$, LEN\$, MATCH\$, =\$, TYPE\$, (plus many more)

**Misc** – PEEK, POKE, RANDOMISE, TEXT, NOTE.

In addition to the above set of Apple-related extensions to the language, CASE statements, including nested CASE

*\* Mike Glover is the managing director of Leicester Computer Centre. A regular contributor to Windfall, he has written several articles on machine code programming and using the Epson printer.*



structures, are supported.

The following rather trivial example shows the structure of the nested CASE statement without taxing the intellect too much. Non-Forth programmers should start reading at the word PARTNER (RETRY is a present for Forth buffs). This table of Forth words with their nearest Basic equivalent may also help:

<b>FORTH</b>	<b>BASIC</b>
PAGE	HOME
..	PRINT
KEY	GET
CR	? CHR\$(13)

(See Figure I).

The development tools in this version of Forth are:

**Copy:** This utility, as its name implies, allows the user to copy all or part of a disc from one disc to another using single or multiple drives. Unformatted discs may be prepared by typing the word FORMAT.

Beware, FORMAT is lethal! It doesn't molycoddle and ask if you really meant to clear your disc it actually gets on and formats it, no questions asked.

Those of you like me that want to be protected from yourselves may like to redefine FORMAT as follows. This example serves to show what is meant by an extensible language (See Figure II).

Note that by redefining FORMAT, all future references to the word will use the latest version, although the redefined version itself uses the original.

**Document:** This utility is used to produce paginated listings of source screens (Forth programs). It can also produce a summary of screens and an index of your disc. A useful word is OUTLINE, which only prints those lines that start with '(', ':', or 'C'. This provides a summary of your code.

**Decode and dump:** These are two very useful, and not often provided, tools. Decode will produce a listing of the word it was asked to decode showing its composition, Figure III shows the output generated by asking it to decode my version of FORMAT.

The 'P' shows that the word is in a protected dictionary and the ':' shows that it is a colon definition.

The dump utility will dump selected memory locations in both hexadecimal and character format to the screen or printer.

**The editor:** This enables us to create and modify Forth screens (blocks of 1024k). The screen is divided into 16 lines of 64 characters. If you do not have an 80

## By MIKE GLOVER

column card then only the leftmost 37 characters are displayed. The screen will, however, scroll right and left to show all of its contents. I would have liked a further command to enable the screen to flip from leftmost to rightmost characters as the CTRL A key in Pascal.

There are three main modes in the editor:

**Overwrite**  
**Insert**  
**Delete**

These do much as their names imply, and I found them easy to use. There are

some changes I would like, for example CTRL I instead of CTRL T for tabbing, but because of the nature of Forth I can customise the editor – written in Forth, of course – to suit my own tastes.

A word of warning from the recently wise – if you change an editor make sure you keep a working copy or two. If you hurt it there is not much you can do, as you need an editor to fix it!

A search and replace feature is included in the editor and will, if required, deal with more than one screen at a time, the user specifying which screen to stop at.

One feature I particularly liked was the ability to do a trial load. By typing CTRL L from any of the modes the editor saves

```
Metacrafts FORTH      review by MIKE GLOVER

SCREEN £2
0 ( NESTED CASE EXAMPLE                               18 JUL 83 MGG )
1 : (RETRY)  R> 2+ @ EXECUTE ; ( RETRY COURTESY K LANDER )
2 : RETRY    COMPILE (RETRY) LATEST CFA , ; IMMEDIATE
3
4 : PARTNER  PAGE ." STATE SEX ( M OR F ) " KEY
5           CASE ASCII M OF CR ." ARE YOU FAT OR THIN?( F OR T ) " KEY
6           CASE ASCII F OF CR ." HI THERE FATMAN "          ENDOF
7           ASCII T OF CR ." HELLO SKINNY "                  ENDOF
8           DEFAULT CR ." I DID ASK FOR F OR T BYE... "
9           END-CASE ENDOF
10          ASCII F OF CR ." MARRIED OR SINGLE? ( M OR S ) " KEY
11          CASE ASCII M OF CR ." NICE MEETING YOU "          ENDOF
12          ASCII S OF CR ." ARE YOU FREE TONIGHT? " ENDOF
13          DEFAULT CR ." WRONG RESPONSE BYE... "
14          END-CASE ENDOF
15          DEFAULT DROP RETRY END-CASE ;
```

Figure I

```
: 3BELL      BELL BELL BELL ;
: FORMAT    3BELL CR ." WARNING THE DATA ON DRIVE " DR ?
           ." IS ABOUT TO BE DESTROYED "
           ?CONTINUE IF FORMAT THEN ;
```

Figure II

```
DECODE FORMAT
DEFINITION OF FORMAT

4558 U: 3BELL
455A P: CR
455C P: (." ) WARNING THE DATA ON DRIVE
4579 P DR
457B P: ?
457D P: (." ) IS ABOUT TO BE DESTROYED
4579 P: ?CONTINUE
4599 P OBRANCH 004 4
459D P FORMATok
```

Figure III



the current screens and tries to compile them.

If no errors are found then you are returned to Forth. If on the other hand you do make a mistake, then a warning message appears and you are returned to the editor with the cursor on the place that the trial loader thinks there is an error.

This is a feature which will be appreciated by beginners and experts alike and puts Metacraft's Forth in a class of its own as far as I am concerned.

Forth is very fast compared to other languages. Table I will give you some idea how it compares in speed with Pascal, Basic and how Metacraft's Forth compares with some other versions of the language. The test being used is "Eratosthenes' sieve" to find 1899 primes repeated 10 times, as defined in Byte, January 1983.

One of the things that makes Metacraft Forth quick to use is the use of overlays. This term may not be entirely clear to all readers so a brief explanation may be in order.

Most of Forth is written in Forth itself, including for example the editor. If this were to be loaded in from a series of Forth screens then there would be some delay as the system compiled the new words.

With Metacraft Forth this delay is eliminated by keeping the code in its compiled (usable by the Apple) form and then loading it directly into memory as required.

Another nice feature of this version is the ability to change the system overlays to suit your particular configuration.

While the manual made it quite clear how to remake the whole system, it was less explicit on how to remake individual overlays.

A call on the hotline soon fixed that and while a call to Crewe is less glamorous than ringing California it is much more convenient, to say nothing of the cost. It also served to show that the product was supported effectively.

For the record, this is the procedure to follow should you wish to suppress line feeds in the printer overlay.

- Activate the editor and display the screen (source code) containing the printer constants by typing 106 LOOK.
- Change 0 CONSTANT HAS-LF on line 4 to 1 CONSTANT HAS-LF.
- Quit the EDITOR, then remove it by typing FORGET OVERLAY.

Product	Seconds
Metacraft Forth	164
Fig Forth	190
Pascal (Apple)	509
Transforth	1080
Applesoft Basic	2806

Table I: Benchmarks

```

SCREEN £106
0 ( The Game of Life                                22-MAY-83 KGL )
1
2 : life ( gen n --- ) ( play game of life using space in block n )
3   start
4   0 do
5       display.status
6       check.neighbours
7       display.generation
8   loop
9   display.status ;
10
11
12
13
14
15
KEY:
STACK DEPTH: 2
STACK(H) 6B 14
STACK(D) 107 20
LOOP INDEXES J I : 2564 1797B
EMIT:
fWORDS=0

```

Figure IV: Example of de-bugger display

- Recompile the modified printer source code by typing 106 LOAD.
- Remake the printer overlay by typing 31 SAVE OVERLAY.

One feature that is particularly useful is the debugger. This allows you to move through a program whose code is displayed on the screen, a cursor showing you which word will be executed next.

Pressing RETURN causes that word to be executed and the cursor moves on to the next word. This allows the programmer to see that the control flow is as intended – or not, as the case may be.

At the same time the contents of the stack are displayed at the bottom of the screen in both hexadecimal and decimal. Also shown are the stacks depth, the loop counters, a count of the number of executed words (useful for measuring performance) and any text that may be output by the program under test (see illustration).

I am not aware of any other version of Forth that has this feature built in, indeed very few language systems have anything like it.

Users of Apple Pascal have a tantalising D)debug shown on the command line but it has not yet been implemented, presumably because it has bugs. (Basic users might take note of Ian Trackman's Super Trace) (See Figure IV).

Screens 33 to 140 contain all of the system overlay source code, most of it written in Forth enabling the adventurous user to customise it. They also provide some very useful examples of programing style.

Screen 13 is automatically loaded as part of the boot sequence. I change mine to print a friendly message, but a more practical use would be to set it up to load

in and run an applications package. Full instructions on how to create turnkey systems are included in the manual.

The 170 page manual is very comprehensive, but does not set out to teach Forth. It is divided into four main sections covering:

- Installation guide – Loading & running, printer output mass storage, backups.
- Development tools – how to use the editor, copy, decode, development and debug.
- System description – internal structure, extensions assembler, strings etc.
- Glossary of word sets.

I would have liked to see some more worked examples, such as the method of changing printer constants.

If you want to learn the language then I strongly recommend The Complete Forth by Alan Winfield. This book costs £6.95 and is published by Sigma Technical Press. Don't be put off by the lurid cover design – the contents are really good.

Starting Forth by Leo Brodie is also an excellent introduction to the language, and is regarded by some as the best computer book ever.

#### SUMMARY:

- \* Metacraft's Forth is a powerful development tool for the Apple II and Iie.
- \* Support is UK based and is good.
- \* At £69 it is very good value for money.
- \* If you don't know Forth then also get the Complete Forth.
- \* Requires 48k Apple II or Iie with at least one disc drive.
- \* Supports 80 column cards.
- \* Floating point word set is available for £15.

**Health warning** – Forth is very addictive.



Computer training falls into two main categories – training users to get the best from their machines and using the computer as an interactive aid in training personnel to carry out specific tasks, such as operating a chemical engineering plant.

In this special section we look at both categories from the viewpoint of people actually involved in the training industry, and also investigate the importance of training to dealers.

A measure of the importance Apple UK attaches to the first category is given by the fact that the company has started a promotional scheme for potential new users which includes a free computer familiarisation course (this would normally cost £25).

Under the same scheme three Apple training centres have been established in London, Birmingham and Manchester. They are run by Keyboard Training in conjunction with Apple.

# Shortfall poses major threat to Britain's micro boom

THE importance of training in the micro market is universally acknowledged, but it is practically impossible to quantify this importance exactly.

A rough and ready guide to the American market comes from the Californian market research company Dataquest, which says that \$14 billion will be spent on personal computers by 1986 and the training industry will have captured \$3 billion of this.

But the American pattern does not seem to apply in the UK. In the States the Computerland chain is busy ensuring that all its stores are equipped with attached classrooms, whereas in Britain there seems to be a tendency for dealers to become less and less directly involved in providing training.

Britain is probably the most sophisticated country in the world when it comes to micros and has the highest per capita level of home computer ownership.

But this comparatively high level of sophistication does not mean that Britain does not have a significant training need, particularly in the prime growth area – the

**The biggest threat to the development of the micro market in Britain could be the lack of adequate training facilities, according to RICHARD LEE, managing director of The Computer Training and Education Centre. Here he argues that training is of vital concern to dealers.**

professional and business markets.

In the long, and even medium term the greatest single threat to continued development of the micro market is likely to be a combination of fear and ignorance among potential users – it's a combination which results directly from inadequate training facilities.

There will be little point in the industry developing more and more powerful hardware and more and more sophisticated software if possible users are unable to take full advantage of them.

It follows from this that cutting prices

to the bone and so selling cheaply at the expense of offering training as a support service may win short term sales, but will be self destructive in the longer term.

Proper training is urgently needed to bridge the gap between increasingly powerful micros and the millions of technically untrained executives, professionals and small businesses who already own, or are on the point of acquiring, such machines.

If training fails to keep pace with developments in hardware and in software then these developments are largely wasted.

It is reasonable to suggest that inadequate user training for micros has meant that growth in the market has already slipped behind its potential.

Dealers must be prepared to meet the needs of their customers in the field of training if they intend to stay in business for any length of time and if they wish to capitalise on business opportunities available at the moment.

It doesn't follow that dealers need to lay on training courses themselves but

## PACKAGES PREFERRED TO MANUALS

*A QUICK, do-it-yourself guide to the selection of training for businesses is given by Bill Reeve of Circuit UK:*

The cheapest option is to use the manuals that come with the system, or to buy some teach yourself manuals. This approach is fine if your staff have the time and the inclination to learn this way. It normally requires some previous computer experience as well.

Computer-based training packages go one better. If well written, they are less "wooden" than a written manual, and can detect and explain operator errors better.

Better still are courses supervised by an instructor, either in-house or externally. If the groups are small enough, it allows for

individual problems to be covered. This human element is especially important for those new to computers.

There are some advantages in having an in-house course:

- People learn best when they are relaxed and the familiar surroundings of your own office tend to favour such relaxation.
- The best training uses examples from your own day to day work. Such examples are readily available in-house.
- Courses are less likely to disappear into a void of generalities that do not relate to your particular needs, if everyone on the course is from similar backgrounds and has similar interests.

However external courses score when you consider that one computer terminal per trainee is the best way of training, and courses run in-house are liable to be interrupted by telephone calls, or superiors wanting urgent work done.

In addition, written manuals, computer based training and most internal courses follow a standardised pattern. This means you often come away still uncertain how it all relates to your own business.

The very best type of training is that which has an appreciation of how the expertise is going to be applied afterwards. This is only possible with customised courses, which at the moment means instructor supervised courses.



that they must be able to advise customers on where they can get such training.

Ideally the dealer should be able to arrange for his customers to obtain full training of whatever type they need in order to gain maximum benefit from their new Apple.

There is still a need for basic introductory courses for the beginner, but increasingly the business user is looking for more specialist training.

Such courses as programming and advanced programming in Basic, CisCobol, Z80/8080, 8086 Assembler and Pascal, are language based courses which demand a fair degree of computer literacy from the participants.

Other requirements include general introduction courses for management and courses geared to particular software packages. In the longer term there will still be a need for the language based and general introduction courses, but the great growth area will be tied to packages.

Assuming that I am right about this development, it will mean that training emphasis will move even further away from the retailers and towards the software houses and specialist training houses.

There is no question that training will remain a vital part of the micro boom. It is reasonable to assume that a point will be reached when the need for basic introductory courses will diminish, but that is still a very long way ahead.

It is also reasonable to assume that by then the choice of new software will be such that there will be a massive growth in refresher courses and package-specific courses.

The businessman who wishes to get the most from his micro doesn't need to be fluent in Basic or Cobol or whatever, nor does he need to be able to write a program.

What he needs is a basic understanding of what can and cannot be done and a detailed knowledge of the particular software packages he wishes to use.

The basic "make friends with your micro" courses lend themselves to being run by dealers themselves and, to some extent, are available through evening classes and adult education courses.

It is not reasonable however to expect dealers to get involved in training much beyond this basic level, and most adult education setups wouldn't really know where to begin running courses tied to software packages.

This leaves dealers with a serious problem. They need to provide ever more sophisticated training back-up in order to retain and develop market share. But profit margins have been squeezed to a point where investment in training as a support service by dealers has to be strictly limited.

I cannot claim to be totally objective when I say that the solution lies with the

specialist training house — but it is what I truly believe. The dealer does not have to commit money and other resources to providing a comprehensive training programme. He needs simply to know where such facilities are available.

In this way customers have the chance to buy as much or as little training as they require. They are not put off the dealer by an absence of training support, neither are they faced with a product price which is uncompetitive because it includes a training element.

As a trial run, we at CTEC have begun to market our courses through dealers just like any other product.

Dealers sell our courses for us on a straight commercial basis and make a profit from so doing or, if they choose, they absorb some of the cost and use

CTEC to provide the sort of support training which they no longer invest in themselves.

We have provided training for a number of dealers for some time but the idea of merchandising the courses at the point of sale is a new one. At the moment we're running our test with some 50 dealers and the first indications are that it is going to be very successful.

There may be other ways in which dealers can overcome the problem of ensuring that training is provided in ever more sophisticated and specialist forms.

One thing the dealer cannot do, however, is to walk away from the situation — at least not if he wants to remain seriously in business.

Training is, and will remain, a key element in the micro market place.

## *The road to more productive use of your Apple*

TRAINING is important. Yet although micros have been increasingly used in business, education and the home in the last six or seven years, training has always received a low priority rating.

Why should this be? Perhaps the public has been indoctrinated with the idea that, being such a technologically advanced machine, the computer will be able to put itself right if an error occurs.

Another common misconception is that the programs have been written in such a way that errors will never occur.

Unfortunately, as most users will know, this is not the case.

From the very first moment you start to use an Apple you are starting on a learning curve, which should continue to increase as time goes by. In some respects you receive a lot of "training" from the booklets which accompany the machine.

Apple Computer has long recognised the need for full documentation to support the user and I would say that it was probably the first micro manufacturer to provide training courses for dealerships, schools and colleges, and to promote business awareness by holding seminars up and down the country.

So much for a pat on the back for Apple, but what emphasis has been carried forward into the dealership network? Sadly, the answer is very little. Training is still assumed to be too costly,

both in terms of time and money, and only productive in the long term.

Training falls into three distinct areas. The self taught, teaching by manuals or training aids, and training by an outside agency. Let us first look at the self taught method.

The approach adopted by most home computer users is the "suck it and see" method. If it works ... fine, if not, try again. So much trial and error has to take

---

By JEFF TURNER

---

place before any positive results are achieved that the user can easily become disillusioned with the computer and wish that the purchase had never taken place.

Happily, most true home users are tenacious to the Nth degree and will seldom give up. However, the businessman who starts using the Apple in his home is quite different. He cannot afford the time to keep repeating processes to see if he can get the machine to do what it was bought to do.

As he, or she, will probably want to use



ready-made programs, the learning should be made easier by inbuilt prompts, but this only relates to the running of the software and not to the general use of the computer as a machine.

I have encountered a large number of computer users who do not know how to handle discs, replace printer ribbons, feed paper into the printer and so on. Where was the training when the computer was delivered, or sold?

Most dealers assume that a person

buying a computer for their home must know what they are doing and, therefore, no training is required. Unless you happen to have a friend who owns an Apple, or belong to a local user group, you will be out on your own!

Always ask for a small amount of training when buying a machine or additional peripheral card. If the dealer can't answer your questions then I would not really trust him to train you.

A reputable dealer should always take

the trouble to find out how to use the equipment that he sells and should be able to pass on this knowledge.

Being self taught is very "British". In a sense you are fighting adversity and overcoming the disability of ignorance - yet while doing this you could have been using your Apple much more constructively and increasing your enjoyment into the bargain.

I learnt to use many business programs in this manner, although I was helped by having studied programming at college and having worked with mainframes for a while.

The micro was a new breed of computer and in the early days the *only* way to learn was to teach yourself. Nowadays there are many people who have had experience of Apple computers and there is no reason why their knowledge cannot be channelled into training courses for new users and businessmen.

The number of times I've heard salesmen, dealers, and even myself (though it's hard to admit to it) say: "You'll find the information you need in your manual".

In one sense this is a valid response to a query from a user. Manuals are written to inform, and therefore train, and should always be read by the user. Most manuals seem to be relegated to the bottom shelf of the cupboard where they are promptly forgotten about until a salesman says "You'll find the information you need in your manual".

I know that the writing of a manual is

## HOME USER TRAINING

Method	Cost	Results
Games	Low	Better manipulation of keyboard. Improved responses to screen commands.
Manuals	Low-moderate	More understanding of software.
Trial and error	Nil	High waste of time but more awareness of the computer and software achieved.

## BUSINESS SYSTEM USER

Method	Cost	Results
Seminars	Usually low	Increased understanding of software.
Disc-based courses	Low-medium	The step by step approach means a self paced learning curve. Good for an operator who is already competent.
College course	Medium	Usually based around the writing of programs rather than the operation of the Apple.
On-site training	Medium-high	The personal attention offered, though costly, should produce most results. Don't try to learn too much too soon.

## FINANCIAL MODELLING COURSES

'Hands On' Practical Microcomputer Modelling

- 1 DAY COURSE VISICALC
- 1 DAY COURSE ADVANCED VISICALC
- 3 DAY COURSE MICROMODELLER

## CONSULTANCY

Model Design, Writing, Support

- VISICALC
- MICROMODELLER
- PROSPER STAR
- PROSPER +
- MARS
- FCS/EPS

## VAL WARDEN CONSULTANTS

110 Western Road,  
Tring, Herts. HP23 4BJ  
Tel: Tring (044 282) 6774/5

## TRAINING FOR MICRO USERS

Go on a course at the Computer Training and Education Centre

CTEC training conducted at our London Centre and on-site at client's premises is used by numerous government departments and blue chip UK companies.

### SCHEDULED COURSES

Introduction to Microcomputers	1 Day
Basic	5 Days
Advanced Basic	3 Days
Programming in C	3 Days
CP/M 80 Operator Level	2 Days
CP/M 86 MSDOS Operator Level	2 Days
Introduction to d/Base II	1 Day
Programming d/Base II	2 Days
Pascal	5 Days
Wordstar Wordprocessing	3 Days
Z80 Assembler	5 Days
8086 Assembler	5 Days

Provider of Training to the Manpower Services Commission

Contact The Courses Secretary  
Computer Training and Education Centre Limited  
102-108 Clerkenwell Road, London EC1 5SA.  
Tel: 01-251 4010.





not a welcome task. The information has to be succinct, easy to read and, when possible, interesting as well. As a training aid the manual should always be accessible to the user, and should always be read at least once.

The ideal training environment is a one-to-one arrangement. This can only be achieved when a home user has a knowledgeable friend, or a company spends money to buy in the expertise. Outside agencies cater for a wide range of training requirements through:

**Seminars:** Where a group of people will listen to a talk then apply the information just received in a hands on session.

**Demonstrations:** Explanations of the use of a specific program, the reports that can be produced and the possible pitfalls of the program (if the demonstrator is being honest).

**In-house training:** A service which may be costly but is probably the most effective. A user will be taught at his own Apple and using his own package. By covering the program operation as it

would occur on a normal day the user feels more at ease with the program, after training, and should make fewer mistakes when operating the Apple.

All Apple users, whether in business or at home, have gone through one of the above training processes and should recognise areas where further training may be required.

I am convinced that some form of training IS required and the sooner it is done the better. Training results in a more productive use of the Apple.

# CBT ... training on tap whenever it is needed

COMPUTER based training (CBT) covers any aspect of instruction delivered or managed by a computer.

Information may be presented via computer-generated text, graphics and simulations, by retrieval from a database, or through other media such as audio or video tape and disc.

Trainees can interact with the computer which can direct them to the next most appropriate learning sequence for their needs, provide tests and compile reports on an individual's progress for the trainer.

CBT has several advantages over conventional training methods. It provides a consistent standard of training "on tap", wherever and whenever it is required.

It provides a highly interactive and individualised method of learning and trainees can follow a CBT course at their own pace.

Studies have proved that this shortens courses by as much as a third and that retention may be higher.

For the trainer, updating courses is faster and easier, while the identification of an individual student's problem is easier to anticipate, particularly if management reporting facilities are incorporated into the system.

The advantages of using CBT in simulations of dangerous situations or those where the equipment used on the job is very expensive are obvious.

Most CBT at present is related to computing skills such as learning programming or the use of particular applications programs.

But it has considerable potential in other areas such as medical education, safety training for workers in chemical plants and in the use of robotics for engineering.

The practice of CBT is not widespread

\* Janet Rothwell is Senior Consultant, Training Strategy Division, National Computing Centre

**The Apple can be a powerful teaching tool. Here computer-based training is examined by JANET ROTHWELL with special reference to authoring languages and a look ahead at the potential for combining video technology with micros.**

in spite of its advantages. This is due partly to the conservatism of many trainers, but more specifically to the cost of developing software.

The off-the-shelf CBT packages available tend to be limited to such areas as computer literacy, learning touch typing and word processing.

A number of business simulation exercises are beginning to appear like BL Systems "Fillet and Billet" game, and Mills and Allen's "The Petty Cashier" (both available for the Apple), but otherwise packages are intended for educational use and are not suitable for the training area.

Several companies specialise in writing customised CBT courses, but although such packages can be inexpensive - as little as £1 for each student hour - when used for a large number of trainees the costs are prohibitively high for smaller companies with specialist needs.

This leaves trainers with the option of developing courseware, as software for CBT is known, for themselves, and it can be a time-consuming business.

Apart from the time taken to design a traditional course, there is the additional time of writing and testing the software - and most trainers do not have a background in computing.

But this isn't always necessary.

Trainers will have already the skills for structuring material and conveying it to trainees and have access to Apples which could be used to deliver small modules of training during the working day.

There is no need to wait for a course of

training to have a sufficient number of trainees, or to reschedule work to allow a training course to be held.

CBT also eliminates the need for expensive travel and hotel costs and for extensive rewriting and reprinting of training materials because of new legislation or new management directives. CBT material is readily updateable.

There is a range of possible approaches to using the Apple for computer based training. All rely upon the trainer planning a course of instruction in a systematic way following through the stages of analysing the training problem, setting objectives, devising validation methods, questions, tests and so on.

Much CBT material has been written using conventional programming languages, particularly Fortran and Basic which were designed primarily for calculation and data manipulation and to handle text line by line.

This is not really appropriate for CBT applications, which are concerned mainly with presenting information and checking the responses of trainees.

A number of authoring languages and systems are available which enable courseware to be developed much more quickly than using conventional programming languages - a time saving of as much as 60 per cent is claimed for some systems.

An authoring language is a high level applications language specially designed for writing courseware.

An authoring system will also incorporate a method for inputting text, graphics and instructions in author mode and for executing instructions in student mode.

A good system will both improve the productivity of the programmer and also make the computer much more accessible to an inexperienced computer user.

One of the earliest author languages,



developed in the USA in the early 1970s, was Pilot (Programmed Inquiry, Learning Or Teaching).

It originally had only eight single letter commands but is available in a highly developed form as Super Pilot from Apple.

This package has 22 commands and four editors which simplify the task of inputting the program and creating the special effects that an author may require in a lesson.

Super Pilot is a further development which has increased file handling and computation capabilities and allows the control of peripherals such as video discs and other audio visual devices.

(Apple Pilot was covered in *Windfall* in January, June and July 1983 and Super Pilot in *Windfall* in August 1983.)

An inexpensive system with great potential for trainers who wish to develop their own courseware, Super Pilot is written in Pascal and it is possible to link into Pascal programs to further enhance the power of the system.

For beginners it comes with Super Copilot, two discs of lessons which teach the user how to program in Super Pilot. In themselves they provide an example of how such CBT material can be written.

The whole package is well documented, providing a language manual and a manual for the editors, and is designed to protect the user from too many problems.

Super Pilot's main disadvantage is that it can be slow. Some instructions access separate parts of the interpreter and there can be delays while "special effects" files are brought into memory from disc.

Another Pascal-based authoring facility available for the Apple is Staf2, although the original Staf2 is Fortran based (this version needs the Z80 card and CP/M to run on the Apple).

Staf2 is an enhanced version of Staf (Science Teachers Authoring Facility) designed and specified by the Staf Development group.

Its origins lie in Lal (the Leeds Author Language), and it was developed originally under the national Development programme in Computer Assisted Learning.

A Staf2 program is written as a text file using a suitable editor. This file is then validated to produce a random access file (known as the teaching program) which is in turn processed by the Staf2 interpreter to present the program to the student.

It can be used to write complex programs involving calculations and special subroutines which provide the mechanism by which the tutorial or training session may be adapted to the student.

Statistics can be collected on responses, and the responses themselves may be stored in files.

Its range of text matching options is wide, and an answer counter can be employed to control the number of attempts at any particular question, and to suitably route students to the next action, for example, remedial instruction or more

difficult questions.

It is not a particularly easy language to learn for inexperienced users because of the nature of its coding, especially in the handling of subroutine switching, but it does cope well with most types of training problems if they do not rely too much on graphics.

Microtext is an authoring system developed at the National Physical Laboratory which is currently available for the BBC and CBM micros. Apple and CP/M versions will be available by the end of the year.

It is a frame-orientated system designed for ease of use. Lessons are designed as a series of numbered linked frames, so that an author can concentrate upon the content and flow of the training material.

It is easy to use, and a particular good point is that provided certain precautions are followed, such as limiting the size of the text area, Microtext programs can be portable across a range of different manufacturers' machines, since they are stored in straightforward Ascii format.

Files can also be prepared using a word processor. The graphics are of the teletext type which may not be satisfactory for some applications, but as with Super Pilot, there is the capability to control other types of media.

The future for CBT lies with the convergence of computers with video technology. The ability to present clear visual examples opens up many exciting possibilities for all types of training.

sibilities for all types of training.

Owl Micro-Communications has adapted its Viewdata editor to enable an Apple to be linked to the Philips Laser-vision video disc player and to write simple CBT training packages.

It can also be used to download software through the viewdata system into the Apple for the delivery of courseware in local sites.

More sophisticated authoring systems such as Combat, which is a menu driven system offering very comprehensive facilities for the author, need more memory than the Apple offers.

In this case the development work is done on a larger machine and courseware is then down-loaded to the Apple for delivery to the student. Some companies have created networks especially for this purpose.

This article has discussed some of the possibilities available for using an Apple in training. As well as evaluating the features of any particular system for CBT, the trainer should first evaluate the likely effectiveness of using a computer in any given area, and management will require a detailed look at its cost effectiveness.

In addition, people learn better when there is some variety in the learning situation, and CBT should therefore be considered as a component part of the total learning environment, which can enhance training and offer considerable benefits to both the trainer and the trainee.

## Solution on tape ...

**Cassette-based training may be a solution for someone who doesn't know how to switch on a micro or boot a disc but who does know how to use an ordinary tape cassette. Here RICHARD SUMNER puts the case for this type of training.**

MANY users never receive the correct training, and thereby never realise the full potential of their systems. The problem may lie in the instruction offered.

The busy professional new to computing does not want to waste time, nor be spoken down to. The new employee needs to be brought up to speed in hours, not weeks. The rusty user may need only superficial help.

Several self-help systems on the market cater for these needs. Some require a certain amount of knowledge (or help from a third party) such as where the instruction is itself disc-based. Others require a good deal of reading and study.

By using audio-cassettes however, you can teach yourself. These systems assume no experience, no programming knowledge, and no machine handling ability.

You learn by actively doing, not by passively reading. In effect you are sitting down with a knowledgeable friend, using your own Apple at your own pace and convenience.

The system lends itself to sophisticated software packages such as Visicalc and Wordstar. Many people often only use a fraction of their capabilities and cassette-based courses can help you get the best out of them.

A typical example of cassette-based learning is the FlipTrack series, recently brought to the UK from the US.

The actual course is recorded on one side of the tape. Whenever a user wants more detailed information on a topic, he simply turns the cassette over, without having to rewind the tape. There are six Apple-related titles in the series, costing between £45.95 and £89.95.





# Package deal

**THIS** game has none of the zap and kill aspects of your average arcade game, but could well be described as the thinking man's Space Invaders.

As a storeman your function is to fit a selection of boxes into a specified area with as little wasted space as possible.

Each box is marked with a price tag and your points tally depends largely on the total value of the load you manage to tuck away.

The game is best described as a graphics plus text game. It is well written, in Basic, and is worth the few minutes spent typing it onto disc.

Required skills? To achieve high scores demands a little thought, a good eye for fitting shapes to a space (or less good guesswork) and, to some extent, a gambling instinct.

```

10 TEXT : HOME : HTAB 15: PRINT
  "PACKAGE DEALS": PRINT
20 PRINT "YOU ARE SUPERVISING THE
  LOADING IN"
30 PRINT "SEQUENCE OF 5 TRAILERS
  FROM AN OFFERED"
40 PRINT "SELECTION OF 12 CONTAINERS
  (OF 10"
50 PRINT "DIFFERENT INTEGRAL UNIT
  DIMENSIONS"
60 PRINT "GIVING BETWEEN 1 AND 16
  UNITS CAPACITY)."
70 PRINT "THE TRAILERS MAY NOT BE
  LOADED BEYOND":
80 PRINT "THEIR 4 BY 2 BY 2 UNIT
  S CAPACITY (SAME"
90 PRINT "SIZE AS LARGEST CONTAINER)
  - ATTEMPTING"
100 PRINT "TO DO SO RESULTS IN A
  FINANCIAL PENALTY."
110 PRINT "YOUR INCOME COMES FROM
  ACCEPTANCE OF"
120 PRINT "THE OFFERED VALUES."
130 PRINT "THE DELAY IN MAKING YOUR
  DECISION ON"
140 PRINT "ACCEPTANCE OR REJECTION OF
  THE OFFERED"
150 PRINT "CONTAINER DETERMINES YOUR
  EXPENSES."
160 PRINT : PRINT "YOU WILL BE RATED
  ACCORDING TO YOUR"
170 PRINT "FINANCIAL GAINS AND YOUR
  ABILITY TO"
180 PRINT "MAXIMISE THE TRAILER
  LOADS."

```

```

190 DIM A(9),B(9),C(9),D(9),E(10),F(10),G(9),H(9),J(9),K(12)
200 FOR A = 0 TO 9: READ A(A),B(A),C(A),D(A): NEXT
210 FOR A = 0 TO 9: READ G(A),H(A),I(A),J(A): NEXT
220 FOR A = 768 TO 901: READ B: POKE A,B: NEXT : POKE 232,0: POKE 233,3
230 PRINT : PRINT "HOW MANY SUPERVISORS (1 TO 6)?"
240 POKE -16368,0: GET A$: IF A$ < "1" OR A$ > "6" GOTO 240
250 FOR A = 1 TO VAL (A$)
260 :
270 REM START OF TRAILERS
280 :
290 FOR B = 1 TO 5: FOR D = 0 TO 10:E(D) = 0:F(D) = 0: NEXT
300 HOME : HGR : HCOLOR= 3: SCALE= 1: ROT= 0
310 HPLLOT 60,37 TO 220,37 TO 220,112 TO 60,112 TO 60,37
320 FOR Z = 0 TO 9:X = G(Z):Y = H(Z): GOSUB 1130: DRAW Z + 1 AT I(Z),J(Z): NEXT
330 VTAB 22: PRINT "SUPERVISOR " :A:" TRAILER NUMBER " :B
340 VTAB 24: PRINT "PRESS <SPACE-BAR> TO CONTINUE "
350 POKE -16368,0: GET Z$: IF Z$ < > " " GOTO 350

```

```

360 :
370 REM START OF OFFERS
380 :
390 FOR C = 1 TO 12: HOME : VTAB 21
400 PRINT "A B C D E F G H I J UNITS": FOR D = 0 TO 9
410 PRINT CHR$ (F(D) + 160 + 16 * (F(D) > 0)):" " : NEXT
420 IF C > 1 THEN HCOLOR= 0: GOSUB 1130: HCOLOR= 3
430 Z = INT (10 * RND (1)): IF E(Z) > 1 + 17 / (D(Z) + 2) GOTO 430
440 HTAB 37: PRINT F(10):E(Z) = E(Z) + 1
450 X = 65 + INT ((151 - B(Z) - C(Z)) * RND (1))
460 Y = 107 - INT ((66 - A(Z) - B(Z)) * RND (1)): GOSUB 1130
470 PRINT RIGHT$ (" " + STR$ (C),2):". LOAD " : CHR$ (193 + Z):
480 D = 16 * D(Z) + INT (12 * SQR (D(Z)) * SGN (RND (1) - 0.5)) * RND (1))
490 PRINT " (UNITS="D(Z):") OFFERED AT $":D
500 PRINT "ACCEPT (A) OR REJECT (R)?"
510 FLASH : PRINT " : NORMAL : POKE -16368,0

```



# GAME

```

520 K(A) = K(A) - D(Z): IF PEEK
    (- 16384) = 210 GOTO 860
530 IF PEEK (- 16384) < > 193
    GOTO 520
540 :
550 REM LOAD ACCEPTANCE
560 :
570 IF F(10) + D(Z) > 16 THEN Z$
    = "CAPACITY": GOTO 1080
580 IF Z < 2 OR Z = 9 GOTO 850
590 :
600 REM SINGLE COMBINE TESTS
610 :
620 ON Z - 2 GOTO 630,680,640,64
    0,660,670
630 IF F(8) > 0 GOTO 1070
640 IF F(7) = 0 GOTO 680
650 GOTO 1070
660 IF F(5) > 0 OR F(6) > 0 GOTO
    1070
670 IF F(2) > 0 OR F(3) > 0 GOTO
    1070
680 IF F(10) - F(0) - 2 * F(1) <
    13 - Z + (Z = 4) GOTO 850
690 :
700 REM MULTI-COMBINE TESTS
710 :
720 ON Z - 2 GOTO 750,770,790,81
    0
730 IF Z > 16 OR F(4) = F(5) OR
    F(4) = 2 GOTO 850
740 GOTO 1070
750 IF F(4) = 2 OR F(4) = 0 AND
    F(5) < = F(3) GOTO 850
760 GOTO 1070
770 IF F(4) = 1 OR F(4) = 3 OR 4
    * F(8) + 2 * F(5) + F(2) =
    4 GOTO 850
780 GOTO 1070
790 IF F(2) = F(3) OR F(3) = 2 GOTO
    850
800 GOTO 1070
810 IF F(4) = 1 GOTO 1070
820 :
830 REM JUSTIFIED ACCEPTANCE
840 :
850 F(Z) = F(Z) + 1: F(10) = F(10)
    + D(Z): K(A) = K(A) + 100 *
    D
860 NEXT : K(A + 6) = K(A + 6) +
    F(10): NEXT : NEXT
870 :
880 REM SUMMARY
890 :
900 TEXT : HOME : HTAB 12: PRINT
    "MANAGEMENT RATINGS"
910 PRINT : PRINT "SUPERVISOR $
    VALUE LOAD %VALUE %LOAD"
920 FOR A = 1 TO VAL (A$): Z$ =
    STR$ ( ABS (K(A)))
930 IF LEN (Z$) < 3 THEN Z$ =
    RIGHT$ ("00" + Z$, 3)
940 IF K(A) < 0 THEN Z$ = "-" +
    Z$
950 Z$ = RIGHT$ ("      " + Z$
    , 11)
960 PRINT "      ": A: LEFT$
    (Z$, 9): ".": RIGHT$ (Z$, 2):
970 PRINT RIGHT$ ("      " + STR$
    (K(A + 6)), 5):
980 PRINT RIGHT$ ("      " + STR$
    ( INT (K(A) / 2040 + 0.5)), 8
    ):
990 PRINT RIGHT$ ("      " + STR$
    ( INT (K(A + 6) / 0.8 + 0.5)
    ), 8)
1000 NEXT : VTAB 24: PRINT "MORE
    CONTAINERS TO BE LOADED (Y/
    N)? ":
1010 POKE - 16368, 0: GET A$: IF
    A$ = "Y" THEN RUN
1020 IF A$ < > "N" GOTO 1010
1030 HOME : VTAB 10: PRINT " R
    ETURN FOR MORE 'PACKAGE DEAL
    S'. ": END
1040 :
1050 REM CAPACITY/CONTAINER ERR
    OR
1060 :
1070 Z$ = "CONTAINER"
1080 HTAB 1: PRINT " YOU CANNOT
    ACCEPT THIS ": Z$:
1090 K(A) = K(A) - 50 * D: FOR D =
    1 TO 3000: NEXT : GOTO 860
1100 :
1110 REM SUBROUTINE, CONTROL &
    SHAPE DATA
1120 :
1130 HPLLOT X, Y TO X, Y - A(Z) TO
    X + B(Z), Y - A(Z) - B(Z) TO
    X + B(Z) + C(Z), Y - A(Z) - B
    (Z)
1140 HPLLOT TO X + B(Z) + C(Z), Y
    - B(Z) TO X + C(Z), Y TO X +
    C(Z), Y - A(Z) TO X + C(Z) +
    B(Z), Y - A(Z) - B(Z)
1150 HPLLOT X, Y TO X + C(Z), Y: HPLLOT
    X, Y - A(Z) TO X + C(Z), Y - A
    (Z): RETURN
1160 DATA 10, 5, 10, 1, 10, 5, 20, 2, 10
    , 5, 30, 3, 10, 5, 40, 4, 10, 10, 20, 4
1170 DATA 10, 10, 30, 6, 10, 10, 40, 8,
    20, 10, 20, 8, 20, 10, 30, 12, 20, 10
    , 40, 16
1180 DATA 18, 87, 9, 77, 170, 15, 178,
    19, 244, 154, 253, 129, 85, 154, 10
    8, 129, 80, 20, 109, 16
1190 DATA 235, 20, 241, 24, 0, 154, 38
    , 125, 165, 154, 200, 140, 233, 95,
    267, 96, 0, 30, 30, 33
1200 DATA 10, 0, 22, 0, 35, 0, 48, 0, 58
    , 0, 69, 0, 82, 0, 93, 0, 105, 0, 117,
    0, 125, 0
1210 DATA 9, 30, 23, 174, 55, 110, 9, 3
    , 63, 76, 228, 7, 0, 54, 21, 55, 54,
    45, 101, 228, 103, 33, 28, 63, 6, 0
1220 DATA 41, 173, 223, 51, 54, 118, 4
    , 5, 12, 4, 0, 54, 54, 54, 45, 101, 36,
    36, 28, 63, 7, 0
1230 DATA 45, 45, 222, 27, 110, 17, 63
    , 55, 54, 45, 45, 4, 0, 45, 45, 222, 2
    7, 110, 17, 63, 55, 54, 5, 0
1240 DATA 41, 45, 222, 27, 54, 54, 14,
    45, 37, 60, 7, 0, 54, 46, 45, 222, 51
    , 110, 9, 36, 36, 36, 4, 0
1250 DATA 41, 245, 54, 54, 23, 45, 5, 0
    , 73, 49, 54, 54, 30, 63, 28, 4, 0

```



# ABG

## Top quality Pascal-based system for business graphics

APPLE II Business Graphics (ABG) is a piece of software of the quality that you always mean to write yourself – if it wasn't for the house, car, kids, work ... and all the other things that prevent one from doing so.

ABG is not perfect – after all it was written by someone else – but if I ever meet the person who wrote it I'm not sure whether I would shake their hand or kick them for reminding me of how really good software should be written.

My views may be coloured by recent comparison with a piece of mainframe CDC software, the operation of which made unravelling the Rosetta stone seem like child's play. Even given this factor, ABG really is a very good package indeed.

It is a Pascal-based system for producing a variety of business graphics presentations. It needs as a minimum a 64k Apple and two disc drives, although to be useful you will also want a printer and a digital plotter. One doesn't need to have Pascal – or indeed any knowledge of it – to use the package since it is quite self-contained.

It consists of two copies of the master disc (they are copy protected), a data/demo disc and a comprehensive manual in the usual Apple house style. The manual's size may seem a little daunting at first, but it is very clearly written, well indexed and one only needs about 20 per cent of it for most applications.

The program is not menu driven, which given all its facilities is not surprising, however it makes up for this in two ways. Firstly there are comprehensive HELP lists available simply by typing HELP followed by the area you want information about, such as HELP EDIT or HELP DRAW.

Secondly all commands are in plain English (well American, so there's a few words they don't spell too well). The really clever part is that you can abbreviate the commands as much as you desire as long as the resulting instruction is unique. Thus SET HORIZONTAL TITLE "SALES" could become SE HOR TI "SALES" or even SE H T "SALES".

The tutorial section takes you through exercises in creating three data files on oil consumption, production and imports, editing them and graphing them. I was amazed to see that within a few minutes of starting I had produced an impressive

colour triple-bar bar chart of these quantities fully labelled and titled.

The process was easy, quick and entirely painless. The production of pie charts or line graphs from the same data required only a couple of instructions. A further four instructions and I had a line graph of imports as a percentage of consumption.

ABG produces full colour graphics and operates in two modes. One cycles just green/violet; the other uses five colours. I used it on a monochrome monitor for

---

---

By PETER GORRY

---

---

some of the time and I preferred it on the whole to using it on a colour monitor since the graphics text was crisper.

The problem with reviewing such a large package (the manual runs to 230 pages) is knowing where to start, so I shall fall back on the King's advice in Alice in Wonderland "Begin at the beginning and go on till you come to the end. Then stop." However I'm afraid I shall be guilty of the sin of omission along the way.

The natural starting point is the data itself. This is stored in a POINTS file in which the horizontal and vertical values can be numbers or labels. The ability to use labels is fundamental to a good business graphics system since there are

numerous times when an axis should have names rather than numbers. These could be sales personnel, EEC countries or even types of fish – whatever your business requires.

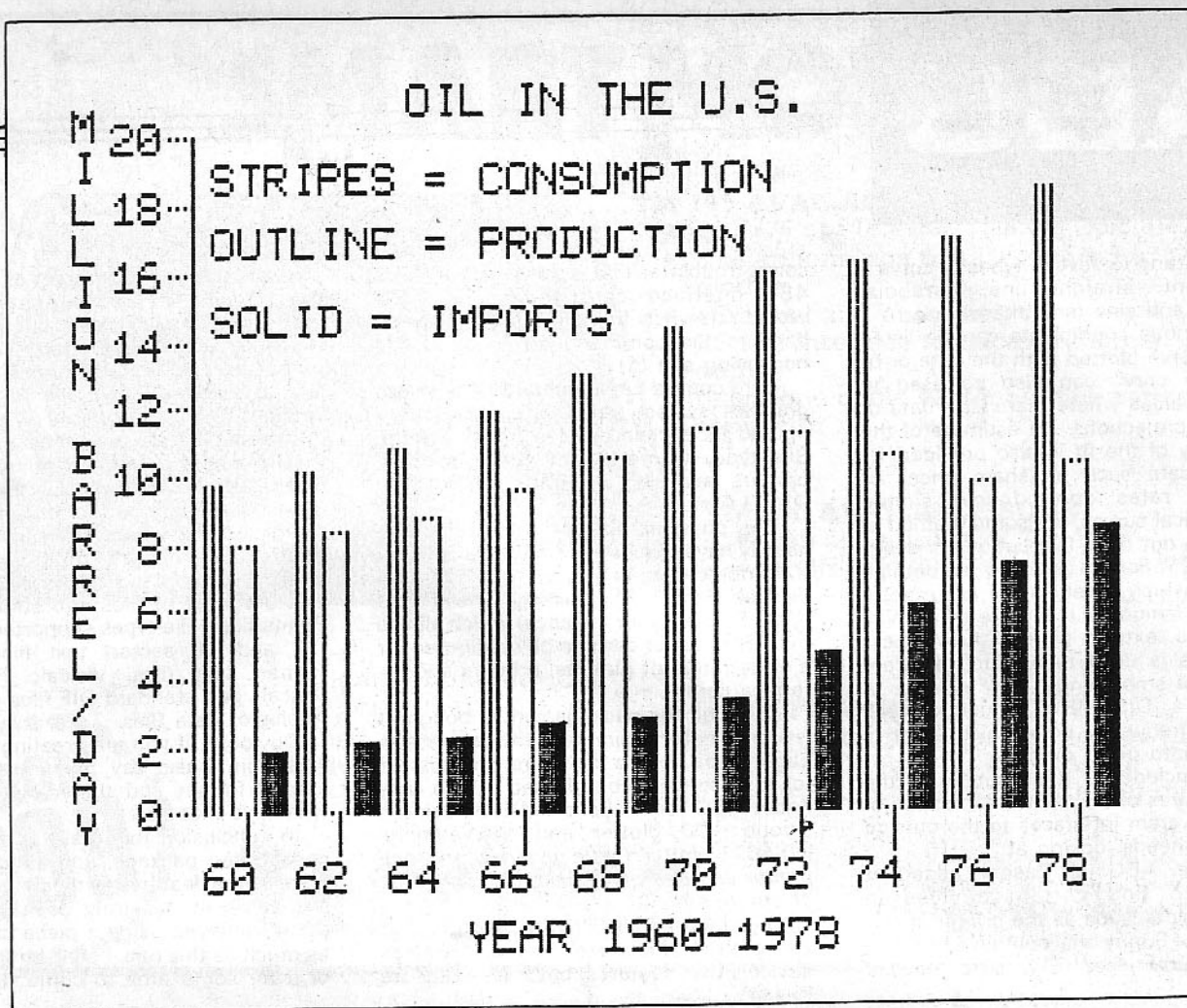
The points files can be extensively edited, added to, deleted etc. However labels carry a few restrictions that numbers don't. Most importantly, the various business and mathematical manipulation options provided can obviously only be applied to numerical values. The horizontal and vertical axis titles are also stored with the points, but that is the only extra information stored.

The weakest part of ABG is undoubtedly the way it treats titles and added text. Any graph worth its salt is going to have, in addition to the axes titles, a general title and probably some text defining certain elements of the graph. It is very easy to put them on the graph but they are not saved with the data.

The method for saving such extra information is somewhat clumsy. It involves creating a TAKE file – really just the set of instructions you typed to add the extra information. Running the TAKE file repeats the job you did.

There is a command to list many of the settings of the present graph – ranges, titles etc – and this gives you the backbone of such a TAKE file, but it doesn't provide information on all of its features, including where the basic data came from and any manipulations you





have performed, so it will need these editing into it.

To complicate matters further, the TAKE file often requires a large file space and a "Virtual File" needs to be created. This really is a needless piece of gobbledegook, and considering how effortlessly various other complex tasks are performed I am surprised that titles/text cause such heavy going.

One advantage of the TAKE file system however is that it can be used to create an automatic graphics presentation on the computer screen with the facility to program pauses in at key points.

Once data is typed in and saved, drawing a graph is simplicity itself. A bar chart requires only DRAW BAR, a pie chart DRAW PIE, a line graph DRAW LINE and for points shown by symbols, DRAW MARK.

Everything is done automatically, but many of the values can be explicitly overridden if you desire. The bar chart can have up to four bars for each horizontal label/number and, for instance, a command of DRAW BAR 3 will tell the system to leave room for two more data sets after the present one. The bars can be solid colour or drawn in outline only. I would have preferred a shading option since this is often more suitable for graph plotter output.

Pie charts can also be solid or outline with the ability to leave slices out, but you can't offset a slice from the centre to emphasize it. The slices are labelled auto-

matically and if there is room on outline slices the labels occur in the slice.

Line graphs can have a variety of colours or dashed line types and data can be plotted in eight different symbols. DRAW MARK is the command to plot points, but I found, by trying to guess the command, that DRAW POINTS seems to do the same thing, although it doesn't appear in the manual.

General text is put on the graph using FLOATING TITLES. These can be positioned on the screen by using a set of keys - I,J,K,M as in editing Basic - or by typing in a screen co-ordinate (0-100 in X and Y). I would have preferred the co-ordinates to be the ones on the axes rather than the nominal 0-100 since it is a little difficult judging exactly where the 0-100 co-ordinate is, especially since the graph doesn't cover the whole range.

The text can combine upper and lower case letters, but you cannot control the character size. A set of special symbols is available but plotters generally will not reproduce these, and one major omission I found was that try as I might I could not find a £ sign. This must count as a serious nuisance for the British user.

Horizontal and vertical grids can be imposed on the graph and lines can be drawn on the screen fairly simply. Enclosed areas can also be filled in automatically. This can be especially effective in creating overlaid data, but it will not look good on a plotter where drawing one colour on top of another generally is not

very satisfactory.

We now come to the additional functions ABG provides. The first is a simple set of statistical information on the present data set. The command LIST STATISTICS produces minimum and maximum values, the sum, mean, standard deviation and variance of the data.

The sum is particularly useful if you want to express things as a percentage. For the statistically minded it doesn't say whether the standard deviation is with respect to N or N-1, and I didn't have time to check which.

A number of mathematical manipulations can be carried out on the data, such as adding, subtracting, multiplying or dividing the data by set amounts or by the values in another file. This latter facility allows one to express data as a percentage of another set of numbers very easily. For instance, if you have SALES data and EXPENSES data over a period of time the following will produce a line graph of expenses as a percentage of sales:

```
LOAD EXPENSES
MULTIPLY 100 ... to make a %
DIVIDE BY SALES
DRAW LINE
```

It really is as easy as that.

It is common when looking for trends and projections to try to fit the data to some simple mathematical curve. ABG



# ABG

provides fitting to several types of curve – a constant, straight line, parabola, logarithm and sine (any phase).

The various coefficients can be listed and the curve plotted with the data or by itself. The curve can also be used to generate values where there isn't data or for future projections. An estimate of the uncertainty of the fit is also provided.

Often data such as share prices or exchange rates don't follow simple mathematical curves, but some method of smoothing out daily fluctuations is desirable. ABG offers a running smoothing function which uses past and future values to "smooth" the value of a given point. The extent of the future/past parameters is used to produce different amounts of smoothing.

Finally a DISTRIBUTE function will show the frequency with which a set of data falls into given ranges.

This concludes a rapid survey of the major features offered but the question of how the system interfaces to the outside world still needs looking at.

Because ABG is Pascal based it expects to find certain devices in particular slots, such as the printer in slot 1, while slot 3 may only contain a terminal or 80 column card. This latter caused

some trouble at first since I had an IEEE 488 interface card there and ABG wouldn't switch from the graphic screen back to the command screen with this occupying slot 3.

Hard copy is fundamental to a business graphics system, and ABG comes configured for certain devices. These are the Silentype, Qume Sprint 5/45 or 5/55 printers and HP 7225A/B or Houston DMP3,4 plotters.

The physical size of the graph can usually have two or three set values. However don't despair if you have different devices since a PIK package is available (Data Efficiency for instance) which allows the ABG master discs to be configured for a wide range of plotters, printers and interface cards.

Although the package can be bought it is a once only requirement and your dealer ought to be able to configure your system for a reasonable cost. I used PIK to configure ABG for an Epson MX100 printer, Strobe 100 plotter and a Watanabe WX4671 plotter. (*Note generally only one printer and one plotter can be configured at any one time.*)

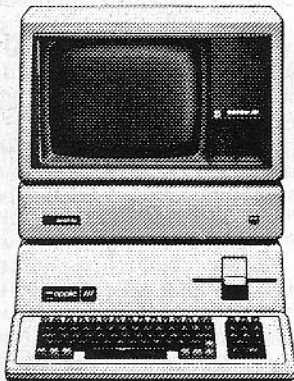
The output differs slightly depending on the plotter since use is made of the device's own features, but the results are

a reasonable representation of what is on the screen. To produce a graph on a plotter the drawing instructions must be repeated with the plotter allocated for output. Hence complex graphs should be set up using a TAKE file to make this simple.

The last feature to cover is the ability of ABG to utilise data files from a variety of other sources. I didn't try this out, but given the superb performance of the rest of the package I can't believe any difficulties would be met. An INTERCHANGE command allows ABG to convert a variety of files into standard ABG Points files. File types supported are DOS (13 and 16 sector) text files, Pascal/ Fortran text files, Visicalc Print files, Visicalc and standard DIF files and finally Appleplot data files.

Obviously if you are creating your own files from Basic say, they must be in a correct format and the manual explains what this must be.

In conclusion this really is a very well constructed package, and although there were some features it didn't offer or did awkwardly it has truly been some time since I enjoyed using a piece of software as much as this one. I shall be making use of it for some time to come. 🍏



## APPLE III



# ActionData Limited

# AD



# apple

TM Information Centre

Apple II/IIe Level 1 Dealer  
Apple III Level 1 Dealer  
National Account Support Dealer  
Lisa Information Centre

**The Old Court House**  
Hughenden Road  
High Wycombe  
Bucks HP13 5DT  
Telephone: 0494 444741







I WAS delighted to see the article by Mike Glover and Christopher Roper on the Epson MX80 printer. However, I am still in trouble, I can't get CTRL @ or CTRL A on the screen, and the alternatives given at the end of the article don't work either.

I am using an Apple II Plus with Applewriter II through an Epson Apple II interface kit type 2 (no. 8132 and 8131) to an Epson MX80FT3.

Any help you can give me will be very much appreciated.

Incidentally, without a lower case generator it is necessary to type ESC three times to get the capital letter for the proper code.

I recently bought Personal Finance Manager, by Software Dimensions Inc. This works well in all respects save one - it refuses to recognise that I have a printer in slot 1. Is there an easy fix for this?

Many thanks for an excellent magazine - Michael J. Owner, Dorking, Surrey.

● The reason that you were having difficulty in getting CTRL @ and CTRL A on the screen was that you were trying to do this directly from the [G]lossary (?=Define/\*=Purge) command line.

We should have made it clear that the glossary file must be created as a normal text file, SAVED and then LOADED as a glossary file.

It is true that for users without a shift key mod the ESC key will need to be pressed three times, the first two to put the flashing [ on the screen, the next one to make the command letter a capital.

Some programs check for the presence of a printer card by checking the "signature bytes" in locations \$CNO5 and \$CNO7 (where N is the slot number).

The 8132 card says that it is a printer card by having \$48 in both. Personal Finance Manager must do some further or non-standard checking as it determines that there is no printer available if the 8132 card is fitted. There are two cures for this:

- Change a byte on disc from 0 to 1.
- Configure the program using another card. It will then work with the 8132. Don't be tempted to reconfigure with the 8132 present or you will be back to square one!

I. Rochlitz, of Geneva, also rang to say that the articles had been most useful but he had some difficulty turning off underlining using ESC 0 in Applewriter IIe.

There is a small family of commands, such as underlining, that can be turned on and off. Generally this is done using CTRL A (Ascii 1) for "on" and CTRL @ (Ascii 0) for "off" although alternatively it may be 1 (Ascii 49) and 0 (Ascii 48).

Thus "underlining on" is normally ESC - CTRL A (don't forget the minus between the ESC and the CTRL A) and "underlining off" is ESC - CTRL @.

Applewriter IIe and Applewriter III do not permit the use of CTRL @, so for these

programs you should substitute 0 for CTRL @ and 1 for CTRL A.

Turning off underlining now becomes ESC - 0 (escape minus zero).

Some readers also had difficulty in understanding just how to determine what numbers to send the Epson to indicate how many bytes of bit image data was to follow.

When sending information using Ascii codes the largest value that can be transmitted is 255 (all bits set of an 8 bit number). This means that if we want to transmit a number that is larger, say 280, we have some difficulty.

To overcome this problem the information is sent in two bytes (a 16 bit number) the first one containing the high 8 bits and the second the low 8 bits.

While this is easy to understand for those familiar with binary and hexadecimal, it may still be confusing to those stalwarts who cling doggedly to decimal.

In decimal simply divide the number you wish to send by 256. The integer part of the answer is the high or first byte. The remainder is the low byte.

An example:  
LET HI = INT( 280/256 ) : REM HI = 1  
LET LO = 280 - (HI \* 256) : REM LO = 24

Mike Glover

## Bouquet for Apple '83

WE have found the articles in Windfall magazine both useful and interesting, and compliment you on the high standard of presentation.

This year we attended Apple '83 at Slough for the first time, and found it far more interesting than other exhibitions.

The demonstration of the Lisa was most impressive. We can't wait until the price drops! - M.J. Ballard, Woking, Surrey.

## Hi-res problem

I BOUGHT an Apple II Plus a year ago and can now program most of my scientific problems into Basic. My knowledge of machine language is still weak however.

I have a problem with the hi-res screens. How can I show page 1 while I draw on page 2, and viceversa, how can I

draw on page 1 while showing page 2?

I have tried many ways using the POKEs in the -16304 area. Any help would be most welcome. - Bachir Kassir, Beirut, Lebanon.

● A pointer in page zero at \$E6 (230) controls which page will be drawn on regardless of which is being shown. POKE 230,32 will cause page 1 to be drawn on and POKE 230,64 will cause page 2 to be drawn on. Thus to draw on page 2 while looking at page 1:

**HGR2:HGR: REM CLEAR BOTH PAGES**

**POKE -16302,0 : REM FULL SCREEN**

**POKE 230,64 : REM NOW DRAW ON**

**2**

Drawing routines here:  
**POKE -16299,0 : REM FLIP TO PAGE**

**2**

Notice the order of HGR2:HGR, which leaves us looking at a blank page 1 while drawing on 2. It is up to you what you really want to do at this point.

## Graphical instructions

AFTER copying Max Parrott's Apple Darts from the March 1983 Windfall and running it, I found that as soon as the instructions were printed, drawing of the dartboard followed with the instruction text getting intermingled with the graphics.

I checked through the listing several times but to no avail. Any suggestions?

I suppose I am a bit of an outsider to Apple users, since I have a Micro-Professor MPF II computer with 64k RAM.

Although it is fully compatible with Applesoft Basic, I have experienced a most frustrating problem of getting no reaction from the keyboard when playing the games out of past issues of Windfall.

Referring to Humpty Dumpty in the December 1982 edition, I found I had the same problem as Mr Oldacre (Think Tank, January 1983), in that the program hung after the first brick had fallen.

Following investigation, I found that in the DATA line 600 20,6 was typed instead of 206. After rectifying the mistake, the program ran perfectly except for the fact that I was unable to move the "little man" in the task of blowing the bricks back into position.

The same problem has occurred in Michael Hambley's "Scram" (in which I am constantly receiving BAD SUBSCRIPT error in 220) where the player is to use the I, J, K and M buttons.



I have read in previous issues of *Windfall* in the gamesmanship reviews of some games having the capability of letting the player change the games keys around to avoid deterioration from constant use.

Is it possible to do this in the games published in *Windfall* as it may help me in getting manoeuvrability?

I would appreciate any suggestions. Thanks for your excellent magazine. — **H.D. Orrick, Auckland, New Zealand.**

● Thank you for your letter regarding games and problems. Starting with your last point about varying the keys, I think that this is a good idea. I hope readers will take it up.

Working backwards through your queries:

The error message BAD SUBSCRIPT in Scram suggests that the variable I has gone over the limit for the dimension of the array MC.

However if MC has been filled okay in line 1220 and the error has not occurred for lines previous to 220 I don't see how this is possible.

This leads me to think that there is an error in the machine code routine at 768 and that Applesoft is picking up the wrong error message.

I can only suggest that you carefully check line 40120; check especially for periods in place of commas and missing digits.

In Humpty Dumpty, the little man's lack of motion suggests that the paddle buttons are not being read.

Check first that the minus sign is in front of the first two variables B0 and B1 of line 430.

Then check that these two variables are B0 (that is zero and not the letter O) and B1 (one and not L or I).

If this is okay check that the variables B0 and B1 are correctly typed in lines 70 and 80. If there is no problem here check that variable LI is typed as such in lines 70, 80 and 430.

Instruction text cannot become intermingled with the graphics in Darts on an Apple II unless you are using a hi-res character generator — that is if the two are truly intermingled.

If by intermingled you mean that you can see four lines of text at the bottom of the hi-res screen, then look at line 740 to see if you have typed the minus signs in front of the two POKEs and check that the numbers are correct.

If they are correct and you have these four lines of text then the Micro-Professor is not handling the soft switches in the same way as an Apple.

We played with a Micro-Professor some time ago and had a lot of trouble making certain programs work. The same programs worked perfectly on an Apple.

Furthermore, we could not get a single hi-res program to work although lo-res

and text seemed fine.

Perhaps readers with Micro-Professors will tell us of their experiences. We believe that a new model (Micro-Professor model MPF III?) is becoming available, but what the differences are and whether old versions will be up-gradable or not we don't know.

Incidentally in July Apple was awarded an injunction against Sirtel UK prohibiting the company from importing, advertising, selling or otherwise dealing in the Micro-Professor II. The injunction is binding until the full trial of the action.

**Max Parrott**

## Converting CALLS

I AM in the process of writing a program for my children which displays a picture and then asks the operator to spell the name of the displayed object.

To carry this out I am using SAM the Software Automatic Mouth, to speak to the child and Graphics Magician to draw and display the pictures.

My problem is this — the machine language program for SAM occupies memory space \$9600 to \$58A0 and is called by CALL 38131.

The picture draw routine from Graphics Magician occupies memory space \$8C00 and is called by POKEing 36352 and 36353 and then CALLing 36400. If it is an object to be placed over a picture the POKEs are 36354, 36355 and 36356 and the CALL is 36361.

The question is how can I use my 16k memory card to hold these two routines and how do I convert the CALLs? I would appreciate any help.

I read the articles about using the 16k card for "Help" pages, but it doesn't help me, so explanations in words of one syllable would be appreciated! — **Peter S. Ballard, Hong Kong.**

● We have no first hand knowledge of these programs, so therefore are unable to give you the precise instructions to accomplish your object.

Many commercial programs have an element of protection to frustrate you in your efforts. Perhaps a reader who is familiar with these can tell us how to do it.

## Club plans

IN preparation for organising Longfellows first elementary (age 6-12) Apple computer club I decided to write and ask for ideas and/or suggestions.

Perhaps we could begin some penpal type activities across the ocean. If anyone is interested, they can contact my home. — **Lila Stoller, 2766 N. Prospect, Colo. Spgs., Colo., 80907, USA.**

## Tidal predictions

AS a retired naval officer and sometime (1975) navigation instructor I was interested to read P.J. Henderson's letter on tidal prediction (*Windfall*, September 1983). Though I have no off-the-shelf solution, perhaps I could point him in the right direction.

Tides are raised by the gravitational force of sun and moon. These bodies move around each other in orbits which are neither circular nor coplanar and have widely differing periods.

However prediction of the tide-raising force at a given time and place, though extremely laborious, is relatively simple — ideal computer fodder.

The pattern of tide raising forces can be predicted by summation of all the cyclic gravitational influences ranging from the 12½ hours (half a lunar day), which dominates the tides we experience, through to such esoterics as the Chandler wobble with a period of some 28,000 years.

Some 400 separate cycles can be identified and predicted, though at most only 60 or so are significant, and often a dozen will do.

Tidal prediction essentially comprises the addition of a series of sine waves with amplitudes, periods and phase relationships determined from astronomical observations.

Given this pattern, the tides at a particular place are found by application of a phase lag, since the oceans do not respond instantaneously, and a real tidal amplitude which will allow for local effects such as the funnelling in the Severn Estuary.

This permits prediction of the height of tide at any time, or conversely the time at which a particular height of tide will be reached.

These local corrections can only be obtained from a period of tidal measurements on the spot. Depending on the precision required, a month's data may suffice, but a full year will be much better.

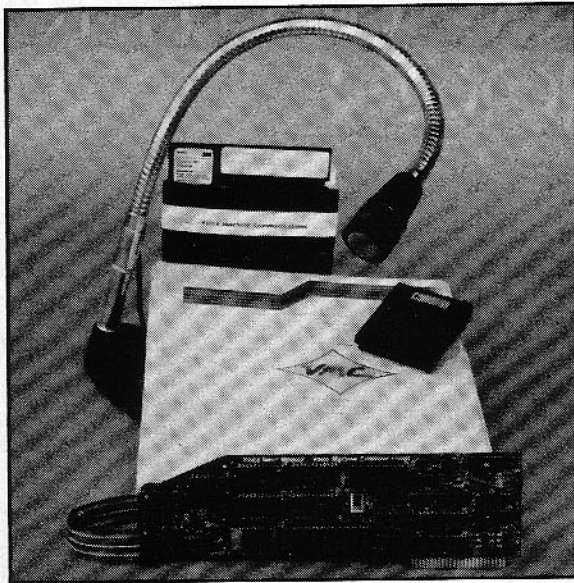
The necessary constants and data to accomplish these computations will be found, if memory serves, in the Admiralty Tidal Handbook and the Admiralty Manual of Tides.

These can be obtained from local library, ship's chandler or perhaps harbourmaster; but possibly the best solution would be to write to The Hydrographer of the Navy, Taunton, for expert advice.

Anyone wanting a slightly more detailed — and much more cogent — account is referred to the Admiralty Manual of Navigation Volume I, available as above. — **S.T. Howes, Kilbarchan, Renfrewshire.**



# VOICE MACHINE COMMUNICATIONS, INC.



The VIM converts spoken words to commands or data for your application programs. The Voice Input Module has unexcelled spoken word recognition accuracy at an unmatched price.

## DESCRIPTION:

### VIM FOR APPLE II CONTAINS:

#### Voice Input Module 2020C with:

- 16 channel audio spectrum analyzer
- 6803 high speed microcomputer
- 8K Bytes of RAM, 4K Bytes of ROM

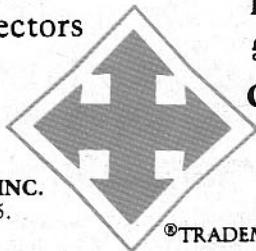
#### Voice Utility Diskette with:

- Vocabulary builder/editor
- Prompting vocabulary trainer
- Recognition software
- Vocabulary tester
- Merge vocabularies
- Change vocabulary
- 11 Application vocabularies

**Microphone, Users Manual** and all necessary cables and connectors

Deluxe model & options available  
VISA - MASTERCARD

VOICE MACHINE COMMUNICATIONS, INC.  
1000 S. Grand Avenue, Santa Ana, CA 92705.  
Phone: (714) 639 6150



®TRADEMARK

Announcing  
SPEECH RECOGNITION

# VOICE INPUT MODULE

FOR APPLE II® & APPLE II-E®

## FEATURES:

- No application programming necessary
- Near perfect recognition 98%+
- Unlimited vocabulary using eighty word/phrase subsets
- Recognizes anybody's voice easily
- Multi-lingual recognition
- Allows simultaneous input of voice and keyboard

## APPLICATIONS:

The VIM is designed to add voice input to any existing Apple II application:

- Graphics
- Word Processing
- Data Input and Retrieval
- Education
- Business
- Industrial Automation
- Programming
- Testing & Measurement
- Control Systems
- Games & Entertainment
- Handicapped — Environmental Control

£899 inc VAT

## CASCADE GRAPHICS DEVELOPMENT

185 Lower Richmond Road  
Richmond, Surrey  
Phone: 01 878 7661



# Apple opens door to independence

IN 1978 I wrote an article for an American journal called "Rehabilitation World", the sub-title of which was: "Science has gone mad when it costs thousands of pounds or dollars to switch on a light."

I was, of course, referring to complex and expensive electronic equipment being invented and marketed with disabled people in mind, when simple mechanical or electro-mechanical devices were already available and only required minor adaptations for the most severely disabled person to use.

I became severely disabled in 1960, at the age of 23, having contracted an unknown viral infection of the spinal cord, the result of which was to leave me paralysed below the shoulders. Being formally a self-employed plant machinery engineer and essentially a very active and practical person, it took me a considerable time to physically as well as psychologically adapt to my new mode of life.

Eventually, and to cut a very long story short, I developed simple adaptations to my telephone, tape-recorder and electric typewriter, which I was able to control with sticks held in the teeth. With the typewriter, for example, I can type at 30 words a minute, put paper, carbon-copies and envelopes in and out of the machine, all with mouth-sticks.

These achievements opened doors to relative independence and enabled me to

---

By MICHAEL A. ROGERS

---

write. To date I have been fortunate in having one book and several articles published.

The physical result of typing millions of words like a woodpecker over the years, driving my chin controlled electric wheelchair, painting in oils with brushes held in my teeth, plus the thousand and one other movements made with the head and neck during the course of a day has,

to put it mildly, given me a "pain in the neck." Therefore to try and reduce the degree of head and neck activity, as medically recommended, a good friend Roger Jefcoate suggested I should try using a word processor.

Disability tends to arrest time. Often, as in my case, initially one spends considerable time in hospital, the result inevitably being that it's difficult if not impossible to keep up with changes in technology. This is particularly so in the computer world where, over the last decade, development has been staggering. I was, therefore, apprehensive and slightly frightened by the prospect of what I might be letting myself into.

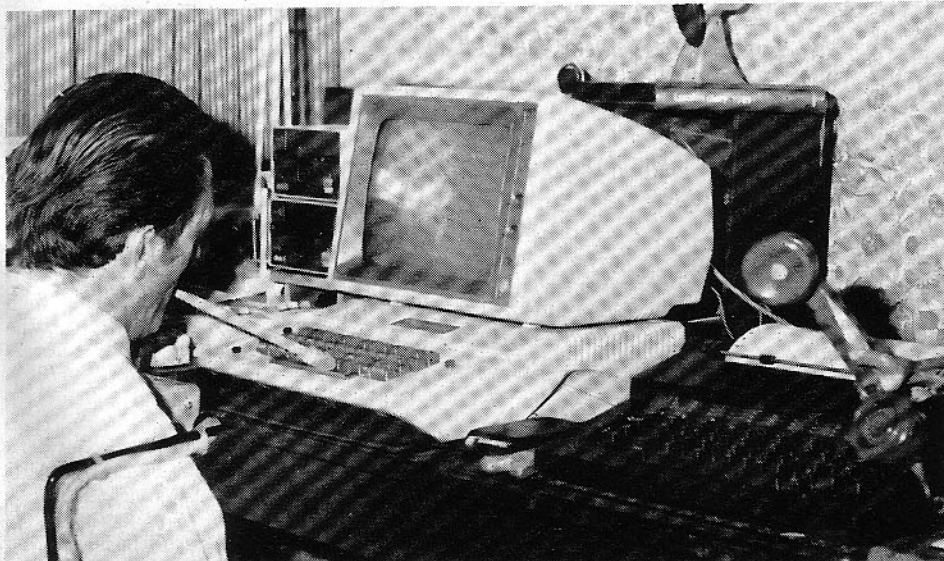
Roger Jefcoate, who lives in Mursley, Bucks, is as far as I am aware the country's only independent freelance assessor for aids and equipment for the disabled. He knows his stuff and recommended I should try an Apple II.

"Fantastic," I replied, "but where would I get the cash from to buy that lot?" "I will launch an appeal for you," Roger replied.

I had visions of waiting a couple of years or longer. But no, within a matter of months he telephoned suggesting I should be looking around to see what was available and where I could get the best terms.

This is where true friends are essential. Dennis Mimmack, of Denmack Business Systems in Radlett, agreed to provide the equipment at cost price. Dennis recommended I should have the Apple II with twin disc drives, Philips amber monitor and the Format 80 word processing program.

I discovered the perfect printer for my needs, the Olivetti Praxis 35, which would



Michael Rogers word-processing on his Apple



# DISABLED

also double as an electric typewriter and provide me with a choice of daisywheel type faces.

Not wishing to cause embarrassment, the money was forthcoming from various sources in record time, and much was given at considerable discount. I will never be able to thank everyone concerned enough.

On arrival it was necessary to have the Apple modified so that I could fully control the equipment with the mouth-stick held between my teeth. Roger Jefcoate again came to the rescue. He had previously introduced me to The Professional Workshops at Neath Hill, Milton Keynes, where I had seen the Apple II demonstrated and tried it for myself.

This centre, run by the Spastics Society, is an Apple dealer in its own right. They employ a number of disabled as well as able-bodied people, providing a complete computer service from programming to purchase, from conversions to consultations. Their technical wizard, Peter Deacon, swiftly converted my machine by the simple addition of four extra switches.

Readers familiar with the Format 80

program, will know that it is necessary to switch to an 80 column line. This is simply done by depressing the additional switch on the right hand side. Apple might well profit by including such a modification as standard.

Certain instructions within the same program require more than one key to be depressed simultaneously. To allow for this a further three push-down switches have been added to the left-hand side of the keyboard, giving lock-down of the SHIFT key, CONTROL key and REPEAT key. Don't ask me to explain how to wire these switches up. That's for the experts.

With these minor modifications I now have complete control of the Apple. I'm even able to change discs with my mouthsticks. That has to be seen to be believed.

And the benefits? Quite fantastic. No more re-typing, the formatting or correcting facility is out of this world. And the final typing - simply perfect.

Not only is the word processor providing tremendous help in writing another book, I'm able to use it for all domestic correspondence, preparation of lecture notes and articles. Now that I'm confident with the word processor programme, the

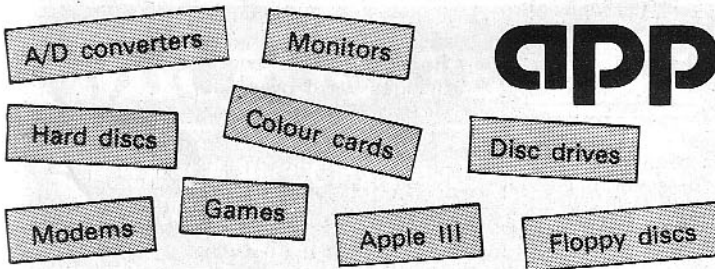
next stage is to explore the use of the Apple as a computer.

Drawbacks? Yes, just one. The keys of the Apple II are not as positive or as light as those on the electric typewriter. Any ideas?

However I must now change my ideas concerning "technology." If expensive and elaborate equipment can improve the quality of life for the disabled, then it should be used. At the National Spinal Injuries Centre at Stoke Mandeville hospital, doctors are today seeing patients that have been paralysed a number of years returning with problems related to excessive wear to joints due to over activity.

Consequently the old ideas of getting disabled people to literally do everything that was physically possible for themselves regardless of the consequences will, I'm sure, change as technology becomes available to do the work without the same degree of physical stress and strain.

I would again, through this article, like to publicly express my most sincere thanks to all those who have so generously helped me to bite my own Apple.



## apple classifieds

**INTEGER CARD.** Genuine Apple product. £45. As new. Phone 0279 33711.

**APPLE DISC II** with controller plus Database and Viscalc for Apple II Europlus, never been used £225. Phone St. Albans 68639.

**80 COLUMN CARD** Apple IIe, boxed, as new. £85, call 402 5061, evenings, weekends.

**SATURN 128K RAMCARD**, hardly used, c/w manuals, software £250 o.n.o. 01-592 9039.

**APPLE II 48K Autostart ROM**, manuals, Zenith green screen monitor. £500. Evenings 061-797 5717.

**GAMES FOR SALE** as new including Jawbreaker, Cyber Strike, Epoch, Autobahn, Phantoms Five, Space Eggs. £7 each. Phone 0371-850254 (evenings).

**APPLE II EUROPLUS** 64k twin disc DOS 3.3, printer, colour monitor, Pascal language card, games, business software, manuals, accessories £2,200 o.n.o. 061-225 7512.

**CORVUS IOMB** hard disc with Constellation multiplexer for 2 Apples, excellent working order. Phone Tony Pearce 01-221 1164 (day) 01-675 3084 (evenings).

**ITT 2020 48k for sale.** Complete with UHF modulator, games paddles, cassette recorder and manuals. £330. Telephone Allen Harris, Bristol (0272) 47865 (evenings).

**VIDEX 80-COL CARD**, new, £95. Videx Enhancer II, new, £45. Synergiser Z80 card, Smarterm 80-col card, Addram 16k card, Supercalc £250. Tel: 061-427 2315.

**DISC III** £170. Sup-r-term 80 col £90. Integer card £50. Mountain A/D and D/A. £110. Romsey 390296.

### apple classifieds

- Classified ads can only be accepted from private readers, not companies.
- The cost is 20p per word, with a minimum of 10 words prepaid.
- Your ad will be printed in the next available issue of Windfall.
- Your accompanying cheque should be made payable to Windfall.
- Ads can only be accepted on this form (or a photocopy of the form).
- There is no maximum to the number of words you include in your ad.
- Ads too long for the form should continue on a separate sheet of paper.
- Ensure your phone number or address is included in the ad.

<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	10 words £2.00
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	15 words £3.00
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	20 words £4.00
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	25 words £5.00
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	30 words £6.00

Cheque enclosed for £

Name \_\_\_\_\_ Address \_\_\_\_\_

POST TO: Apple Classifieds, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.