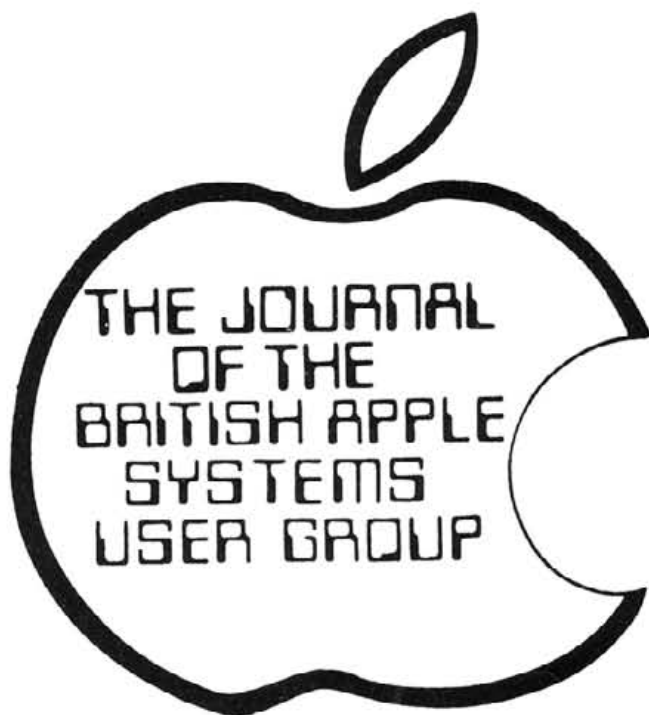


**hard core**



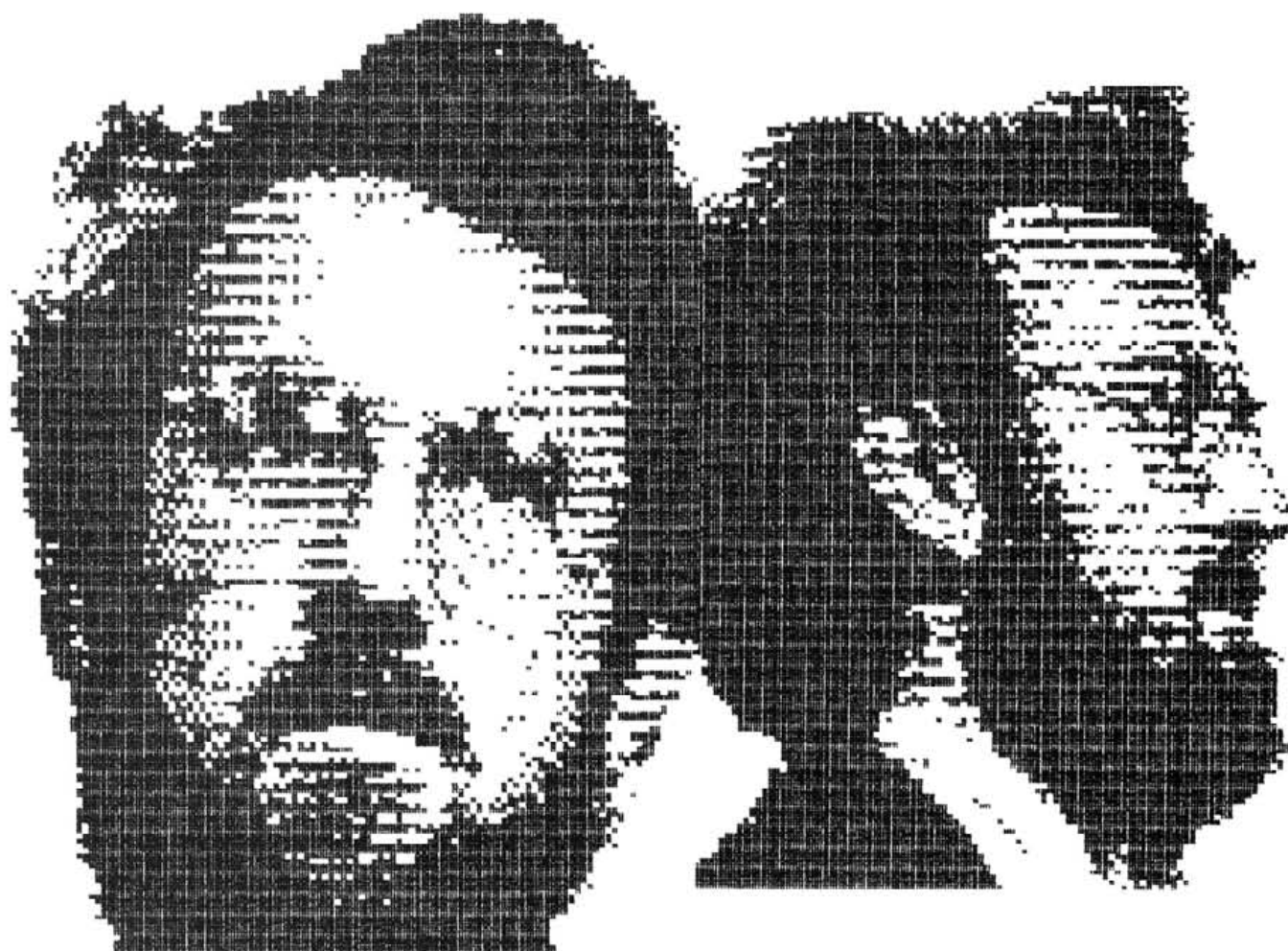
---

**OCTOBER 1981**

**£1**

**VOLUME 1 N°5**

---





For everything

**apple .. ring****Lux Computer Services Ltd.**

108 THE PARADE HIGH STREET WATFORD WD1 2AW  
 Telephone: Watford (0923) 29513

- Business Software • Professional Courses • Pascal •
- Recreational Software • Interfacing • Cromemco Systems •
- Level 1 Service Centre • Utilities • Colour Monitors •
- Supplies • Accessories • Enthusiasm • Open Tues - Sat •

**BLUE CHIP SOFTWARE**

10 Huson Close London NW3 3JW Tel: 01-722 8827

**SUPER TRACE by Ian Trackman**

£27.50

Super Trace is a high-speed 6502 machine-code utility routine for use in debugging Applesoft programs. As each of your program statements is executed, it is displayed at the bottom of the screen, together with the values of any variable types or functions that you may have pre-selected.

- \* Programs can be written, loaded and edited without affecting Super Trace
- \* Does not interfere with DOS
- \* Can be invoked and disconnected as often as required during program run-time
- \* Simple to use. Run-time control by single keystrokes
- \* Optional display of all statements or only statements containing selected variable types or functions
- \* Variables highlighted in inverse display and their values displayed
- \* Normal screen display not affected
- \* 6 run-time speed settings plus instantaneous stop and single-step
- \* Optional review of previous 10 program statements without destruction of screen display

**SYMDIS by Ian Trackman**

£22.50

Symdis will take machine code from anywhere in ROM or RAM (except zero-page) and produce from it a disassembled source code with all relative branches and internal absolute address references converted to alpha-numeric labels. Every affected operand will be replaced by an address-related symbol and all target addresses will be labelled with their correct symbols. Sections of the source file can be produced as text or data with the hexadecimal or ASCII values of the object code displayed. The source file, in standard ASCII format, can be examined on the screen or saved directly to disk. The resulting source file can be loaded into any Assembler/Text Editor which handles standard ASCII format source-files (such as the DOS Toolkit). Alternatively, by using the conversion program provided, the source file can be presented to an Assembler/Editor in the form of a binary-file block memory dump.

All Blue Chip Software programs are supplied with comprehensive instructions and require 48K of memory, one disk drive with DOS 3.3 and (except for Symdis) Applesoft / Palsoft in ROM.

Dealer enquiries welcome.

# hardcore

THE JOURNAL OF  
THE BRITISH APPLE SYSTEMS USER GROUP  
P. O. BOX 174 WATFORD WD2 6NF

EDITED BY DAVID BOLTON

## BRITISH APPLE SYSTEMS USER GROUP

### COMMITTEE

FRANK KAY	Chairman
JOHN SHARP	Secretary
DAVID BOLTON	Treasurer
EDDIE PAYNE	Events Organiser
JOHN RODGER	Librarian/ Information Officer
JOHN ROGERS	Software Librarian
TONY WILLIAMS	Membership Secretary

### BANKERS

National Westminster Bank Ltd.,  
Brompton Square, London SW3 1HH.

MEMBER OF THE  
INTERNATIONAL APPLE CORE

COPYRIGHT (C) - The contents of this journal are copyright of the British Apple Systems User Group and / or the respective authors. However permission will normally be granted for non-commercial reproduction by user groups affiliated to the International Apple Core, provided the author and source are properly credited.

The opinions and views expressed are those of the various contributors, and are not necessarily supported by the British Apple Systems User Group.

## CONTENTS

4 COURSES	
4 LIBRARY	
5 EDITORIAL	
5 BEGINNER'S PAGES	John Sharp
EDUCATION	
7 MICROCOMPUTERS IN PRIMARY EDUCATION	Roy Garland
8 SUNDAY SCHOOL	Tony Williams
10 BASIC TUTOR	
10 ARISTOTLE'S APPLE	Norah Arnold
12 CHAIRMAN'S CORNER	Frank Kay
13 LOCAL GROUPS	
14 SHAPE MANAGER	A.P.Mullan
16 SHAKE HANDS WITH YOUR APPLE	Ian Pawson
18 VISICALC 3.3	Ian Trackman
21 HOW APPLESOFT STORES ITS VARIABLES	Peter Wicks
22 VIDEX VIEW	Brian Cowell
23 DOUBLEDOS	John Sharp
24 SOFTWARE LIBRARY	John Rogers
26 SECRETARIAL SECTOR	
27 ITT HIGH-RESOLUTION GRAPHICS INTERNALS	Ian Trackman
31 BOOK REVIEWS	Chris Murphy, Ian Trackman, Cliff Wooton,
35 IT'S A MYSTERY	?
36 READERS LETTERS	
41 MORE ON PROGRAM POINTERS	John Rodger
42 ENHANCED MODE WITH THE EPSON	John Sharp
44 EAMON REPORT	David Row
45 THE ANDROMEDA RAM BOARD AND THE ITT	Jeremy Ensor
45 DO YOU KNOW THAT?	
46 PIPPIN'S PAGE	Vernon Quaintance

## COURSES

by Eddie Payne  
(BASUG Events Organizer)

Those of you who were with us around six months ago will remember our abortive attempt at organising a course on PASCAL at Brunel University. Unfortunately, things went wrong both with the timing (it clashed with a Bank Holiday) and with the fact that we only had about a quarter of the members that we have now. Because we could not raise the required number of people to make it worthwhile the whole thing fell through. Everyone's money was returned and it was all a great pity.

This time we will do it properly and the problem will be too many not too few applicants. The course was to have been run by Les Johnson who has said that he is willing to try again in the new year.

It is planned to cover two full weekends plus probably one other day (Saturday or Sunday, five days in all) and move from the very beginners through to a full understanding. If your knowledge of Pascal is nil don't be put off, I'll be on the course and I want somebody else who doesn't know anything to keep up with me.

As soon as Les Johnson returns from America I will confirm dates, times, etc. In the meantime please let me know if you are interested. The location will be Brunel University, Uxbridge. If you are likely to require accommodation, please let me know.

I will be organising some one day courses to be held on Sundays. The subjects we have compiled so far :-

- Basic Basics (that's for me)
- Applesoft.
- Machine code
- Business software.
- Using the Z 80 Card.

The location of these courses is likely to be Milton Keynes. This has good access from the M1 and we hope we would see some members from the northern groups as well.

Let me know of any other things you feel would make a good subject for a day's teach-in - I'll try and organise it.

To do anything along the above lines I need your help, both with some idea of what you want to happen, and also if you feel you would be able to give a short presentation talk or help out with one of these subjects?

Looking forward to some positive feedback.

## THE LIBRARY

There is a library which consists of a number of books and articles related to the APPLE and to programming in general. These are available by post, but at a cost which will have to be born by the member as the membership fee could not hope to cover a large amount of postage for such items. The library is not only growing by donations, and in some cases by purchase, but also by exchange with other USER GROUPS. There are also copies of APPLE ORCHARD, MICRO 6502 (from Aug 81). It is not possible to print a list here, it would be out of date by the time you received your copy. The way to get a list, which will contain weights of each item so that you can decide whether you wish to send first or second class is outlined below. Such a service will require a reasonable turnaround time by the borrowers if long queues are not to develop. It is not going to be feasible to supply books in general overseas because of the delays in posting and the heavy cost.

The one exception to this must be 'APNOTES'. There is not room in this issue a complete list. At the moment copies are available only as borrowable copies. The photocopy offer in HARDCORE 1 is withdrawn, unless any member comes up with a cheap offer to the members. (We have over 700 members now and the physical copying is a task in itself).

TO FIND OUT WHAT IS IN THE LIBRARY and the latest APNOTES, write to the librarian (John Rodger - not John Rogers, he is the Software librarian) at PO BOX 174 Watford WD2 6NF, ENCLOSING A STAMPED ADDRESSED ENVELOPE. If this is not sent your letter WILL BE IGNORED. He will then send you a list which will tell you how much postage you will need to send also. If there is anything you think needs to be there, send your suggestions.

## **POPULAR COMPUTING**

**Chas Andres**

CREATIVE COMPUTING





EDITORIALFIRST YEAR

The first year of BASUG is coming to a close. We have been very successful if the rate of growth is a measure. From the 40 people who came to the inaugural meeting we have now grown to over 740 and the membership stretches to the other side of the world.

The workload has reached gigantic proportions as well, but we have a willing band ready to soldier on to bigger and better things. The important thing now is to form your own local group and get to meet other members.

Not everything has been set up that we hoped for, but we haven't had any complaints yet! The new year promises something big. Keep your weekends free!

The world of APPLE has changed as well. Apple has come to Cork and Microsense has become APPLE(UK) Ltd (we think, but they are not broadcasting the name very loudly). Those of us who decided to mortgage our houses to buy the best of the micros a year or more ago, have seen that we made the right decision. Although prices have come down in real terms, RAM has really fallen in price.

CHRISTMAS IS COMING

This issue has followed on fairly rapidly after no.4, because of the little bit of trouble we had. It is planned to have no.6 out early in December, so that you can look through the Adverts and see what goodies you want your family and friends to give you. This will give us a rest before we start again with the new flood of post in the new year. How do we know it is going to be a flood? Well, you will all be rushing to renew your subscriptions won't you? But the Christmas issue still needs to be written first and that is where you come in. We need your letters, ideas, novelty programs, articles and above all advertising. A bumper 60 page issue is planned. It will not happen if you don't supply the copy. It is also imperative to find the advertising. This is where you can help. If you are a dealer, support the magazine with even a quarter page. If you are a member and have a dealer who could advertise, persuade him to do so. The circulation is increasing and the response to advertisements is well worth the £15 per quarter page or £27.50 per half or if you really want to make an impact £50 for a full page.

AN APOLOGY

In the rush to get the magazine out we left out a credit to Norah Arnold for the design for the front cover; John Rodger helped as well but since there have been a few comments about sexist editorials I thought I had better put ladies first. The logo was originally done for BASUG Disk of the month (disk 19 in fact) and looks really good in colour. It has even been suggested we have some printed as posters.

This month's cover is a HI-RES dump of the secretary, in two of his many guises, which was converted to the APPLE by means of a Video camera.

BEGINNER'S PAGES

By John Sharp

WHEN I'M CALLING YOU

When we dealt with PEEKS and POKES, we also mentioned that sometimes you might POKE in a machine language program, and then CALL it. The word CALL suggests that you can call on something that is already there. There are some routines in the APPLE that are waiting to be used which are not commands in BASIC. In fact in INTEGER BASIC there is a need to carry out certain operations for which there is no command, even though it might be there in APPLESOFT. One such case is the HOME command in APPLESOFT, which clears the screen and sets the cursor to the top left of the screen. If you are in INTEGER BASIC you have no alternative than to use CALL -936. This will work in APPLESOFT as well and you sometimes still see it in APPLESOFT listings where an INTEGER programmer has not lost the habit when moving over to APPLESOFT. There are two other CALLS which are very useful for which there is no command in either INTEGER or APPLESOFT. One clears to the end of a line (the equivalent of ESC E if you are in the editing mode), and the other clears to the end of the page (the equivalent of the ESC F in editing mode). But first let's look at an example of why you may need to use them.

Suppose you have an INPUT statement which you then check to see if a number or string has been input correctly, and if not repeat the input until it is right. You might write a program thus:-

```
10 TEXT:HOME
20 PRINT "IF YOU HAVE TEN CATS AND THREE
   DOGS "
30 INPUT "HOW MANY ANIMALS DO YOU HAVE
   ALL TOGETHER ";A$
40 IF A$ <> "13" THEN GOTO 30
```

If you run this program and the person answering types in the wrong answer, another input line will come up on the screen. Eventually the PRINTed statement in line 20 will scroll off the top of the screen and the question will disappear. there will be a list of wrong answers with the question before them. We could rewrite the program, so that by using VTAB we write the question (i.e. line 30) on the same line of the screen each time, overwriting the last one. So insert a line

```
25 VTAB 10
```

and alter line 40 to :-

```
40 IF A$ <> "13" THEN GOTO 25
```

Now if you run the program there will not be scrolling and lots of answers, if a wrong result is typed in. However, if the first answer was 1111, when you come to type in the second answer, the 1111 is still sitting on the screen, although the cursor is waiting on the first 1. In order to clear the line first you could use a PRINT statement, but this would produce complications in VTABbing. However, if we put in a CALL -868 in line 30 before we carry on the problem is solved. Line 30 would now read:-

```
30 CALL -868: INPUT "HOW MANY ANIMALS DO YOU HAVE ALL TOGETHER ";A$
```

Try the program as outlined making the changes and see the result. I much prefer this sort of program instead of a screenful of wrong answers.

CALL -958 is useful in the same way. You might have a screen which is full of text but which has some instructions at the top, and you want to print only on the bottom half of the screen. There are two ways to do this. One is to alter the TEXT WINDOW.

If you POKE 34,T where T is the number of lines down you do not want to print you can only write below this line. So the following short program will print four lines then only print below those lines. If it is necessary to clear the screen below the line, the HOME command will do this.

```
10 TEXT:HOME
20 FOR N= 1 TO 4
30 PRINT "THIS IS LINE ";N
40 NEXT N
50 POKE 34,4
60 FOR M = 1 TO 50 :PRINT "A LOAD OF RUBBISH AND YET MORE RUBBISH ";NEXT
70 HOME
```

Note that line 10 has a TEXT:HOME to clear the screen, whereas line 70 only has HOME. If you put a TEXT command in a line it will automatically remove any TEXT WINDOW settings you may have. This is why it is often a good idea to use it as the first line of your program, so that anything left over from another program does not cause problems. To help others a TEXT:HOME in the last line when you exit your program, will help you if you are running another program where it is not the first line.

BUT .....suppose you wanted to clear the screen starting at a middle of a line. You now need to use a CALL, CALL -958. This is a less likely problem, but could occur if you want to delete the text starting from the beginning of a particular sentence. It is so unusual that I cannot think of an example program. It will allow you to clear the screen and still keep the scrolling, whereas the text window setting will keep above the line you have designated until you alter it.

If you want to print on the left of the screen or just in one corner, you can set the sides and the bottom of the text window with other POKES.

Look in the APPLESOFT reference Manual for more details.

### DIFFERENT CATALOGS

When I wrote last time about using the cursor to run over the catalog title to run it, I assumed many people were as lazy as me. This doesn't appear to be so. So let me extend it to let you put inverse or even flashing catalog titles.

To recap first. Let me go over using the method of moving the cursor over a title to run a program.

Type CATALOG and get the catalog up on the screen. If you have a very big catalog, you will not be able to use the method if the program has disappeared off the top of the screen. Press the ESC key and then the I key until you are on the line where the program name sits. Now type R (twice so that first you break the control over the cursor, and then print an R) then U and N. You will probably find that a number still exists from the file length, so use the space bar to type over with a blank. Now using the right arrow, run over the rest of the name of the file. Press return and you should be away.

When you carry out the following sequence of events, to give you flashing or inverse titles, this is the only way (or the modification I shall describe) of running the program. So if you are sitting comfortably in front of your machine, I'll begin.

Suppose we want to save a program under the name "TESTING INVERSE". Type the following in immediate mode:-

```
PRINT "SAVE ;;INVERSE: PRINT "TESTING INVERSE": NORMAL
```

The words

```
SAVE TESTING INVERSE
```

will come up, but with the last two in inverse of course. Now press the ESC key and as with the method outlined for running the CATALOG program, copy over this line with the right arrow once you have positioned the cursor on the S of SAVE. Now catalog the disk and there will be a program called TESTING INVERSE written up in inverse. To run the program, use either the method of writing a line (but with LOAD or RUN instead of SAVE) or copy over the CATALOG heading as described. If you just type RUN TESTING INVERSE then FILE NOT FOUND will come up because the comparison does not match.

To FLASH the name just write FLASH instead of INVERSE when writing in immediate mode.

As a last note, you may like to try the following CALL to see what it does :-

```
CALL -1184.
```

# ..... EDUCATION

## Microcomputers and Primary Education

by Roy Garland

It is difficult to escape from the publicity that microcomputers are receiving in the media. Fellow club members will be aware of the manner in which machines are often treated; they are accorded an air of mystery and there is an implication that only the brighter members of MENSA will ever be able to understand how to operate or use them. This image is somewhat remote from the truth but what is perhaps nearer to reality is that they are going to play an increasingly important part in our lives. Schools have a part to play in removing the veils of mystery and giving their pupils information enabling them to form judgements and discuss issues in an informed and unprejudiced manner.

My professional work involves me both in teaching in primary schools and in training primary school teachers both at a pre and inservice level. My own beliefs regarding the contributions that schools should be making to a debate on the use and misuses of micros and my own personal fascination with the machines has enabled me to develop ideas and practical thinking on the topic. Many readers may agree that schools should be doing something in what might vaguely be called the field of computer literacy but may have reservations about the involvement of comparatively young primary school children: aren't they too immature? Shouldn't the schools be concentrating on the 3R's? What of the costs involved in buying a micro at a time when other resources are scarce? I simply disagree! I have been associated with primary education in one form or another for over thirty years seeing fads come and go, but in my judgement these machines are of a qualitatively different order to some of the fashions of past eras. They make available to young children powerful ideas regarding relationships and logical thinking; as anyone who has a young child will testify, they have an enormous power to motivate and fascinate children and, of course, they can be used to practise basic skills. In fact it is a untruth enshrined in myth that primary schools neglect the so called "basics". Research evidence has shown that schools give the highest priority to this type of work and that in some cases their curriculum would be improved by a general broadening.

The business of cost is important; I doubt very much whether primary schools would be able to afford Apple systems; far more likely they will go

for lower cost units such as the Acorn, its off-spring the Proton, or the new VIC-20. It might seem a wry comment on our financial priorities of the moment that individual club members are able to afford at a personal level what their local primary school cannot. Primary schools are not extravagant institutions in spite of what we might read; something in the order of £1 per month per child is all that is allowed for materials and equipment.

A problem of substantial proportion is going to be the training of primary school teachers to use the machines. Teachers have to be introduced to the micro in a professionally positive and relevant manner; they need to have the opportunity to explore for themselves the possibilities that are on offer. If not then purchase of a micro by a school will be a waste of money. In fact I do not think that at this time I could unreservedly advise a school to purchase.

Perhaps the picture that I have unveiled is too gloomy; I think not. Although I am sure that there is an important place for the micro in the primary school classroom the obstacles in placing it there are considerable. For example, primary schools are staffed in the main by women yet cultural factors seem to place males at the keyboard; how many lady members are there I wonder in BASUG? (Ed. At a rough count 40. Our Data Base doesn't provide for M or F!) One has to start however; I have used the Apple with primary school children. My own interests are not so much in establishing ways in which it might support or enhance the more conventional aspects of the curriculum but in asking what our children should know about the micro; again I might oversimplify, but if pupils study Norman castles might they not also be pointed towards the computer?

The manner in which I have sought to explore this belief is by placing personal data about the children and their classmate in text files; addresses, ages, hobbies, heights and weights, etc. With appropriate software written by myself the children have been able to search and retrieve data thus demonstrating one important use of the computer and providing many opportunities for me to raise questions of a wider social significance in discussion; who owns this data? Is there anything here that is really private - that we really shouldn't hold on file? Whose responsibility is it to see that mistakes are corrected? Has anybody the right to see this data? My experience is that children of ten and eleven are alert to these questions and willing to discuss them. This term our work with text files is going to end in a grand crime! A child will become a criminal and we are going to use the Apple to search the data to hunt for clues and identify suspects.



Interesting though such work might be, it does take place in isolation. My own thinking over a year ago suggested that there should be a national approach to the problems raised by micros in primaries. By a series of fortunate coincidences I was able to find a like-minded number of colleagues up and down the country; the result was a national conference held during Easter this year at the University of Exeter's School of Education attended by 150 members. A national association has also been formed with I am afraid another acronym; MAPE - can you guess? Yes, the Micros And Primary Education association! Another conference is planned for April 1982 when the association which at the moment is seeking to establish a sound regional structure will be launched.

Microcomputers are in our primary schools; over the years the number is likely to grow substantially. I hope that the expertise and the experience contained in such organizations as BASUG will be able to make a contribution to the movement by demonstrating to teachers the potentials available, by advising on or even writing software, by loaning machines or by coming into the classrooms to talk to the children. Has your local primary school shown an interest? If you have a child in the school why not have a word with the classteacher or the head to see if you can bring Apple in? The teachers might be a little apprehensive; the children will love it!

### ARE YOU A PARENT, TEACHER OR LECTURER?

IF SO READ ON.....

Following the recent BASUG meeting concerning the use of Microcomputers in Education, it is hoped to form a Special Interest Group devoted entirely to the subject of Education.

Among those members regularly attending BASUG meetings all levels of education are represented, from Primary through to Higher Education. Also many members have shown an interest in writing programs to help their children's mental development, whether the child be eighteen months or eighteen years.

Is this situation reflected in the membership as a whole? We will never know unless you tell us!

Anyone interested in joining the EDUCATION GROUP please write to:-

Mrs. Norah Arnold,  
Education S.I.G.,  
B.A.S.U.G.,  
P.O.Box 174,  
Watford, WD2 6NF.

### SUNDAY SCHOOL

By Tony Williams



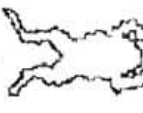

Sunday 20 September saw the national meeting of BASUG devoted to Education, held appropriately enough in the Old School House, Park St. Forty or so souls - including some six or so teachers - turned up to witness John Rodger, a lecturer at Luton Poly (and also BASUG's information officer whose bearded mug-shot adorns the cover of Hard Care No 3, for those interested) kick off with a round-up of the variety of fields in which the computer and more specifically the micro computer are coming to play a role in education. Frank Kay then took Apple Pilot through its paces, demonstrating its legendary Legends program in which text, graphics and sound are mixed to make an attractive interactive lesson package. Norah Arnold carried on with Pilot and illustrated how the character generator in its Graphics Editor package could be used to put up BIG LETTERS on the screen, of great help in remedial work. The demonstration part of the afternoon collapsed in a shambles when Tony Williams' very own Teachers Toolkit refused to work on Frank Kay's machine (an Apple). Horror, horror, a hung program in mid-demonstration! No matter that it immediately worked on Dave Bolton's machine (an Apple!), for by then the public was deeply immersed in coffee. One of our members, Paul Tilling, is a pilot, but with a remarkable lack of bias, he brought along not Pilot but Aristotle's Apple, another educational package which he uses for training purposes. Helpfully, he told us that it costs 98 Singapore dollars. Will the man please step forward who told me that it is the existence of software like this that will finally influence his decision in favour of buying an Apple... we have important matters to discuss...!

The other and equally important purpose of the session was to get the Special Interest Group, Education, off the ground. Elsewhere Graham Dane has written that Hard Core devotes too little time to education and that he would like to establish links with others involved; Brian Spielman of MUSE has lately joined BASUG. The lines are now open. Contact Norah Arnold through our Box Number.

Apologies were sent in by Dave King, Microsense's go-getting education chief, who was given all of two days' notice of our meeting and not surprisingly was committed elsewhere. Next time he promises to tell all... to the SIG. Our apologies go out to all those members who would have liked to attend but because of our various communications fiascos found out too late. We must do it again.



# Wida Software

 <p><b>apfeldeutsch</b></p>	<p>Complete Self-tuition package in German for the Apple/ITT 2020:</p> <p>9 diskettes )          6 audio cassettes ) £120.00          textbook )          workbook )</p> <p>(Classroom version also available          20% discount for schools)</p>
	<p>Stand-alone set of perpetual self-testing drills and routines in German. Article and adjectival endings, word order, time of day, money, etc. Fine for O' Level revision (BA Finals too!)</p> <p>Apple disk £15.00</p>
<p><b>Teacher's Toolkit</b></p>	<p>Define and write your own instant assessment tests - any subject - with no knowledge of computing. Write your own Introduction panels, titles, help notes. Any length. Multiple choice specimen included.</p> <p>Apple disk £20.00</p>
	<p>German:          As above but with Lower case facility and easy access to Umlaut and Eszet characters. No hardware modification necessary.</p> <p>Apple disk £30.00</p>
	<p>Coming shortly:          Russian Teacher's Toolkit (Cyrillic)          Greek Teacher's Toolkit (Greek)          French Teacher's Toolkit (French)          (Will do Hebrew upon request!)</p>
<p><b>Shape Manager</b></p>	<p>The <i>SINTA</i> Shape Manager does for shapes what a word processor does for words.</p> <p>£59-95</p> <p>a  b  c </p>

## BASIC TUTOR

College of Marketing and Design  
Dublin

Dear Sir,

I am teaching Computing in the above college. We have 16 Apples in the computer department and as we were short of Tutorial assistance I decided to ease the situation by writing a tutorial on BASIC.

While I am not entirely satisfied with the tutorial (and never will be), I believe use could be made of it, and I am now making it public domain software.

The tutorial is menu driven and user orientated. It is written in Applesoft (48k 3.2 DOS). It will only start by booting the disk. Normal RESET and CTRL-C are deactivated. The disk is also full (extra space if converted to DOS 3.3).

The disk may now be copied and distributed freely, as long as no profit is made by any distributor. As I am not in a position to distribute it widely myself, please make it available in whatever way you deem most appropriate. Please return the disk (if possible with other software on it).

Yours faithfully,  
Diarmuid McCarthy

(Ed. As you can see we have put the disk in the library, as DISK 30. We have had a brief look and it looks to be a useful introduction to using a disk system as well as to using BASIC. We have written to Mr. McCarthy offering him a choice from the library, and trust he will join as he would obviously gain as much benefit as our members will from his tutorial.

One point in reply to him openly. It does not help in the distribution of software if any form of protection is used for disks. This only hinders the distribution. If you wish to transfer the disk to 3.3 DOS you will find that the when you try to run the programs the 'FILE NOT FOUND' error will come up. This is because all the file names are saved as for example 'MENUMENU' followed by 4 CTRL-Hs. A CTRL-H is a back space so the CATALOG will only show 'MENU'. The CTRL-H moves back the cursor each time and so 4 CTRL-Hs write over the second version of the program title. To run the MENU program which is his HELLO program, when the disks boots, would mean you would have to initialise the disk with the same title which you can do as follows by typing:-

```
PRINT CHR$(4);"INIT MENUMENU";FOR N = 1
TO 4:PRINT CHR$(8);NEXT
```

or if you have MUFFINED it to run the MENU program type:-

```
PRINT CHR$(4); "RUN MENUMENU";FOR N=1 TO
4:PRINT CHR$(8);NEXT
```

Since many beginners are TAPE users then we will give priority to transcribing it to TAPE and altering it to run correctly since it is DISK menu driven.)

## Teacher writes Basic Tutor—pass it on . . .

IT is not often that Computer Weekly - even its software editor - gets diskettes of software through the post. But some unsolicited "public domain" software came our way the other day, in the shape of an Applesoft Basic Tutor from the College of Marketing and Design in Dublin.

Diarmuid McCarthy teaches in the computing department of the College, and he resorted to writing his own Basic tutor when he discovered that the manpower problem was a more difficult one to solve than machine power, when it came to teaching his students on the department's 16 Apples.

"While I am not entirely satisfied with the tutorial, and never will be, I believe use could be made of it, and I am now making it public domain software," he said.

The idea is that people should

use the Tutor freely and copy it, pass it on, and distribute it as much as possible, while making donations of their own utility and educational software if possible.

McCarthy's only restriction on the use of the disc is that no-one should make a profit from the distribution of it, but the problem is finding a suitable central distribution point for the free software, since he feels he is not in a position to distribute it himself.

The tutorial is menu-driven, and user-orientated. It is written in Applesoft and requires a 48K system running DOS 3.2. The disc is full, but extra space becomes available if it is converted to run DOS 3.3.

The Tutor is available from McCarthy at the College of Marketing and Design, Parnell Square, Dublin 1, Eire.

## ARISTOTLE'S APPLE A Review

By Norah Arnold

A computerized tutor for ANY subject, at ANY level, no programming knowledge necessary. ARISTOTLE'S APPLE sounds like the answer to any hard-pressed teacher's prayers. Indeed, for anyone who is not confident of their ability to create drill and practice programs from scratch but who is able to communicate with the computer at a fairly primitive level, Aristotle's Apple might be extremely useful.

Produced by Stoneware Microcomputer Products and distributed in the U.K. by Pete & Pam Computers, Aristotle's Apple was written by Scot Kamins, a specialist in computer aided instruction. It actually consists of three programs called EDITOR, TEST and HEMLOCK, together with a text file called DRILLS.

Three types of quiz or drills can be created by using the Editor; multiple choice, column matching and fill-in. the multiple choice drill is made up of sets of questions, each of which has five alternative answers. The user has the option of inserting 'All of the above' or 'none of the above' for the fourth and/or fifth choices, and as

these phrases may be used frequently, a very useful time-saving facility of single key entry has been provided.

In the column matching type ten items are displayed on the left side of the screen and ten items on the right. Each item on the left must be matched correctly with its 'pair' on the right. The order of the items in the right-hand column is changed each time the quiz is presented.

The third type is known as 'fill-in' because the user is prompted to type in the answer after the question is shown. Obviously vague questions will cause difficulty and this type may only be used successfully where the material is relatively specific.

Using the Editor program to create new drills is simple if you are willing to spend time previously in preparing your material for entry. The program parameters and limitations make a certain amount of preparation essential if one is to end up with a worthwhile quiz. For instance, none of the ten 'questions' or 'answers' in the column matching type may be more than fifteen characters long. This virtually restricts the kind of material one may present to single word matching in English or foreign language vocabulary or simple mathematical drills such as multiplication tables.

A multiple choice quiz may consist of anything up to fifty questions together with their appropriate answer options. The questions may be up to seventy-five characters long and the answers may contain up to forty characters. This may seem quite adequate at first glance and one can manage to keep within the limitations as long as the subject material is fairly simple. The documentation states that 'some creative abbreviations on the part of the designer should overcome this annoyance'. As a direct result of this I found myself typing in such cryptic answer options as:-

All pos combin of cap&con gds cy cn prod  
and:-

Under conds spec every econ gd has op co

in order to stay within the forty character limitation. 'A' level Economics needs a little more room to manoeuvre.

The fill-in type quiz is slightly more flexible in that it is allotted more space. The questions may have up to one hundred and fifteen characters and the answers seventy-five. The user of a fill-in quiz must type in the whole answer correctly for it to be accepted as correct.

An 'alternative' answer may be provided if desired. Experience has shown that the shorter the answer required, the more likely it is that the quiz will be successful. The chances of a pupil hitting on the exact seventy-five character responses allowed seems fairly remote.

The Editor has two further options which allow modification or deletion of drills. In addition a special purpose program called Hemlock may be used to delete several quizzes simultaneously.

There are many good features about the presentation section of Aristotle which is controlled by the program Test. The directions given on screen are clear and easy to follow. Two modes are available to the user in the Test program; 'Tutor' which provides constant feedback, reinforcement and support, and 'Exam' which tests how well the material has been learned. The thorough reinforcement of correct responses where mistakes were originally made is particularly helpful.

Some of the responses which Aristotle makes to the answers typed in by the user might need adapting in relation to particular age-groups and circumstances. Responses such as 'Guessing madly are we?' or 'Maybe you know something I don't!' might not appear very helpful to infants. Does 'Uh-uh' mean that I got it right, I wonder, - only to have my hopes shattered by a repeated request for the correct answer. Even the thrill of being told you are 'A genius, obviously a genius' wears off after a while, and who could face 'Nope' six times in a row?

The lines to modify if you wish to put in your own responses lie between 670 and 730 in the Test program. The data statements in lines 670, 680, 690, 700 and 710 contain the negative responses and those in lines 720 and 730 contain the positive.

To sum up, many extremely useful quizzes can be constructed quickly and easily with the aid of Aristotle's Apple, although it would probably be most useful to Primary and Middle school teachers and those teaching foreign languages. No doubt pupils might end up feeling rather like Skinner's pigeons if this type of material was over-used but if you are a behaviourist then a few bytes of Aristotle's Apple are just the thing for you.



## CHAIRMAN'S CORNER

By the time you get this issue of Hard Core (they come thick & fast), two important events will have occurred - and in case you haven't noticed, I am referring to the arrival of Apple in person in the U.K. (whom we welcome, with pleasure and anticipation), and the release of the Apple ///. BASUG will, I hope, be of as much interest to the owners/users of this exciting new Apple offering as to Apple II aficionados, and we will support them through an Apple /// Special Interest Group. For those of you who may be wondering whether your Apple II is now obsolete overnight, take heart! The new Apple is not a replacement (and as an Apple dealer, I expect to be selling even more Apple II's from here on), but is really for the Business/Advanced Professional user. Its facilities should also mean more Apple II software becoming available more rapidly, since the /// has advanced software development facilities and an Apple II Emulation mode. We also look forward to Apple applying the development know-how in the /// to new products for the II, although I suspect that these will be a little longer in appearing.

Now that we've got that out of the way, there is another topic which is certainly exercising minds world-wide, and I see no reason not to focus the debate within BASUG, namely the propriety of programs known collectively as bit-copiers. What's a bit-copier? Well, it allows you to copy any 'copy-protected' disk, by going back to the lowest level at which information is recorded, and reproducing this exactly. Some of these are very powerful, and have spurred the copy-protection vendors on to more esoteric and subtle ruses for preventing the bit-copier from doing its stuff, which in turn leads to cleverer bit-copiers, and the whole thing continues in (perhaps ever-decreasing) circles. It seems to me that this might be viewed as a way of keeping a lot of otherwise pleasant people intellectually occupied, so that the rest of us can lead what we consider to be normal, interesting lives, and so is a good thing. So why should we join in the debate? Well, copy-protection has been used for some time as a means of ensuring that people who have had a good idea, and have spent a lot of time and effort and (probably) money in developing it, should have the benefit of others, who recognise the good idea, etc., paying for the privilege of the use of such software. It seems very unfair that bit-copiers (most of which have been programmed not to copy the disks that they are supplied on!) should deny these software producers the rewards they had hoped for, except that Murphy's Law (para. 10097/a) says that you will have a failure in loading copy-protected software often enough to make it imperative that some form of back-up is available for immediate use. And it is the case that many suppliers of

copy-protected software only provide replacements on return of the (truly) illegible original disk. What does the inconvenienced user do in the meantime? It seems that here we have circumstances where bit-copiers can be justified, since they allow the user who has paid for the ability to use some software (be it game or professional/business) to ensure continuity of service while the original disk is replaced (often taking weeks rather than hours). The above is simply a rationalisation of the position that the I.A.C. has now adopted in the U.S.A., and which I personally think is valid. We need a BASUG consensus (although perhaps that word is becoming uncopyable, too), and a policy upon the small, but growing, band of software pirates who are flouting copyright and selling anything to anybody who will keep them in business. Even giving a pirate copy of a program to a friend devalues the program, and undermines the commitment of honest people to distribute software which is of benefit to us all - what do you think? I hope you will think about this, and write to Hard Core with your views.

Those of you who haven't been able to get to a BASUG meeting of any sort may like to know that you are not contributing to the entertainment of those who have, since a contribution of 50p. is extracted from each member at a meeting to cover refreshments, cost of premises, etc.. This is a formula which has been applied from the inception of BASUG, and it works well! Local groups are following this pattern also.

## **type-right**

Turn your Apple II or ITT 2020 into a professional Word-Processing system with full upper and lower case display and proper shift-key operation.

Plug-in fitting, with enable/disable switch, so no conflict with any existing hardware or software.

Complete with manual, fitting instructions and software on disk (including Applewriter conversion, etc).

£46.50 inclusive of VAT, Postage, etc.

### ● **MICROSOURCE**

1 Branch Rd,  
Park Street,  
St. Albans.

Tel: Park Street (0727) 72917



## LOCAL GROUPS - PROGRESS REPORT

We are pleased to say that a lot of activity now in establishing local BASUG groups, and the following information can be added to that published in the last BASUG Update:

South West London -

This group has had several meetings, and established a temporary committee to steer themselves on to bigger and better things. Some 15 members have been present, and in general meetings follow a format of one half-hour before the main core of talks and demonstrations, followed by a further half-hour of time for discussion etc. after the feature for the meeting. They have a list of some 20 topics as subjects for further meetings, with the following firm dates:

Thursday, 15th. October - 7pm to 10pm  
Languages (Pascal, Forth, Lisp, Pilot and others) - contributions and discussions from members will be welcome!

Saturday, 24th. October - 2pm to 5:30pm  
Educational Programs - There is a lot of interest amongst members nationally here.

Thursday, 12th. November - 7pm to 10pm  
Monitors/VDU's/80 Column Cards/Colour - There will be demonstrations & discussions of these, with assistance from Dealers.

Meetings are at Raynes Park Methodist Church (rear hall).

Reading/Oxford/Wallingford -

All those interested in a Local Group in this area should contact:

Chris Murphy,  
Cyderpress Ltd.,  
Church Lane,  
Wallingford,  
OXON. Tel Wallingford 37769.

St. Albans -

The Park Street meetings on Tuesday evenings are now run by a local committee, at least as successfully as those organised by the founding group in the past. Sunday meetings at Park Street are organised as "National" ones, and we hope to see many more members at these - although we have a space problem, and already need to find a suitable larger place to hold us all!

Frank Kay.



### S.B.D. SOFTWARE

15, Jocelyn Road, Richmond TW9 2TJ.  
Tel: 01-948 0461. Telex: 22861

MAGIC WINDOW - £79.95. Will instantly convert your Apple system into a word-processor, no modifications or fancy gadgets to buy. Magic Window's 4-way scrolling allows you to type up to 80 CHARACTERS per line!! This allows you to see your letter exactly the way it will be printed. Inserting, deleting, centring, and justifying can be done on the screen without the need to print a work copy.

BASIC MAILER - £49.95. A mailing list merge system designed to take MAGIC WINDOW document files and replace names, addresses or any other section of the document with individual data, creating customised letters, invoices, etc. Basic Mailer will save you so much time with form letters for mailing that it is hard to believe.

DAN PAYMAR LCA-2 - £39.95. See true upper and lower case characters on the screen. It works with the Magic Window and other programs.

MAGIC WINDOW & MAGIC MAILER - £115.00  
MAGIC WINDOW, MAGIC MAILER, & DAN PAYMAR LCA-2 - £145.00

Microsoft 16K expansion card - £95.00  
Andromeda 16K expansion card - £95.00

VISICALC DOS 3.3 - £125.00. The upgraded version of the most popular program ever.

VISIDEX - £111.00. A most useful cross reference of information. Assign multiple keywords to the index. Vary the input format as you go. Swift retrieval and printing as required.

VISIPILOT - £75.00. Plot your data onto high-res. graphs. 6 formats and 6 colours.

VISITREND/VISIPILOT - £144.00. Use sophisticated trend analysis techniques on your graphs.

VISICALC DOS 3.2. £75.00. Still available, the original Visicalc. Still as popular as ever.

BRAIN SURGEON - £30.00. Thoroughly test your Apple II. Diagnose that tricky fault.

DB MASTER (Version 3) - £130.00. Computed fields, Statistical Analysis. Faster report printing.

### UTILITIES

EXPEDITER - £75.00. At last you can compile your Applesoft program into machine code and watch it run 2-20 times faster.

CRAE - Co-Resident Applesoft Editor - £14.95. Anyone Writing software on the Apple needs this editor. Global changes and finds, renumber, append, modify. 15 commands in all.

CRAE & MCAT - Catalogue - £19.95

*We carry the full range of "Sensible Software" utilities.*

SUPER DISC COPY III - £24.95. The best most versatile copy program on the market. Initialise a disc with or without the DOS sectors. Copy files one by one or the entire disc. View the catalogue and then see a display of the diskettes free and used sectors. Copy from DOS 3.2 to 3.3 and visa-versa. Many other commands.

APPLEGUARD - £200.00. Protect your software! Protect against bit copiers.

AOPT - Applesoft Optimiser - £19.95. Optimise your Applesoft program, remove REM's, pack as many instructions as possible per line.

APLUS - Structured Basic - £19.95. Write programs in a structured manner with your new additional commands and then compile into a regular Applesoft program.

DOS PLUS - £19.95. Three new Dos Commands built-in, 5 commands are user-definable. You can now FLIP easily between Dos 3.3/3.2 from within the program. Also Dos Command Editor. Edit the names of the Dos commands and initialise disks with your own DOS.

DISC RECOVERY - "The Scanner" - £24.95. Scan your discs and mark faulty tracks so they are not used. Also able to REDO VTOC which may recover your 'messed-up' disc.

PASCAL LOWER CASE - £19.95. Utilise your Dan Paymer lower case adaptor with Pascal.

### ARCADE GAMES

Space Warrior	£12.95	Autobahn	£14.95
Alien Rain	£12.95	Star Mines	£14.95
Snoggle	£12.95	Apple Panic	£14.95
Demon Derby	£12.95	Sneakers	£16.95
Galaxy Wars	£12.95	Raster Blaster	£16.95
Gobbler	£12.95	Star Thief	£16.95
Star Cruiser	£12.95	Missile Defence	£17.95
Alien Typhoon	£13.95	Epoch	£19.95
Space Eggs	£14.95		

### ADVENTURE GAMES

Mission Asteroid	£9.95	Oldorf's Revenge	£11.95
Mystery House	£12.95	Tarturian	£14.95
The Wizard & Princess	£17.95	Creature Venture	£14.95
Cranston Manor	£17.95	Soft Porn Adventure	£17.95

### CARD GAMES & SPORTS GAMES

German Whist	£9.95	Pool 1.5	£18.95
Cribbage	£12.95	American Football	£19.95
Draw Poker	£14.95	Hi-Res Soccer	£22.95

### STRATEGY GAMES

Galactic Empire I	£12.95	Tawala's Last Redoubt	£14.95
Galactic Trader II	£12.95	Golden Mountain	£9.95
Galactic Revolution III	£12.95	Kubik	£9.95

Add 15% VAT. Postage and Packing Free  
Write or phone for full catalogue of available software.  
Dealer enquiries welcome.

## SHAPE MANAGER

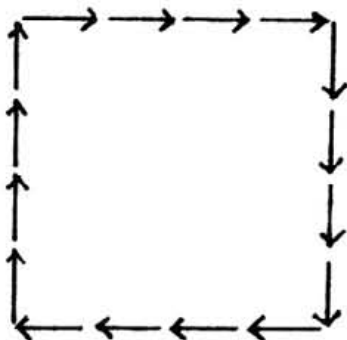
### A REVIEW OF THE NEW SINTA PACKAGE

By A.P.Mullan

Perhaps the greatest selling point at present with the Apple is the high resolution graphics potential. The use of shape tables enhances the graphics potential of the machine considerably. Shape tables are explained in the Applesoft Reference Manual but I would briefly like to outline their generation.

Shape tables are constructed by giving the computer a series of vectors to plot or not to plot. Thus if we take → as our standard vector then this shape:

Dia 1



could be coded into 4 right vectors, 4 down vectors, 4 left vectors and finally 4 up vectors, to generate the square.

Having designed the shape, in this case a very simple one the vector series has to be put into a machine readable form. The following move commands are in operation:

```
00----move up
01----move right
10----move down
11----move left
```

A prefix determines a plot with a move or a move without a plot.

```
0----don't plot
1----plot
```

Since we can code each byte we must transform our vectors into bits. However a problem exists, if the computer detects a 00 in the least significant bit then the rest of the coding is ignored so the coding must be done with care.

The coding for our shape would be as follows:

C	B	A	C	B	A
→	→		101	101	
→	→		101	101	
↓	↓		110	110	
↓	↓		110	110	
←	←		111	111	
←	←		111	111	
↑	↑		100	100	
↑	↑		100	100	
			00	000	000

We must then convert these binary numbers to hex, so our vectors become:

```
2D,2D,36,36,3F,3F,24,24,00,00,
```

which then have to be entered into the computer at the starting address of the shape table together with the index to the shapes and all the other bits and pieces needed to make the shape drawing routine work.

To hand compile anything but the simplest series of shapes is a time consuming and tedious exercise, I well remember when I first started using the Apple compiling numerals for use on the high resolution screen, and spending hours debugging.

However many utilities have now come on the market that take a lot of the tedium out of compiling shapes, and enable this most powerful aspect of the graphics to be used to a far greater extent. This is what shape manager does, allowing shapes to be defined using the computer as an aid. But it does a whole lot more as well, thus the name Shape Manager. Shapes can be duplicated and manipulated using this utility, without having to be a computer expert. The documentation describes the package as an Apple Amplifier, and it certainly is.

On initial booting (and although the disc is initialised on D.O.S. 3.2 it will boot 3.3 as well) you are greeted with the titles, then the main command menu appears:

```
L Load table
B Begin New Table
Q Quit Program
```

Since we have no intention of quitting the program at present pressing B will cause the disc to whirr again and move you into the shape construction phase. Pressing T at this point allows a look at the text page of memory where the various commands are written. Four keys control the position of the plotting spot that initially appears in the centre of the screen, up;(U), down;(J), left(←) and right(→), the plotmode being controlled by P to turn the plot both on and off. Mistakes can be rectified by using B which really does unpick the shape, as can be seen by checking on the amount of memory used.

Other utilities of this sort that I have used do prevent a mistake being plotted, but do it by leaving a hole in the finished graphics. With this package one can see the unpick follow in reverse the original plots regardless of how complicated the shape is. Whilst you are in the plotting mode the shape is said to be open, pressing C at any time will allow you to close the shape and start another one. However another nice touch in this program allows you to recall a shape already closed anywhere in the table, reopen it, modify it and then close it as a newly finished product, all the coding being reorganised to accommodate the changes in the shape.

Not only does the program allow the user to modify shapes at will but several other powerful commands are available. These all make the job of compiling shapes that much easier, in that the computer can be used to compile larger shapes from smaller bits. The first of these commands allows symmetrical shapes to be drawn easily since it will supply the other half of the shape at the touch of a key. Say for instance the user wanted a shape for use in a game, and the shape was symmetrical, then by plotting half of the shape, and ensuring the final plot was on the vertical axis of symmetry pressing H would allow the other half of the shape to be drawn and the plotting origin moved so any extra parts can be drawn, or conversely the shape can be duplicated until you run out of memory. Of course the shape does not have to be symmetrical to use this facility.

One can also make a duplicate of the shape on view, very useful if shapes are to be joined together. The duplicate can be made either forwards or backwards, that is a straight duplication can be made or the coding is reorganised so the last point to be plotted becomes the origin of this new shape. The new shape can be put into any part of the shape table and used as a duplicate or as the basis of a new shape. Shapes can also be joined to make yet more shapes. Thus a copy of the shape currently on view is joined to the copy of the shape one above it in the table, the join being made at the last plotted point on one shape being joined to the first plotted point on the second shape. This of course gives another shape in the table, which can be operated on using any of the other commands.

Many other commands are available to enable the user to have full control of the shape plotting. All the movement commands have effects on the shape being compiled and when it is closed the finished, or partially finished shape. The whole shape can thus be moved around the screen at will as can an aiming mark provided to avoid parallax problems, enabling the position of a transparent vinyl sheet that has a small drawing on that can be used as a template for shape making. With the movement of a full shape facilities are provided to enlarge and shrink and rotate the shape. There is also a facility to put an image of the current shape on the screen for comparison with an altered image or to relate one shape to another. Altogether the shape managing and manipulation commands are very powerful, more that I can do justice to in this paper.

That is not all though, for once a shape table has been constructed more facilities are available. Obviously one can save a table to disc or tape. Perhaps the tape save is slightly superfluous unless a tape man has borrowed the disc system needed to operate the shape manager

from a more fortunate friend -i.e. one with a more understanding bank manager. However upon pressing S the chance will be given to save to disc or tape. If saves to disc are called for then an ingenious system has been devised. The program will ask for the name of the shape table let us say BILL. Upon save three control "T"s are added onto the name of the table and it is saved in this form. At the same time a small binary file with the originally designated name is also saved. This binary file actually loads the shape table into memory and sets up the pointers for a Basic program to find the table. Thus if any Basic program BRUNS the shape managing file Himem is automatically set to protect the shape and all pointers are set. A useful asset this, since it saves some coding to set up these pointers. When loading in a shape table from disc that needs some extra shapes the whole disc is scanned for these control "T"s and a list of all the shape tables are displayed so selection is simple. Another asset the program has is to merge one shape table already in memory with another on disc, and then use this complete shape table either in a program or add to it. Shapes can be erased, joined, duplicated etc to your hearts content and the new table then saved.

#### Documentation

The documentation is excellent, and should be easily followed by most people. Not only does it give a full breakdown of the different commands but contains a wealth of information regarding programs and plotting. It takes a first time user gently by the hand and gives him a guided tour of the shape manager program, using all the commands and explaining what they could be used for. That is not all though - there is a section regarding loading programs above high-res and accessing them, a section on tying shape tables to programs and a very nice little section on animation. The three appendices consist of planner programs for lines, polygons and ellipses which provide plotting instructions for these shapes-and they do work, something I have come to appreciate after trying to debug some magazine programs that did not work!

The whole package comes nicely gift-wrapped in a vinyl folder inside a strong cardboard box. The contents of the package are a disc, documentation, transparent acrylic sheet and fixers, plus a pen to draw on the acrylic sheet.

All in all a very nice utility package that makes shape table manipulation much easier, in fact I would go so far as to say almost pleasant to do. The package is marketed by

Sinta Software  
3 Barnfield Gardens  
Brighton  
BN2 2HQ



**SHAKE HANDS WITH YOUR APPLE**

by Ian Pawson

**History.**

When I decided to change from my trusty UK101 to an Apple I had a lot of data to transfer so I bought the Apple High Speed Serial card to enable RS232 transfer between the two machines. As my printer (an Anadex 8000) was run from the UK101 using its serial interface, I knew that I could use the HSS card to drive it from the Apple.

**Problem.**

The problem, when I tried to use it, was that the HSS card had no Handshaking and so I had to run the printer at 300 baud to avoid its buffer overflowing. The new PROM provided with the card enables handshaking with printers capable of duplex working. It sends out a Control C to the printer and expects a Control F back. This was of no use to me as the Anadex provides RS232 type DC handshaking.

**Solution.**

A solution to the problem requiring no hardware mods to either the HSS card or the Apple, was needed. The method described here is a bit of a brute force one. It stops the 6502 processor from operating when the printer busy line goes low. This facility is provided on the Apple bus pin 21, the RDY line. Taking this line low during the 00 clock halts the processor.

Unfortunately, some electronics are needed to achieve this. The busy signal from the printer is at RS232 levels, i.e. from +12 to -12 volts. Applying this directly to TTL logic would destroy the TTL; also, it is not synchronous with the Apple clock.

Another problem was found to be that when the printer was switched off, its busy line was permanently low, thus stopping the computer. Two solutions to this problem are presented here. The first makes use of a software switch and the Reset line, while the second uses a mechanical switch to enable the busy facility. Either one consists of two TTL IC's, three diodes and a couple of resistors. They are built onto a card (either home made or the Apple or Vero prototyping cards) designed to be plugged into one of the Apple peripheral slots where they pick up the bus signals. As I don't have a language card, I use slot zero next to the HSS card in slot one. (You knew slot zero could be used for things other than memory cards didn't you?). The connexion to the printer is best via a Radio Spares PCB connector to the HSS card DB25 connector pin 5 (CTS)

**The circuit.**

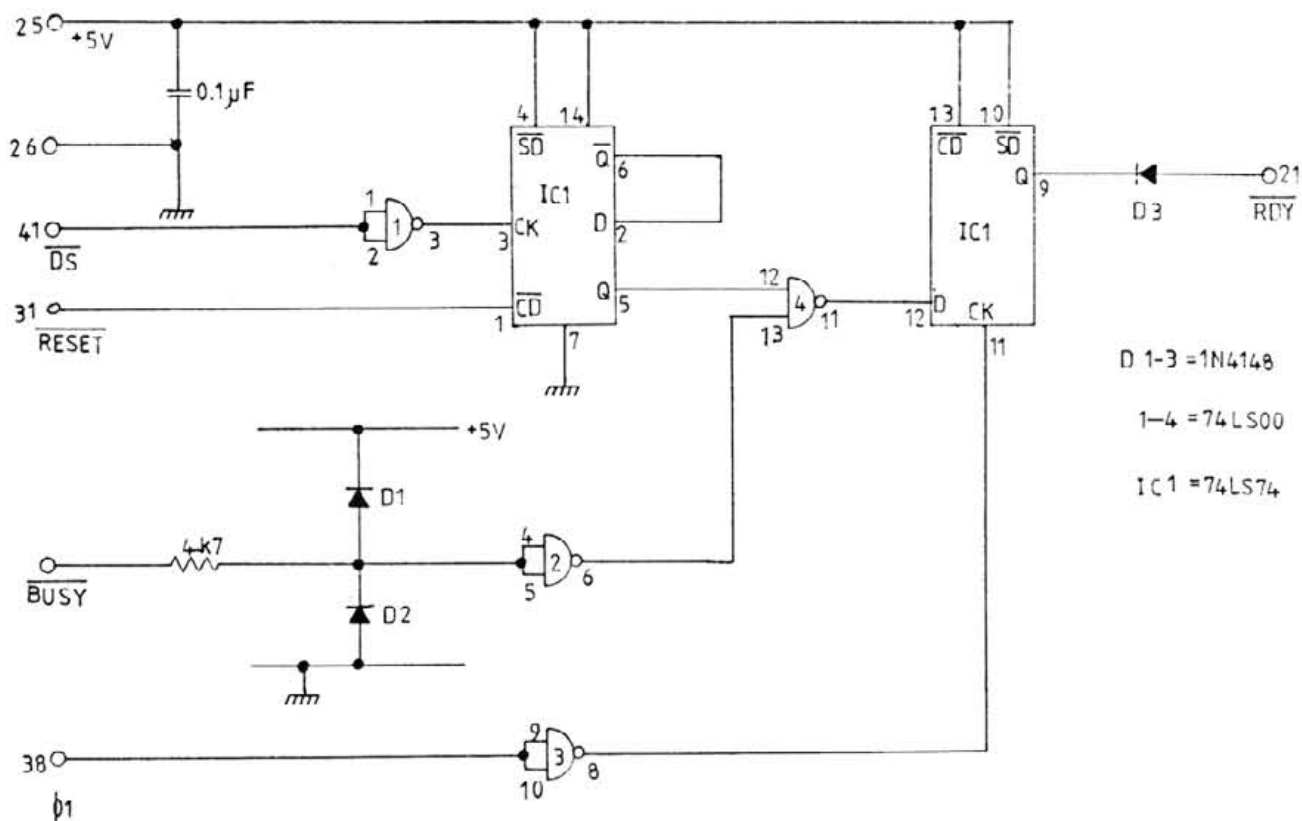
Fig.1 shows how a dual 'D' type flip-flop (74LS74) provides the sync. to the Apple clock by connecting its clock input (pin 11) to the inverted 01 clock line. (bus pin 38). This is recommended by Apple, rather than using the heavily loaded 00 clock line. The second half of the flip-flop is connected as a divide by two, and uses the device select line (bus pin 41) to gate the printer busy line. (gate 4) The reset line (bus pin 31) is used to ensure that at switch on, or by pressing the reset key, the busy facility is switched off. By addressing one of the 16 locations used by the slot that the unit is in (see Apple manual page 82) it switches the busy on. A second reference will switch it off again. If fitted in slot 0, a statement like `X=PEEK(49280)` will switch the busy on or off. (Do not use a POKE statement as this causes two references to the memory location and so will switch the unit twice and thus have no apparent effect). The diodes D1 and D2 clamp the printer busy line to levels acceptable to TTL logic and R2 limits the current. This is then inverted by gate 2 and fed to the flip-flop via the blocking gate 4.

One problem with this method, however, is that with the Autostart ROM at switch on, it searches the peripheral slots for a disk controller card. This means that, although initially reset, the busy card will be switched on. If the printer is switched off, the computer will stop (very confusing the first time that it happened). The solution to this one is to use a mechanical switch to enable the busy function. This is shown in Fig.2. The switch connects an earth (0v) to SW thus enabling gate 4 to pass on the printer busy signal to the flip-flop and thus to the computer. In this circuit only one half of the flip-flop is used and the spare inputs are connected to the +5volt line. Both circuits use all four gates in the 74LS00.

In use, the handshaking is not noticeable to the user till the end of printing when the cursor returns to the screen and the printer carries on printing out its buffer contents. The HSS card is initialised in the normal way with a PR# command. The switch can be operated at any time, even when the printer is operating.

Using this simple handshaking unit, I am now able to run my printer at 1200 baud without worrying about line length and printer buffer overflow.

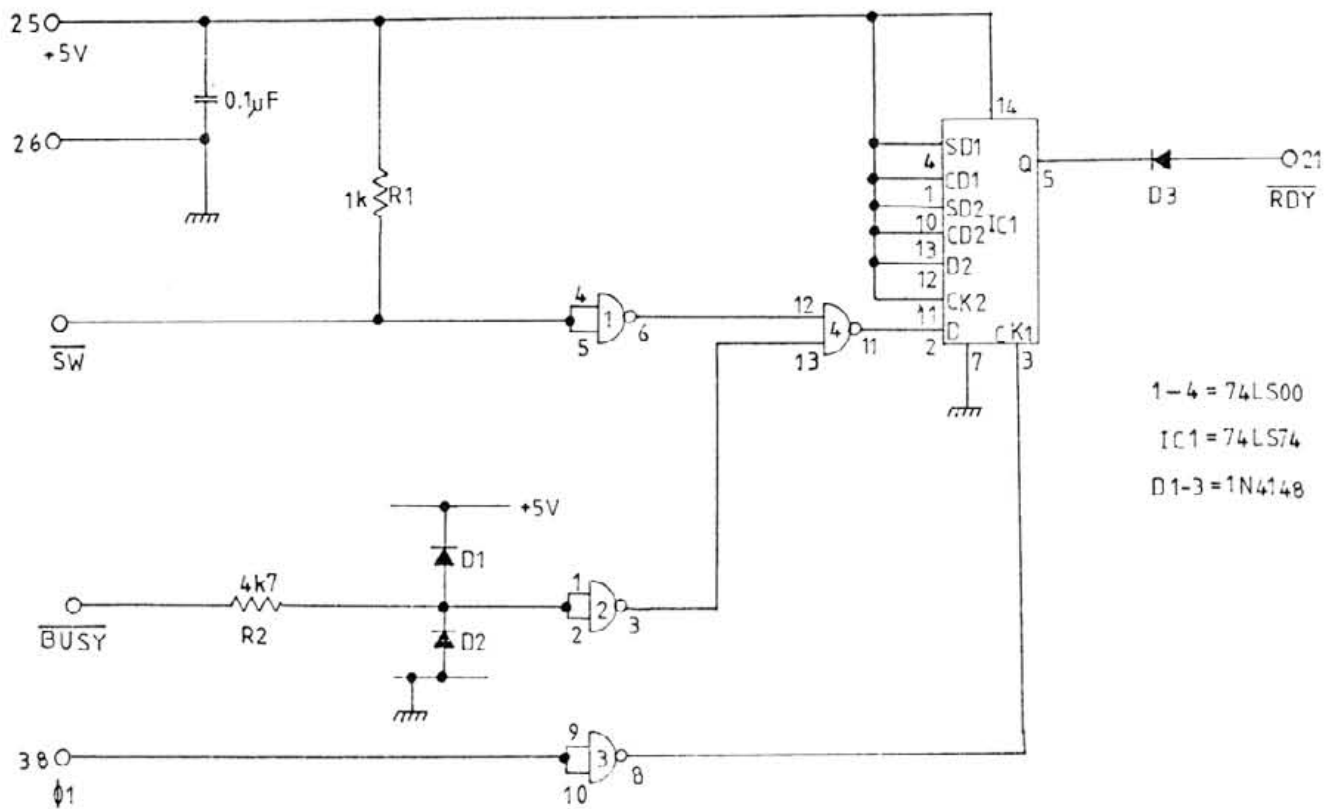




D 1-3 = 1N4148  
 1-4 = 74LS00  
 IC 1 = 74LS74

FIG 1

IP-81



1-4 = 74LS00  
 IC 1 = 74LS74  
 D1-3 = 1N4148

FIG 2

IP-81

## VISICALC

### Some Notes on the New (DOS 3.3) Version

by Ian Trackman

Here are some notes on modifications that have been incorporated into the latest version of Visicalc. Comparisons will be made with Version 1.37, which was widely distributed during 1980.

The new version operates under DOS 3.3, although it will still boot properly from a DOS 3.2 controller card.

The Tutorial Section of the Manual has been slightly revised and a completely new 50-page Reference Section has been added which deals individually with each of Visicalc's commands and features. (There are also 9 replacement pages which have to be inserted into the Manual to correct printing errors). The new Manual also contains a "Command Structure Chart", which explains the commands in diagrammatical form.

Instead of "backing out" of an unfinished command with repeated <escapes>, you can now cancel it with a single Control C.

One of the problems with the earlier version was that it was not possible to re-call a formula which ran off the edge of the screen. This has been cured by the addition of an "Edit" command which allows any entry to be re-called into the "Edit Line" at the top of the screen and then examined and modified by the insertion or deletion of characters.

The printer output command has been enhanced so that it is now possible to send escape, control and other characters to your printer from within Visicalc.

You can now produce a hard copy of all of the formulae in a work-sheet, although Visicalc presents them in a single vertical column, rather than in the work-sheet's grid format, and the formulae are listed in reverse order, that is, starting from the bottom right-hand corner of the work-sheet and moving upwards towards the top left-hand corner. (A number of separately-available Visicalc "list" utility programs enable the formulae to be printed in a better format).

The work-sheet can also be saved to disk in "Data Interchange Format", which is Personal Software's suggested industry standard for storage and retrieval of data in a two-dimensional (row / column) format. A Technical Note on the format is included with the Visicalc package.

When a work-sheet is loaded from disk, the existing screen is not cleared and the new entries simply over-write the old information. This has the advantage that one generalised "skeleton" of, say, descriptive labels and dates can be set up and different files of figures and formulae overlaid into it. Several sectional work-sheets can also now be combined into a master copy.

The new version has a specific command to exit from Visicalc and re-boot a standard disk.

Almost hidden away in the Reference Section of the Manual are six new, powerful, built-in Boolean functions - NOT, AND, OR, ISNA and ISERROR, which return TRUE and FALSE. "IF" tests a co-ordinate for TRUE or FALSE and then returns the value of one of two arguments, depending on the result of the True / False test. It is now possible to have built-in cross-casting checks of rows and columns of figures (very important in financial applications) with a formula such as :-

```
@IF(@SUM(K1...K10)=@SUM(A12...K12),@SUM(A12...K12),@ERROR)
```

These new functions, when coupled with the recalculation feature, turn Visicalc's command structure into a rudimentary programming language of its own. As an example it is now possible to build up a forecasting model under which additional machinery is purchased and extra staff are taken on as turnover reaches specified target levels. Surprisingly, the enormous capabilities that this enhancement now opens up are not explained in the Tutorial Section of the Manual.

The improvements to Visicalc regrettably take up additional memory, so that an empty work-sheet now offers only 18K of free work-space as against 25K in the old version. The Manual, quite correctly, suggests that a Ram Card would be needed for larger applications.

The price has also increased to £110

# Time Machine II™



## SPECIFICATIONS:

### REAL-TIME CLOCK AND CALENDAR

- Provides seconds, minutes, hours, day of month, month, and year in software selectable formats.
- Battery backed up.
- Leap year is automatically handled.
- Crystal controlled for accurate time  $\pm 0.01\%$ .
- 24-hour military format or 12-hour with AM/PM format.
- Six date formats and seven time formats including DD/MM/YY and MCI Apple Clock formats.
- Four software selectable interrupts (1, 1024, 1,600 and 3600 seconds).

### CONTROL ROM

- 2K driver executes in the \$C900 space and does not conflict with peripherals which occupy the \$C800 space.
- On-board firmware to:
  - read the time and date using INPUT statements.
  - permit commands to be sent to select time and date formats, enable or disable interrupts, and to enable one of two application routines.
  - permit commands to set the clock/calendar.
  - display time and date on the top line of the CRT.
  - write time and date into an Applesoft string.

### PC BOARD

- Dimensions: 2.75 x 5.50 inches.
- Solder mask on both sides.
- Gold-plated connector fingers.
- Silk screened component outlines and reference numbers.
- Complete schematic and logic diagram.

It's the close attention we've paid to small details that make **TIME MACHINE II** different. For example, we felt that you should not have to write a large program of PEEK's and POKE's to read our clock—or to set it. So, we added READ and SET commands into an on-board ROM. Then to make your programming life even easier, we added FORMAT, INTERRUPT, and APPLICATION commands. Add to this three INPUT commands and two interrupt routines and it's easy to see why we're calling **TIME MACHINE II** the first intelligent clock designed for your Apple. And now we're adding a library of applications diskettes.

Quality engineering combined with innovative software makes **TIME MACHINE II** a solid value. Tomorrow's technology at yesterday's prices...and available today.



PROTOCOL COMPUTER PRODUCTS

Apartment 3  
114 Widmore Road  
Bromley  
Kent BN1 3BE  
Telephone: 01-460 2580

# AUTOCRAT COMPUTERS

## SPECIAL OFFERS

### QUALITY 5.25" DISKS

10 for £15.49+VAT

> EPSON MX80 FT <  
>> £349 + VAT <<



Post & Packing Free

## SPECIAL ARRANGEMENTS ON APPLE HARDWARE FOR BASUG MEMBERS

AUTOCRAT COMPUTERS,  
264 Preston Road,  
Harrow, Middx., HA3 0PY.

Tel: 01 908 3636

# MICROSOURCE

**MICROSOURCE** is a new concept, specialising in utilities for the APPLE II/III 2020. New, because as users ourselves, we know what you need and so are in the unique position of being able to offer expert advice. We can thoroughly recommend the products below either we use them ourselves, or have had them specially written to fill a need. We will gladly supply more information and answer your specific enquiries.

## AN OPENING SELECTION OF OUR WARES :-

### RAMEX CARDS

£95

The only RAM Extension card for the APPLE which does not involve pulling chips out of the mother-board. In combination with your System Master Disk, you can now have the equivalent of an Integer or Applesoft ROM card, as well as being ready for PASCAL. Compatible with 250 card, Visicalc, The MILL £60P card, etc, etc.

### JAILBREAK

£33

Removes those bars from ITT 2020 graphics and allows you to run APPLE programs without the problems of incompatibility. Switchable between the APPLE and ITT Graphics.

### SPECIAL OFFER JAILBREAK & RAMEX CARD for ITT

£125

with software and details of how to use them to best advantage.

### PROGRAM LINE EDITOR

£22

Gives you the editing facilities APPLE left out. You can now insert/delete/find/display normal and control characters in lines without having to copy the whole line. No more spaces in strings or fiddling with the cursor keys. With the added bonus of defining your own commands through the ESC key. Cuts down your mistakes and saves you time.

### OMNIFONT

£27

A set of programs which allows you to use your EPSON printer to produce hard copy using ANY of the fonts in the DOS 3.3 TOOLKIT. Use the ones provided or define your own from BASIC or even from APPLEWRITER. Use all the facilities of your EPSON (including enlarge, enhance and underline) with APPLEWRITER.

### OMNIDOS

£22

Accesses both 3.1 and 3.3 disks—Automatically. Uses no extra memory. Solves all the problems of having to Muffin and DeMuffin files. Also includes utilities to personalise your disks, alter DOS commands and protect your disks from copying.

### DISPLAY-WRITER

£37

The program for the APPLE owner wishing to use the computer for advertising purposes. It allows you to set up a changing display on the screen in various character sets and colours, in upper and lower case to advertise or inform. Compatible with Versawriter.

### APPLEPLOT CONVERTER

£10

Allows you to print the graphs on an EPSON printer in all modes. (Graphics facilities must be present on the printer).

### SUPER-VADERS

£17

A totally new version of the ever popular game.

### RELOCATED INTEGER

£16

Allows you to run any Integer program without the expense of a language card, including those using Hi-res graphics. Mini-assembler included. Specify memory size and DOS 3.2/3.3.

### AUTO-INDEX

£18

Allows you to make a Master Catalog disk, with fully automatic updating, and comprehensive search and sorting routines. Requires 48k and disk. Specify DOS 3.2/3.3.

### GAMES EXPANSION PORT

£11.50

Forget having to take the lid off everytime you want to change paddles for other accessories. Run two sets of paddles.

### BENEATH APPLE DOS

£11.95

This book is a goldmine of information on how DOS works, what it does and where all the locations are for the various routines.

Send SAE for complete CATALOG.

If you don't see what you want -please ask- we can obtain most APPLE related products quickly and at competitive prices.

We also accept payment by credit card and can offer credit terms by arrangement.

We wish to expand our range. Contact us if you have any software to market.

DEALERS please ask for terms.

## MICROSOURCE

1 BRANCH ROAD, PARK STREET, ST. ALBANS, HERTS

TEL: 0727 72917 & 09273 75093

## PEDRO COMPUTER SERVICES

**THE BRAIN SURGEON:** This Diagnostic Package is designed to check every major area of your computer, detect errors and report any malfunctions. The Brain Surgeon will put your system through exhaustive, thorough procedures, testing and reporting all findings. £ 30.00

**APPLE II SERIAL I/O:** Plugs into any slot: RS232 Input and Output: On Board Switch selectable 5 to 8 data bits, 1 or 2 stop bits. Program for using an Apple II for a video or intelligent terminal. Write for full details. £ 45.00

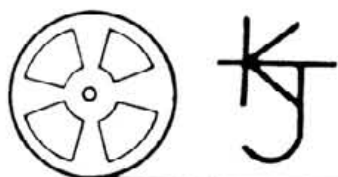
**VU 3** is a modular disk based utility program which has been designed to take data, produced by VISICALC and to reorganize and restructure so that it can be read by other VISICALC programs. This program will allow the user to enter data into VISICALC from any program merely by inserting data into an array. VU 3 will also transfer data generated from VISICALC into any of the user's programs via an array. £ 30.00

**APPLE 8-CHANNEL RELAY SWITCHING UNIT:** has 8 input channels and 8 relay output channels. Plugs into any slot. Drive software included. £ 150.00

Mail order welcome. Please send for catalogue.

65 GLEBE CRESCENT  
KENTON, MIDDX.  
HA3 9LB

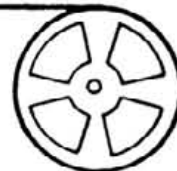
Telephone 01-204 9351



**kay-jay studios**

135 COLES GREEN ROAD LONDON NW2 7HN

TEL: 01 460 8993



You know we sell Apples, but did you know we also sell:  
 C12 cassettes £5 for 10  
 BASF 5.25" Discs £23.50 for 10  
 8" Fanfold paper, 2000 for £11  
 9.5" Fanfold paper, 2000 for £15  
 Stakpak cassette filing system, 5 drawers £2.75  
 Cassette head cleaners, 75p each  
 All prices include VAT and P&F  
 Why not ring us for a current price list  
 BARCLAYCARD & ACCESS WELCOME.



HOW APPLESOFT STORES ITS VARIABLES

by Peter Wicks

About two months ago I had need of a machine-code program to print current variables, from Applesoft and their values, so I started to re-read the Applesoft manual (yet again!), Page 137 seemed the obvious place to start but what did it mean? On deeper analysis I found that 69,6A HEX pointed to the end of the program (and hence the beginning of variables) and wrote a one-line program to test this:-

```
10 A=10:B%=10:C$="test"
```

A CALL-151 and 69,6A gave me

```
0069- 1C 08
```

81C,83F gave me

```
081C- 41 00 84 20
0820- 00 00 00 C2 80 00 0A 00
0828- 00 00 43 80 04 14 08 00
0830- 00 20 28 44 49 53 43 20
0838- 50 4F 53 54 2D 49 4E 49
```

I knew that the ASCII code for 'A' was 41 HEX and that the Apple ASCII code for 'A' was C1 HEX. The 41 HEX at 081C was obviously 'A' and the 00 HEX at 081D HEX was because I had not used a double letter variable, but where was the 10? Continuing to 0823 HEX gave C2 which was 'B' in Apple ASCII code and I now understood the (pos) and (neg) on Page 137. The 6502 stores numbers in two's complement, and thinks of the numbers from 80 HEX to ff as negative. (see "Programming the 6502"-Zaks, p. 18).

Now for the moment of truth! 082A HEX yields 43 the ASCII code for 'C' and positive! I had thought this manual was the ultimate in literature and here it was incorrect!!

The only problem now was to find the method that Applesoft uses to store values of variables.

Working backwards strings are stored from HIMEM downwards and the first character is pointed to from the variable tables, viz:-

082C gives the length of the string 082D, E gives the starting address of the position of the string in memory (sometimes pointing to its position in the program - as in this example).

Integer variables are stored in the variable table

as 16-bit two's complement. Real variables are stored in a binary standard form. (Look at the way in which a calculator displays 100,000,000,000 as 1 11 meaning 1 times 10 raised to the power of 11). I tried twenty values for A until a pattern started to emerge.

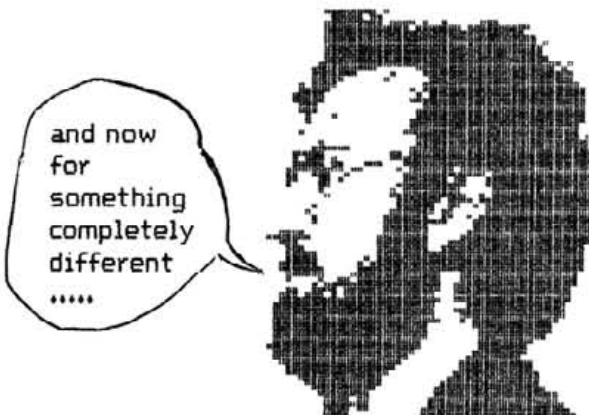
The number to be stored is firstly converted to binary, then the decimal point is moved until the number starts 0,1xxxxxxx etc. The number of times the decimal has been moved is placed in the 'exponent' byte by adding 80 HEX. The rest of the number is placed in the 'mantissa' bytes which I shall call M0,M1,M2 and M3. This is also in two's complement and the highest bit of M0 is a sign bit (i.e., '1' means -ve and '0' means + ve). The binary fraction 0,1xxxxxx etc., is now stored in successive bits of M0, M1,M2 and M3, viz:-

```
10=1010 (in binary) and 1010 = 0,1010 times 100 (in binary)
081E 84 which is 80+04 HEX,
081F,22 20 00 00 00
giving!
M0 = 00100000
M1 = 00000000
M2 = 00000000
M3 = 00000000
or + ve 0,10100000000000000000000000000000 (in binary)
NB, there is no need to store the first '1' in 0,1010 since it is always a '1'
```

Try a few simple numbers yourself but be careful since large numbers and -ve numbers are difficult to calculate.

The array variables are stored in a similar manner and the information is on Page 137 of the Applesoft Manual.

Is anyone using this program with an EPSON MX 100?



## VIDEX VIEW

By the Reverend (Squadron Leader) Cowell

As a relatively new owner of a Videx 80-Column card, I felt that perhaps my initial reactions and experiences might be of interest to others. The hardware consists of a board which is, as the makers claim, about the same size as the Language Card. The size may seem irrelevant to some, but I move my Apple around the countryside a fair bit one way or another, and the thought of a board as long, say, as the "Sup'r'terminal" wagging around and putting a fair amount of strain on the edge connectors definitely worries me. I always remove my Mountain Hardware Clockcard, for example, for the same reason (especially since the latter has a hefty battery at the end to increase the moments of inertia still further).

Also included is a connector lead which, when plugged into the board allows the connection of the usual phono-socket type video plug, and a substantial manual. Installation presents no problems, the board being compatible with any non-zero slot number: the Apple standard slot for it would be slot 3 for full compatibility with Pascal and CP/M, to name but a few. Mention is made in the manual of a device known as the 'switchplate assembly' which allows the user to switch conveniently between leads without having to unplug video connectors, but no one in this country seems to have seen one. Pity.

When the board is initialised (the simplest way being to type 'PR#3'), it takes all output to its screen, and listing a BASIC program will result in all 80 columns being displayed. Meanwhile, the standard Apple screen is frozen as far as text is concerned, although HIRES graphics will operate normally should you require them on that screen. To switch between upper and lower case you have to type Control A. This gets pretty tedious to do for single letters, but the manual describes easy ways to simulate the Apple Writer use of the Escape key, for example, to use the board from Applesoft. Meanwhile, all lower case used inside quotes both lists and prints as lower case on the Videx screen, and garbage on the 40-column screen.

That is a quick and dirty way of using the board; the extensive if in places slightly abstruse manual gives full details of how to use it otherwise, together with a commented listing of the on-board firmware. There are problems which have to be overcome with some care; consider, for instance, the following sequence of events:

PR#3 (Switch to 80-cols)  
PR#1 (Switch on Printer card)

Oops! By the PR#1 you effectively disconnect the Videx card and have to reconnect it again... I got so fed up with all this that I wrote some machine-code routines which operate through the ubiquitous ampersand vector from Applesoft, giving all the facilities available from the standard Apple monitor on the 80-column screen including switching the printer without screwing everything up. OK, a sledgehammer to crack a nut, maybe, but I can now use the card in any programs with no side-effects whatsoever. Hint for those who want to do something similar: study the listing of the machine-code printer program on Page 4-10 - the trick is to call the DOS routine at \$3EA to reconnect hooks after you loop in your own routines.

I hope that the above paragraph doesn't put anyone off. In a way it's a real compliment to Videx that they provide sufficient information to be able to customise the board exactly in the way you want it. For some people the manual may tell them more than they want to know. Fair enough, but if you really want to use the alternative character set, change the shape or blinking rate of the cursor, or whatever, somewhere in those pages you will find out how to do it. True, in parts it isn't a model of clarity; nevertheless, the important thing is that all the information is there if you want it, and for my money that's the way it should be!

One final, but important point. Both the Apple Pascal system and the Microsoft Z80 card recognise the presence of the Videx card and use it. So just by booting the relevant disc you find you have an 80-column display, with upper and lower case, straight away. Pascal 1.1 supports lower case, and I find that 80 columns for this are very helpful - maybe you folks out there with only 40 columns don't know what you're missing?



## DOUBLEDOS OR HOW TO SWITCH FROM 3.2 TO 3.3 AND BACK AGAIN.

By John Sharp

When you have DOS 3.3 and all your disks are still in 3.2, you always mean to MUFFIN them over, but if you are like me you never seem to have got around to it. The time comes when you are running a 3.3 originated program, and you decide you need a program, e.g. a binary loaded picture, which is always on one of your 3.2 disks, then comes the process of saving the program your running, or just starting again after you have run Muffin and got set up again. Would it not be better if you could just switch from one DOS to another and load and save from either disks at will. Well this is how you do it.

DOS (both 3.2 and 3.3) consist of two parts. The General procedures like CATALOG and INIT and then the RWTS (short for Read and Write a Track and Sector). The two DOS types are only different in these RWTS routines so if we have a switch to swop them then we can jump back and fore. The method for setting them up is as follows:-

- (i) Boot up your 3.3 Disk.
- (ii) type CALL -151 <return>  
This will put you into the monitor.
- (iii) Now type :-  
8D00<B700,BFFFM <RETURN>  
This will move the RWTS routines to sit below DOS.
- (iv) Save these routines on a 16 sector disk by typing:-  
BSAVE DOS3,A#8D00,L#900
- (v) Next RUN THE BOOT13 PROGRAM, to boot up a 13 sector disk. DO NOT BOOT UP THROUGH a 13 sector controller if you have one as well; I will explain why later.
- (vi) Type CALL -151 <RETURN> to go into the monitor again.
- (vii) Now move the 13 sector, DOS 3.2 RWTS routines, as follows.  
8400<B700,BFFFM <RETURN>
- (viii) These routines can now be saved on your 3.2 disk as follows:-  
BSAVE DOS2,A#8400,L#900
- (ix) Now return to 3.3 DOS by rebooting and MUFFIN the DOS2 file over onto a 16 sector disk.
- (x) Type in the attached listing, and save it as a precautionary measure.
- (xi) Now BLOAD DOS2. Then BLOAD DOS3 and save the total file:-  
BSAVE DOUBLEDOS,A#8390,L#1300

Now to run it all you need to do is BRUN DOUBLEDOS. It is ready and waiting for you to type &2 to switch to 3.2 DOS and &3 to switch to 3.3 DOS.

It takes up about 5.5K of extra memory below DOS, but normally you never use all the memory. I have not used an assembler so there are no comments, and a brief summary may therefore be in order.

The routine first says where to jump to when the & routine is called upon, and then sets HIMEM. This happens when you BRUN the program. The routine from 83A4 to where the DOS 3.2 routines sit at 8400 does the check on which DOS you have asked for and then sets up the registers for the memory move routine in the monitor at FE2C to actually do the move.

When I first set up the routine on my machine which has a two controller cards, one 3.2 and the other 3.3, I booted up on the 3.2 drive in order to copy the 3.2 RWTS routines. I then tried the whole program on another machine, which only had one controller card. When I moved the 3.2 routines and Catalogued the disk, the cursor went away and I had to press RESET to get it back. I tried again and it worked. So the APPLE knows all was not right and needed some pointers resetting. This is why I specify the BOOT13 program to get the 3.2 RWTS routines up.

One note of caution. You may need to reset HIMEM if you type FP or your program alters the HIMEM pointers. There may be trouble with INTEGER programs, although so far I have not experienced any with any program.

Do not use this DOS to initialise a disk - funny things might happen if you do.

8390-	A9 4C	LDA	#14C
8392-	A2 AD	LDX	#1AD
8394-	A0 83	LDY	#83
8396-	8D F5 03	STA	\$03F5
8399-	8E F6 03	STX	\$03F6
839C-	8C F7 03	STY	\$03F7
839F-	A2 85	LDX	#85
83A1-	A0 83	LDY	#83
83A3-	86 73	STX	\$73
83A5-	84 74	STY	\$74
83A7-	86 6F	STX	\$6F
83A9-	84 70	STY	\$70
83AA-	60	RTS	
83AC-	FA	NOP	
83AD-	AA	TAX	
83AE-	F0 4E	BEQ	#83F8
83B0-	20 E1 00	JSR	\$00E1
83B3-	E0 32	CPX	#32
83B5-	F0 06	BEQ	#83E0
83B7-	E0 33	CPX	#33
83B9-	F0 20	BEQ	#83DE
83BB-	D0 F0	BNE	#83AD
83BD-	A2 00	LDX	#00
83BF-	A0 84	LDY	#84
83C1-	86 3C	STX	\$3C
83C3-	84 3D	STY	\$3D
83C5-	A2 FF	LDX	#FF
83C7-	A0 8C	LDY	#8C
83C9-	86 3E	STX	\$3E
83CB-	84 3F	STY	\$3F
83CD-	A2 00	LDX	#00
83CF-	A0 E7	LDY	#E7
83D1-	86 42	STX	\$42
83D3-	84 43	STY	\$43
83D5-	A0 00	LDY	#00
83D7-	20 2C FE	JSR	\$FE2C
83DA-	60	RTS	
83DB-	A2 00	LDX	#00
83DD-	A0 8D	LDY	#8D
83DF-	86 3C	STX	\$3C
83E1-	84 3D	STY	\$3D
83E3-	A2 FF	LDX	#FF
83E5-	A0 95	LDY	#95
83E7-	86 3E	STX	\$3E
83E9-	84 3F	STY	\$3F
83EB-	A2 00	LDX	#00
83ED-	A0 E7	LDY	#E7
83EF-	86 42	STX	\$42
83F1-	84 43	STY	\$43
83F3-	A0 00	LDY	#00
83F5-	20 2C FE	JSR	\$FE2C
83F8-	60	RTS	
83F9-	00	BRK	



## SOFTWARE LIBRARY BULLETIN

By John Rogers

Here is a catalog of yet more software from the B.A.S.U.G software library, for your entertainment, education and frustration.

On the frustration note, it appears from some correspondence that the arrangements for people with tape are a little unclear. The first point is that a lot of the software will only work with a disk based system; this mostly applies to utility and database type programs. Secondly, that to put a whole disk of software onto tape will normally require two tapes (ie. 20 mins.). At one time we had a separate catalog of tapes. With the increasing amount of software and the relatively few members who had tape it became impractical to put all software onto tape. The system now is to ask for a tape to be copied from a particular disk, and all relevant programs will be specially transcribed for you. Where a particular disk has a number of programs which will only work with disk, we will do our best to give you other similar programs to make up the loss.

As a reminder on prices of the SDL the cost of a disk is £3 and of two tapes(all the software from a disk) is also £3.

### DISK No.30

#### BASIC TUTORIAL DISC

This disk has been donated by Diarmuid McCarthy of the College of Marketing and Design, Dublin. It is a menu-driven user-orientated Applesoft tutorial. You may have seen this mentioned in Computer Weekly and other magazines!

### DISK No.31

#### APPLE TUTOR 1

I 003 HELLO
I 004 BASIC PROGRAMMING
I 036 BASIC PROGRAMMING 1
I 047 BASIC PROGRAMMING 2
I 048 BASIC PROGRAMMING 3
I 049 BASIC PROGRAMMING 4
B 022 BASIC-INTEGERS
I 019 CONVENTIONS
I 016 CPU 6502
I 002 HELLO SAMPLE
I 030 MICRO 6502 SIMULATION
I 051 MINI ASSEMBLER TUTORIAL
A 005 NAKED CITY (C)
A 022 RANDOM DRILL TUTORIAL
I 007 SWEET 16 DISASSEMBLER
I 004 SWEET 16 SPEED ?
B 002 SWEET 16 SPEED.X
I 026 TOP DOWN PROGRAMMING

### DISK No 32

#### MATHEMATICS & STATISTICS

A 003 HELLO
A 003 ANGLE CONVERSION
A 012 ANGLE TO METRIC III
A 003 AREA OF POLYGON
A 008 BINOMIAL DISTRIBUTION
A 004 BLACKBODY
A 003 CHI SQUARE DISTRIBUTION
A 004 CHI SQUARE TEST
A 004 COORDINATE CONVERSION
A 008 COORDINATE PLOT
A 003 CURVILINEAR INTERPOLATION
A 002 DERIVATIVE
A 016 DFIT
A 011 DIFFERENTIAL EQN SOLVER DEMO
A 004 EXPONENTIAL REGRESSION
A 004 F DISTRIBUTION
A 004 GAUSSIAN QUADRATURE I
A 003 GEOMETRIC MEAN
A 004 GEOMETRIC REGRESSION
A 004 GREATEST COMMON DENOMINATOR
A 005 HISTOGRAM
A 003 LINEAR INTERPOLATION
A 007 LINEAR PROGRAMMING
A 004 LINEAR REGRESSION
A 005 MANN WHITNEY U TEST
A 022 MATH DRILL III
A 019 MATH MULTIPLY DRILL
A 019 MATH->NEEDS WORK

A 013 MATRICES
A 004 MATRIX INVERSION I
A 004 MATRIX MULTIPLICATION
A 004 MATRIX OPERATION SIMPLE
A 004 MEAN VARI STDR DEVIA I
A 007 MULTIPLE LINEAR REGRESSION
A 004 NORMAL DISTRIBUTION
A 006 NTH ORDER REGRESSION
A 004 NUMBER COMBINATIONS
A 002 PARABOLA PLOT
A 003 PERMUTATION COMBINATION I
A 007 PERMUTATION COMBINATION II
A 003 POISSON DISTRIBUTION
A 006 POLAR EQUATION PLOT
A 011 POLYFIT
A 013 POLYNOMIAL REGRESSION
A 006 POWER CURVE FIT
A 003 PRIME FACTORS I
A 006 PRIME FACTORS II
A 003 QUADRATIC FORMULA
A 006 QUADRATIC SURFACE
A 015 RIGHT TRIANGLE SOLVER
A 016 ROOT FINDER
A 005 ROOTS OF POLY HALF
A 005 ROOTS OF POLYNOMIALS
A 004 SIMPSON'S RULE
A 004 SIMULTANEOUS EQUATIONS
A 004 T DISTRIBUTION
A 005 T DISTRIBUTION TEST
A 003 TRAPEZOIDAL RULE
A 005 TRIANGLE FACTORS
A 007 TRIANGLE PARTS
A 003 TRIG POLYNOMIAL
A 004 VECTOR ANALYSIS
A 003 VECTOR OPERATIONS

**DISK No.33**  
**ASSORTED GAMES**

A 003 HELLO
A 027 BLACKJACK STRATEGY
A 012 COMBAT
A 017 CRAPS BH
A 035 CRIBBAGE I BH
A 028 FOOTBALL PREDICTIONS
A 011 FOX AND HOUNDS
A 026 FRENCH MILITARY GAME
A 020 GOLD MINE
A 022 GOLF II
A 016 HI Q
A 027 HOCKEY I
A 021 HORSE RACE III
A 031 KINGDOM
A 008 LITERATURE QUIZ
A 026 MAROONED IN SPACE
A 004 PING PONG
A 010 ROBOT BH
A 016 SURVIVE
A 018 THONKY I
A 018 WORD MAZE MAKER

**DISK No.36**  
**MUSIC AND SOUND**

A 003 AAA MUSIC & SOUND 183
T 003 AMERICA
I 012 HANON ETUDE #1 IN C
T 002 HAPPY BIRTHDAY
A 015 HARMONIC ANALYSIS
I 019 MUSIC COMPUTER
I 036 MUSIC FUNCTIONS
I 011 MUSIC GEE
I 023 MUSIC IN JB
I 014 MUSIC MAKER
I 017 MUSIC MOZART THEME
I 007 MUSIC RIGBY
I 026 MUSIC STAIRWAY TO HEAVEN
A 004 MUSIC START
I 030 MUSIC THO VOICES
I 022 MUSIC WRITER 3
I 008 MUSICAL KEYBOARD
B 002 MUSICAL KEYBOARD.X
A 017 MUSICAL MEMORY
I 010 NAME THAT TUNE
I 007 SOUND EFFECTS DEMOS
A 005 SOUND EFFECTS HOW TO
I 007 SOUND EFFECTS RANDOM
A 003 SOUND ROUTINE
I 011 SOUNDS AND MUSIC STARHARS
I 006 SHANEE RIVER
T 038 TUNES
A 002 HELLO

**DISK No.34**  
**MIXTURE**

A 003 HELLO
*T 003 AMERICA
A 022 APPLE FILE (C)
A 003 ART CIRCLE
B 006 CARD1
B 006 CARD2
B 006 CARD3
B 006 CARD4
B 006 CARD5
B 006 CARD6
B 006 CARD7
B 006 CARD8
B 006 CARD9
A 002 CURSOR DASH
I 015 DISK SPACE 13
B 002 DISK SPACE 13.X
I 017 DISK SPACE 16
B 002 DISK SPACE 16.X
A 012 FOURIER TRANSFORM HIRES
T 004 G/L DATA
A 041 GEN LEDGER
A 031 GEN LEDGER PRINTER
*T 002 HAPPY BIRTHDAY
T 004 INDEX
A 041 MATRIX MAKER
T 003 MM-TEST
I 022 MUSIC WRITER 3
A 017 MUSICAL MEMORY
A 003 RUBBER APPLE
A 020 SHAPE EDITOR
A 010 SHAPE GENERATOR
A 003 SHAPE INSTR
T 002 SHAPE TEST
A 002 TEXT FILE READER
A 013 TITLE SLIDES
A 006 VISICALC COORD FORMULAS
A 012 VISICALC D FILE PRINTER
A 014 VISICALC FORMULAS
A 006 VISICALC FORMULAS INSTR

**DISK No.35**  
**PRINTER UTILITIES**

A 003 HELLO
B 002 ASCII
B 013 ASCII PRINTER DRIVER
I 003 BAUD RATE
I 005 BAUD RATE ADJUSTMENT
B 002 BAUD RATE.X
B 002 DRIVER A768 L176
A 002 EPSON CATALOG DEL STRIKE
A 002 EPSON GREETING
A 002 EPSON LOWER CASE
A 003 EPSON MX80 CATALOG
A 005 EPSON MX80 DEMO 1
A 007 EPSON MX80 DEMO 2
A 024 EPSON MX80 LABEL MAKER
B 034 EPSON MX80 LABEL.PIC
A 003 EPSON MX80 LETTERHEAD
A 015 EPSON MX80 PROGRAM LIST
A 023 EPSON MX80 REMINDER CALENDAR
A 033 EPSON MX80 SETUP
A 005 H14 DOCUMENTATION
B 002 H14 DRIVER,\$300
B 002 H14 DRIVER,\$390
B 002 H14 DRIVER,\$800
B 002 H14 DRIVER,\$9500
B 002 H14 DRIVER,APMAIL
I 004 ID 440 SCREEN DUMP
B 003 ID 440 SCREEN DUMP.X
I 003 IDS 255 PRINTER DEMO
A 005 LABEL NUMBERING
T 002 LISTER
A 009 LISTER TRENDCOM 200
A 005 PAPER TIGER HGR DUMP
B 003 PAPER TIGER HGR DUMP.X
B 034 PAPER TIGER.PIC
A 002 PICTURE LOADER
I 009 PRINTER IDS
I 006 PRINTER IDS SETUP 1
I 010 PRINTER IDS SETUP 2
I 010 PRINTER IP 125
B 002 PRINTER PATCH
A 021 PRINTER PATCH DOC
A 011 PRINTER PR40 BANNER
A 007 PRINTER PR40 JUSTIFY
A 003 PRINTER QUICK DOC
B 002 PRINTER QUICK DRIVER
B 008 RTTY A2048 L1535
I 003 SILENTYPE BOLD PRINT
B 002 SILENTYPE BOLD.X
B 002 TTY A880 L234
B 002 TTY BAUDOT
B 013 TTY BAUDOT DRIVER
B 009 TTY BAUDOT DRIVER MICRO #14
B 005 TWO TONE ALARM

## DISK No.37

MISCELLANEOUS  
UTILITIES

A 003 AAA UNKNOWN 210
B 012 ALL PURPOSE DISASMB.(800)
B 012 ALL PURPOSE DISASMB.(8800)
B 010 ASCII ?
B 022 ASM A2048 L5119
B 004 CATALOG
B 002 CMP D1 & D2 AFTER COPY
B 005 DATA MOVER A2048 L768
B 002 DRIVER
B 002 FINDERX
B 006 HIRES ?
B 002 INT LINE X-REF(CALL2048)
B 007 LAZARUS-RECOVER A PROGRAM
B 002 LIST.CODE
B 002 LOCK
B 002 MEM.DUMP
B 071 MEMORY VERIFY
B 011 MEMSEARCH
B 071 MEMTEST A21583 L17860
B 003 MEMTEST A640 L384
B 007 ONE ON ONE A2080 L1285
B 002 PRINTER
B 003 RENUMBER MACH
A 003 ROTATING CROSS
B 003 SOFSYREF
B 012 SOFSYREF.S
I 012 SUPER HIRES GRAPHICS ?
B 006 SUPER HIRES GRAPHICS.X
B 002 TEST
B 007 TEST1
B 004 VARIABLE TABLE PRINTOUT
A 002 HELLO

## DISK No.38

MISCELLANEOUS  
UTILITIES

A 003 HELLO
A 003 ALPHABETIZE
A 003 APPEND FILE
A 002 AUTOBOOT 16
B 004 AUTOBOOT 16.X
I 043 BASIC-APPLESOFT
A 005 BSTAT HOW TO
A 002 BSTAT II
A 003 CATALOG SORTER
A 003 COMMAND GENERATOR
B 002 COMMAND GENERATOR.X
A 002 COMP-X MONITOR (C)
B 016 COMP-X MONITOR.X
A 006 DEBUGGING AID
A 004 DECIMAL POINT
A 004 DOS COMMAND FINDER
A 005 DOS COMMAND FINDER REV
A 012 DOS SHORT COMMANDS
B 002 DOS SHORT COMMANDS.X
B 002 DOS SHORT.RESTORE
A 014 DUMP TRACK AND SECTOR
T 002 EXEC FILE
A 013 EXEC FILE GENERATOR
A 005 FREE SECTORS DOS PATCH
A 002 JUMP POINTS DOS 3.2
B 002 JUMP POINTS DOS 3.2.X
T 009 LIBRARY 1
I 017 LIBRARY REVIEW INPUT
A 004 LIBRARY INPUT CORRECTOR
A 019 LIBRARY REVIEW
B 006 LIBRARY REVIEW.X
T 002 LITERAL

A 005 LITERAL INPUT
A 004 LOWEH HIGHM AB
A 008 MEMORY AVAILABLE RAM ONLY
B 004 MEMORY CHECK 6502
A 003 MEMORY CLEAR
I 008 MOTOR SPEED
A 003 MOTOR SPEED DOC
T 002 NAMES
A 007 NAMES FILE CREATE
A 005 NAMES FILE EXCH
A 007 NAMES FILE READ
A 004 PRINT USING I
A 005 PRINT USING II
A 004 RECOVER LOST FILES
A 005 REMOVE STRING SPACES
A 012 SORT NUMBERS WOODRUM
A 009 SORT QUICK ALPHA
A 006 SORT QUICK NUMBER
A 005 SORT WORDS II
A 005 STRING REVERSE FUNCTION
A 007 STRING SAVE DEMO
A 009 SUPER CATALOG BUILDER
B 002 SUPER CATALOG BUILDER.X
A 005 SYSTEM CONFIGURATION
A 004 TAPE RECALL DATA
A 004 TAPE STORE DATA
A 003 THROTTLE
A 004 TOKEN ADDRESS TABLE II
A 005 TOKEN ADDRESS TABLE III
A 006 VARIABLES DISPLAY
A 013 YES NO AND PAUSE

SECRETARIAL SECTOR

The membership having passed the 700 mark, means that the numbers of letters arriving are increasing considerably. Some days there can be over 30. These are a mixture of orders, requests, information from other sources than BASUG etc. At present these letters are processed at the weekend. If there is a slight delay, it could be that you just missed a weekend and got carried over to the next one. We are looking at ways of increasing this turn round. If you have offered to help and it has been practical to do something I will have contacted you (or tried to and had no reply when I've phoned). We are looking to have a working session one evening a week. If you can help let me know. This is particularly necessary every time the magazine goes out.

I also have a large amount of other writing to do, replying to letters etc. Many thanks to the members who are responding to the request to separate requests orders etc on different pieces of paper. Remember to put your name and address

on each one though. If you haven't received a reply it is for one of these reasons.

1. You didn't put a stamped addressed envelope in and I assumed it was for the magazine letters.
2. If you did enclose a SAE I passed it on and the person I passed it to has not come back to me or we are not sure what to say. If you feel the delay is long, try again. Don't shoot the secretary he is doing his best.
3. Your letter could have been mislaid by the Post Office or myself, and could surface in the near future.

One point I should stress is you have a local dealer. If it is a question he should be able to answer, ask him. If you do not get satisfaction, then come to BASUG. Let us know who you asked and EXACTLY what you asked. If dealers are not knowledgeable enough, we are there to help them either through BASUG or by telling MICROSENSE so that they can do something. But PLEASE do not ask for help and when I need more information not reply. I have little enough time to spend on the majority never mind on the minority who do this.



## ITT HIGH-RESOLUTION GRAPHICS INTERNALS

By Ian Trackman

The following notes are the result of my attempts to disassemble and then understand the high-resolution ("hi-res") graphics routines in the ITT (Palssoft) ROM.

Please note that all of the addresses differ from the corresponding Apple routines and so should not be used by Apple ROM owners. In fact, this provides a very easy way for a program to test whether it is running on an Apple or an ITT 2020 (see Letters - Hard Core May 1981). For example, in Basic, IF PEEK (62447) THEN .... it's an Apple, and IF NOT PEEK (62447) THEN .... it's an ITT 2020.

ITT hi-res graphics routines also use four additional locations on zero-page, \$1A - \$1D, which are not marked on the chart on page 74 of the Apple Reference Manual.

### \$F3D4

Initializes and clears hi-res page 2 to black. This is the HGR2 routine in Basic.

### \$F3DE

Initializes and clears hi-res page 1 to black. This is the HGR routine in Basic.

For the remainder of the routines, plotting, drawing, etc. takes place on the current hi-res screen unless you re-direct the "invisible" hi-res cursor to the other screen. This is done by storing the high-byte of the starting address of the desired page in location \$E6, i.e. #\$20 to plot, etc. on hi-res page 1 (\$2000 - \$3FFF) and #\$40 for hi-res page 2 (\$4000 - \$5FFF). Please refer to the Applesoft and Reference Manuals for details of the various soft switches for hi-res graphics.

The entry points for the main routines follow after Basic's syntax checks, so it's up to you to stay within the permitted ranges of plotting and drawing co-ordinates. The vertical co-ordinate must be between #\$0 and #\$BF and the horizontal co-ordinate must stay in the range #\$0 to #\$167 (split between two registers for a 16-bit number). These correspond, of course, to the decimal ranges permitted in Basic of 0 - 191 and 0 - 359.

### \$F3EE

Clears the current hi-res screen to black.

### \$F3F2

Clears the screen to the current colour. This must either be preceded by an HPLOT 0,0 (see below for the appropriate routine) or you have to (a) copy the contents of \$E4 (the colour mask) into \$1C (a temporary colour mask) and (b) enter with #\$0 in the A register.

### \$F452

Positions the hi-res cursor without plotting. Enter with the vertical co-ordinate in the A register and the horizontal co-ordinate in the X and Y registers (low / high address).

### \$F498

Calls \$F452 to position the cursor and then tries to plot a point there. I say "tries" because of the problem of plotting a non-white colour at its complementary colour's X co-ordinate. Enter with the registers set as for \$F452.

### \$F56C

Draws a line from the last point positioned, plotted or drawn to the vertical co-ordinate in the Y register and the horizontal co-ordinate in the A and X registers (low / high address). (Note the difference from the \$F498 entry format).

### \$F5F8

Locates the hi-res cursor and stores its vertical co-ordinate in location \$E2 and its horizontal co-ordinate in locations \$E0, \$E1 (low / high address).

### \$F62F

Sets up an inversion mask and then falls into \$F634 to draw, so that this is the XDRAW sub-routine.

### \$F634

Draws a shape. The shape's vectors' starting address in memory is pointed to by the contents of locations \$1A and \$1B (low / high address). Before calling here, position the hi-res cursor at the shape's required starting point on the screen by calling \$F452. Then enter here with the rotation factor in the A register. Please see below for a routine to convert the shape number to its vector's starting address.

### \$F6C9

Sets up the colour masks. Enter with the colour in the X register in the range #\$0 - #\$7. The masks themselves are #\$00, #\$AA, #\$55 and #\$FF with location \$3F set to #\$FF for colours \$0 - \$3 and set to #\$0 for colours \$4 - \$7. The mask is stored in location \$E4. Although you mustn't enter with a colour number greater than #\$7, you can get some interesting "candy-stripe" effects by calling the "screen-clear" routine at \$F3F2 with a non-mask byte (i.e. not #\$00, #\$AA, #\$55 or #\$FF) in the temporary colour mask at \$C1.

If, for some completely inexplicable reason, a Basic-only programmer has read this far, here is your reward :-

```
10 HGR2
20 HPLOT 0,0
30 FOR I = 0 TO 255
40 POKE 28, I
50 CALL 62458
60 NEXT
```

### \$E7

This location contains the scale factor for shapes, in the range #\$0 - #\$FF.

### \$E8 and \$E9

Pointer to the shape table. Used by \$F716 (see below)

### \$EA

The collision counter. With due respect to Richard Teed (Hard Core - March 1981), I have found that I can use the collision counter by testing it against a range, usually plus or minus 2 pixels over and under the number of pixels in the shape. To set the range, I run my program and keep printing out the value of location \$EA, noting its value when the shape is not in collision. Outside of that range, it has hit something.

### \$F9

This location contains the rotation factor in the range #\$0 - #\$FF.

Because both the monitor and some of the hi-res routines use locations \$3C and \$3F, it is difficult to test some of the calls in immediate mode or using step-and-trace.

For completeness, I now set out the addresses of some other routines which interpret Basic hi-res commands. You shouldn't call them from an assembly language program as you may end up in a one-way journey to Basic's error handler, but they might provide you with inspiration for your own error-trapping routines.

### \$F692 - \$F6A5

The X co-ordinate interpreter. Falls into -

### \$F6A6 - \$F6BE

The Y co-ordinate interpreter.

### \$F6C2 - \$F6C8

The colour selection interpreter. Falls into \$F6C9 (described above) to set up the colour mask.

### \$F6EA - \$F709

Plotting co-ordinate interpreter. ( #\$C1 is the Basic "TO" token).

### \$F70A - \$F70F

Rotation factor interpreter.

### \$F710 - \$F715

Scale factor interpreter.

### \$F716 - \$F73E

Checks that there are enough shapes in the shape table for the shape number specified, then calculates the shape's vectors' absolute starting address in memory. Falls into -

### \$F73F - \$F754

Sets up a drawing position after a Basic "AT" ( #\$C5).

### \$F755

Basic's entry to the DRAW routine.

### \$F75B

Basic's entry to the XDRAW routine.

### \$F761 - \$F7A7

The SHLOAD routine. Can also be called from an assembly language program except that a memory overflow will cause a jump to \$F78C and a Basic "Out of Memory" exit.

Coventry

Dear Sir,

... Some information which may be of use to ITT 2020 owners. I have recently purchased the SSI Torpedo Fire game. To my disappointment it fails to run because, I believe, it uses the Apple's HPLOT command. I have been able to find the statement in error, but I cannot list or correct it. SSI seems to have some special programming that makes this difficult or impossible. Any command except RESET just puts on the disk. This is the only program I have had real problems with and unless anybody knows a fix, ITT 2020 owners should be warned that it is a waste of money to buy SSI programs since their other games may also use the HPLOT command.

Yours sincerely,

G Lindfield

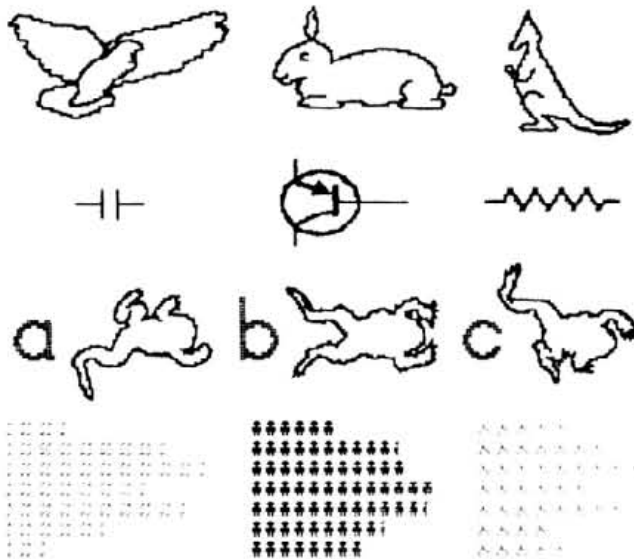
# Got an Apple? . . . or an ITT 2020? . . . then you want a

# Shape Manager

The *SINTA* Shape Manager does for shapes what a word processor does for words.

**Pictograms, special symbols, animations, a company logo, can be incorporated into the output of your own Applesoft programs as easily as plotting a single dot! They can be plotted anywhere and in any orientation.**

**Easy to use, yet one of the most powerful packages of its kind, the Shape Manager enables the swift creation of shapes and shape tables.**



Although input is through the keyboard, shapes can be traced from drawings straight onto the screen, and shape tables can be saved on disc or tape and edited or revised at any time. Loading them back for use is done by a single command.

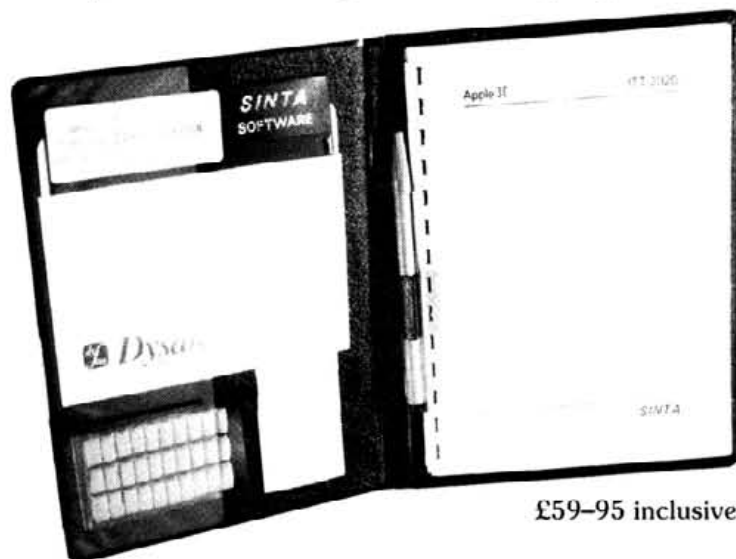
The encoding of the shapes and the tables is done in the most efficient way possible (as opposed, for instance, to mere crude scanning techniques). This not only economises on memory but also means that when a shape is summoned in a program it is fast on the draw.

Over 30 facilities include automatic generation of the rest of a shape after only part of it has been drawn; combining shapes; duplicating shapes; rearranging and mixing tables; instant recall of any shape, with single-key adjustment of position, scale and orientation; and much more.

Access to what is perhaps the most impressive of the Apple and ITT 2020 graphics capabilities is brought to the fingertips.

**The pack includes disc (with backup duplicate on the reverse), tracing materials and comprehensive manual which contains many practical tips on the use of Apple graphics.**

**System requirements: Apple II or Apple II plus or ITT 2020, discdrive, 48K RAM, Applesoft or Palssoft in ROM.**



£59-95 inclusive



THE  
COMPUTER BOOK SHOP  
RUN BY PEOPLE WHO KNOW COMPUTERS

**LION MICRO COMPUTERS**  
*NEW*  
**COMPUTER BOOK SHOP**

(ALSO STOCKING HI-FI, VIDEO AND C/B BOOKS)

**LONDON'S LARGEST  
SELECTION OF COMPUTER  
BOOKS**

PLUS ALL THE HARD TO GET COMPUTER MAGAZINES

OPEN MONDAY to SATURDAY 9-6 *PLUS*  
THURSDAY LATE CLOSING 9-7

GROUND FLOOR  
LION HOUSE  
227 TOTTENHAM COURT ROAD  
LONDON W1  
TEL: 01-580 7383

SEND SAE FOR FULL CURRENT LIST - BETTER STILL, CALL IN.

# Book Reviews

## BENEATH APPLE DOS

Reviewed by Chris Murphy

Although Apple manages to outdo most of its competitors on the quality and depth of documentation provided, one area which has not been as fully covered as some might have hoped is the disk operating system. Whilst at Microsense Computers, I once remember telexing Apple and asking if they would provide us with a listing of DOS 3.2. The answer was one of the shortest telexes I have ever received, consisting of a two letter word (at least they were polite !). For those with more than a passing interest in DOS, to find out more than is contained in the DOS manual has involved many late nights/early mornings and probably considerable frustration. Now, however, it appears that the situation has changed for the better with the publication of BENEATH APPLE DOS, by Don Worth and Pieter Lechner, published by Quality Software, California.

Printed in the same size and format as standard Apple manuals, the book is intended, as the authors state, "to serve as a companion to Apple's DOS Manual, providing additional information for the advanced programmer or the novice Apple user who wants to know more about the structure of diskettes." In order to make the whole thing hang together, there is obviously a certain amount of overlap between this book and the Apple DOS manual, but as one progresses through it, it will be appreciated that there is much in the book of value. One point worth noting before going on to consider the book in detail is that most of the discussion is based on DOS 3.3, although wherever possible an attempt is made to provide information for other releases.

After an introduction, the first topic considered is the evolution of DOS, from the rather crude DOS 3 of June 1978 upto the latest DOS 3.3 of August 1980, with a useful summary of the changes. Chapter 3 then launches into a thorough examination of the (mainly) software side of disk formatting, explaining how tracks are formatted, how data is written and read from the disk, and the difference between the organisation of sectors under DOS 3.3, Pascal and CP/M. Chapter 4 covers Diskette organisation, and provides a little more information than the standard DOS manual. Chapter 5 goes on to discuss the structure of DOS, covering DOS memory usage, the DOS vector table (that part of DOS which is not memory-size dependent, and what happens during booting.

Chapter 6 then starts us off into using DOS from assembly language (I particularly like the cartoon on page 6-1. Perhaps scope for a caption competition, or suggestions as to who one might imagine is behind the door ?), and covers the following topics :- Direct use of the disk drives (i.e. bypassing DOS altogether) ; calling the DOS RWTS routine ; the structure of the DOS Input/Output control block, with examples of the format when used by different commands ; calling the DOS File Manager (i.e. the "standard" sub-routines of DOS) and how the parameters differ for different commands ; how DOS buffers and the File Manager work areas are structured ; and finally, five useful machine code routines to locate a free DOS buffer, to find out which version of DOS is active, to test if DOS is loaded, to determine which type of Basic is currently in use, and to test if a Basic program is in execution.

Chapter 7 is entitled Customizing DOS, and gives two very useful pieces of information : firstly, how to change the boot so that instead of RUNning a program, it will BRUN or EXEC; and secondly, how to take away the pause during a long CATALOG.

To my mind, the most interesting chapter in the book is Chapter 8, which is entitled DOS program logic, and provides a detailed examination of DOS itself, including the addresses of subroutines and the variables and constants used - in fact, a 42-page D-I-Y guide to disassembling DOS.

The book then finishes with an Appendix containing 5 sample machine code programs :-

- 1) Track dump program
- 2) Disk update program (Zap)
- 3) Reformat a single track
- 4) Find track/sector lists
- 5) Binary to text file conversion

and another Appendix with some thoughts on disk protection schemes. Finally, tucked away at the back is a tear-out reference card.

If you are at all interested in finding out more about the why's and wherefore's of the Apple, then I would have no hesitation in saying that you ought to have this book in your computing library. My copy was acquired from the States, but by the time this review is published, the usual mail order outlets will no doubt have copious quantities. Time to start dropping hints about Christmas presents ?

## APPLE MACHINE LANGUAGE

Reviewed by Ian Trackman

"Learn machine language in practically no time at all!" is the claim made on the back cover of a new book entitled "APPLE MACHINE LANGUAGE" by Don and Kurt Inman (Reston Publishing Company Inc. Price £ ?). And, say the authors at the end of the eleventh of the book's twelve chapters, "Congratulations ! You are now an assembly language programmer." Can this really be true - a 295-page book which will transform you into an expert juggler of bits and bytes ? Regrettably, the answer is an emphatic "No !".

"Apple Machine Language" is a book designed for beginners and it reminded me of what used to be said of another introductory book on assembly language programming containing more than its fair share of errors - "When you start to find the mistakes, that's when you know you're beginning to understand what it's all about !". However, I don't believe that to be a very sound principle on which to write a book for readers who, by definition, are new to the subject and who are relying on their tutors to provide them with absolutely accurate information.

Don and Kurt Inman gave me the impression that they themselves were not totally at ease with 6502 machine code nor with the Apple. Throughout the book, they have used the Z80 / 8080 convention of operand notation which describes memory addresses and hexadecimal values in the form "nnnn HEX" instead of the usual 6502 form of "\$nnnn" for addresses and "&nn" for values. Not only will this confuse the reader when he/she moves on to other 6502 material, but it leads to a sudden - and unexplained - switch to standard 6502 notation when examples are given of listings produced by Apple's built-in disassembler. Furthermore, because of the careless way in which the authors fail to maintain a clear distinction between decimal and hex address and values, they fool both the reader and, on at least one occasion, themselves. The Inmans' apparent unfamiliarity with the Apple is demonstrated by two glaring mistakes and a host of inaccuracies when describing Applesoft Basic. Their "Basic Operating System", used for entering machine-code via Pokes, even contains a couple of bugs.

In my opinion, there can be no justification for setting misleading information before one's pupils, particularly when the remedy would have been a few hours' extra work on the manuscript by a well-informed 6502 programmer. The novice who also finds these mistakes in the opening chapters will already be hesitant as to how much confidence to place in the remainder of the book. Fortunately, the error rate decreases, but is by no means eradicated, as the book proceeds.

The scheme of Chapters 4 - 7 is good. Here, the reader is gradually introduced to simple ideas of assembly language through programs which use the Apple's text and graphics displays and the speaker. I liked the way in which ideas were reinforced by reminders as the book progressed and this method should have been followed by repeating comments in each new program instead of saying "same as previous program". When relative branching is used, lines and arrows pointing to the target address would make program flow easier to follow.

Chapters 8, 9 and 10 introduce two-byte mathematical operations, the flags in the status register, more "opcodes" and other features of the Apple Monitor. Interestingly, the authors don't use the Monitor's very valuable listing disassembler until Chapter 11. It is also this late in the book that they introduce the Mini-Assembler and the Step feature, neither of which are available on the Auto-start ROM. It would have been helpful if RAM version listings could have been supplied as Appendices to the book.

Most serious assembly language programming work is done using an Editor/Assembly and yet the book dismisses this whole very important topic with one passing reference. There is no explanation of labels, pseudo-opcodes and assembler directives and I didn't come across a single mention of timing.

Assembly language is potentially a highly unstructured form of programming and, if badly written, can lead to terrible debugging problems. The book hardly refers to program design and ignores questions of parameter passing and proper use of zero-page and the stack. Little attention is given to data structures.

"Apple Machine Language" falls a long way short of the claims made for by its authors. Even if you



do overcome the hurdle of the disgraceful number of errors in it, you will not be a competent machine code programmer by reading this book alone. Even the authors appear to have run out of enthusiasm for their task when they glibly admit in the final Chapter - "There are some instructions " (e.g. TYA, ORA and EOR) "that haven't been covered; but with the knowledge you now have, you will be able dig everything else out by yourself". This book succeeds in its attempt to be a simple introduction to some aspects of machine language programming on the Apple. Besides removing all of the bugs in their book, the authors should now write the rest of it.

(Editor- Ian Trackman is an experienced assembly language programmer who appears to demand high standards from this book. It would be very interesting to hear whether novice readers of the book agree with his views. Certainly the comments I hear are more favourable than this review would suggest, and we should point out that an Addenda sheet is enclosed with copies currently on sale.)

#### BOOKS IN BRIEF

Reviewed by Cliff Wooton

Title: Pascal Programming for the Apple

Author: T.G. Lewis

Publishers: Reston (Prentice Hall)

ISBN: 0-8359-5454-4

Price: Paperback £6.45

First of all let me say that I only have the briefest of experience with the Apple Pascal system. That was when it first came out and I was able to use it over the course of one weekend. I remember being a little disappointed in that there didn't seem to be any documentation comparable to, say, the Applesoft Tutorial. I still don't know much about Pascal but this is the book that I think I would have liked to have had. It is written in tutorial fashion with twenty test questions at the end of each chapter to try out the reader. There are lots of examples and although I have only skipped through it, the book looks good and I think it is well worth the £6.45. I shall certainly use it to learn from as soon as I can get a language card up and running.

Title: Programmer's Handbook

Author: Various

Publisher: Computer Station

Product No: 7702

Price: £14.95

This comes in a loose-leaf folder, three-hole punched, and is quite presentable. However, there is nothing original in the book since it contains copies of index and tabular material from all the literature supplied with Apple orientated products. There are sections covering all of the most popular products by Apple and most 'Big' software packages by independent suppliers. The Apple Monitor examples come out of the Old Red Book and, as examples go, are pretty ropey. There are also several typos which aggravate the situation. The examples suggest that you can store mini-assembler code, for instance, in the memory mapped I/O addresses. !C010! STA 23FF really is a bit NAFF. They have added a new DOS command called MAXIFILES. Several tables have been lifted from the new reference manual, but the typos in the originals have been left in, etc. etc. I am a little bit disappointed in this manual as I feel it has been slung together rather quickly to fill an immediate need. I suggest that Computer Station give it a thorough going over and a proper proof-reading before reprinting it. Then it might be worth the £14.95 price tag for which I would really have expected something rather grand with more original material.

Sorry guys, I don't like to put down anybody's efforts 'cause we are all here to enjoy ourselves, but I reckon this one rates the thumbs down at present.

P.S. Anybody want to buy a programmer's handbook?

Title: Writing Interactive Compilers and Interpreters

Author: P.J. Brown

Publisher: Wiley

ISBN: 0-471-27609-X

Price: Hardback £13.45

This is without doubt the best book I have seen for getting into the subject of compilers and

interpreters, Mr. Brown's writing style is so easy to read (not like mine) that you can literally read this book when you go to bed. It is logically laid out and deals with the various parts of the job of writing a compiler in a straightforward manner.

The book is split into eight parts as follows:

- 1) Planning
- 2) Compiler Structure
- 3) Internal Language design
- 4) The Translator
- 5) Run-Time System
- 6) Other Modules
- 7) Testing and Issuing
- 8) Some Advanced Techniques.

Each part is divided into manageable chapters. The book is not as detailed as the Gries book but Gries is more of a reference type of book anyway. There are in fact very few illustrations in the book but don't let that put you off, as the text is good enough not to need many illustrations. The references at the back of the book suggest at least another 70 items of interest in the same vein. I won't dwell on the details of the book but will say that it has given me sufficient confidence to feel that even I could write a compiler with the assistance of Mr Brown. All in all a superb addition to any Apple-ist's bookshelf.

Title: Algorithms + Data Structures = Programs

Author: Niklaus Wirth

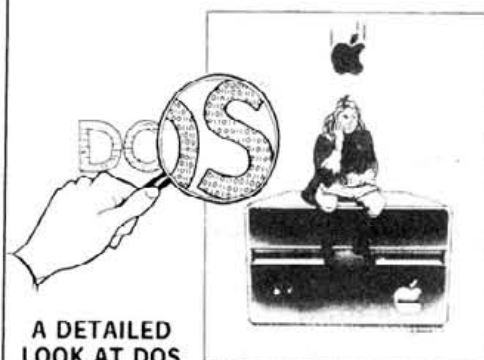
Publisher: Prentice Hall

ISBN: 0-13-022418-9

Price: Hardback £15.00 (approx)

It was once said that the definition of a Pascal book was any book written by Niklaus Wirth. I don't know whether that is true or not but this book certainly uses Pascal as a vehicle for all its examples. Nonetheless it is an extremely detailed book and when used in conjunction with P.J. Brown and D. Gries will cover all the little grey areas in between. There are five basic chapters covering Data Structure, Sorting, Recursive Algorithms, Dynamic Information Structures and Compilers. The ubiquitous Pascal syntax diagrams are also included. The text gets a little on the hairy side now and then but provided the reader can handle Pascal he will be O.K. I seem to remember Guy Kewney saying that he thought this book has perhaps the most in-depth explanation of sorting techniques he had ever seen. I tend to concur with that view as the chapter on Sort Algorithms is particularly thorough. I suggest you rush out and buy this one while you can.

## Beneath Apple DOS



A DETAILED  
LOOK AT DOS

Price £11.95  
inc p&p.

A technical manual by Don Worth and Pieter Lechner

Become an expert on the intricacies of Apple's DOS (Disk Operating System). BENEATH APPLE DOS is the perfect companion to Apple's DOS 3.3 Manual. Containing eight chapters, three appendices, a glossary, an index, and over 160 pages, this manual will serve to completely fill in the many gaps left by Apple's DOS 3.3 Manual. Written for Apple users with DOS 3.3, 3.2 or earlier versions, any Apple disk user would welcome having this carefully written manual at his fingertips.

DEALER INQUIRIES INVITED



PROTOCOL COMPUTER PRODUCTS

Apartment 3  
114 Widmore Road  
Bromley, Kent BR1 3BE

Phone: 01-460 2580

Committed to  
quality hardware  
and software.

Just Published!

## What's Where in the Apple? An Atlas to the Apple Computer

By William F. Luebbert  
Adjunct Professor of Engineering, Dartmouth College

The MOST DETAILED description to date of Apple II Firmware and Hardware.

- This Atlas and Gazetteer of PEEKs, POKEs, and CALLs lists in tabular form over 2000 memory locations.
- Information is presented numerically in the Atlas, and alphabetically in the Gazetteer.
- The names and locations of various Monitor, DOS, Integer BASIC, and Applesoft routines are listed, and information is provided on their use.
- The easy-to-use format includes:
 

The address in hexadecimal (useful for assembly programming):	.....	\$F C58
The address in signed decimal (useful for BASIC programming):	.....	(-936)
The common name of the address or routine:	.....	[HOME]
Information on the use and type of routine:	.....	\\$E\
A description of the routine:	.....	CLEAR SCROLL WINDOW TO BLANKS. SET CURSOR TO TOP LEFT CORNER
- Related register information: ..... (A- Y-REGS ALTERED)

This reference tool offers information every serious Apple user needs. BASIC and assembly language users alike will find the book helpful in understanding the Apple.

128 pages, 8½ x 11 inches, cardstock cover, Wire-O binding £9.95

**MICROSOURCE**

1 Branch Rd, Park Street, St. Albans, Herts.

Tel: Park Street (0727) 72917.

IT'S A MYSTERY

By

(HINT: Test out such Applesoft commands as RUN and LIST)

The magazines of many computer groups in the States have a column which contains one line programs or other such novelty programs. BASUG not to be outdone now begins a new column which will present program listings which do not have an obvious meaning just reading them. Do I hear you saying 'all my programs are like that'. That's probably because they are not structured. However, we don't mean that, we mean the listing doesn't give obvious clues as to what it will do. The idea is to try to work out what the program does before you type it in and run it, and not cheat. There are more mysteries. Where the future programs are coming from is one. You must have some even if you only discovered them by accident. How about a bumper lot for the Christmas issue?

The second program is by.... well that's a mystery.....

The first program is by T Tse:-

Put a little glamour into your programming and make it that bit more fun to enjoy. The following treasures were unearthed after a pleasing adventure into the uncharted lands of Applesoft Basic.

To be able to evaluate the Applesoft gems, we shall need a sample program:-

```
20 FOR I = 1 TO 1111
30 PRINT I;
40 NEXT I
```

Now type in any one of the following lines to the top of the above program:-

```
10 POKE 214,128

10 POKE 243,100
```

You are now left to yourself on an adventure of your own.

```
10 GR
20 FOR N = 0 TO 38 STEP 2
30 COLOR= 0: HLIN 0,39 AT N
40 COLOR= 10: HLIN 0,39 AT N + 1
50 NEXT N
55 X = 16
60 FOR C = 1 TO 7 STEP 2
70 FOR Y = 10 TO 30 STEP 10
80 COLOR= C
90 PLOT X,Y
100 NEXT Y
110 X = X + 3
120 NEXT C
130 COLOR= 2
140 FOR Y = 10 TO 30 STEP 10
150 PLOT 13,Y
160 NEXT Y
170 COLOR= 0:Y = 11: GOSUB 300
180 COLOR= 1: GOSUB 320
190 COLOR= 4:Y = Y + 10: GOSUB 300
200 COLOR= 5: GOSUB 320
210 COLOR= 12:Y = Y + 10: GOSUB 300
220 COLOR= 13: GOSUB 320
230 PRINT "HIT ANY KEY TO CONTINUE,"
240 GET A$
250 TEXT
260 END
300 PLOT 13,Y: PLOT 16,Y: PLOT 25,Y
310 RETURN
320 PLOT 19,Y: PLOT 22,Y
330 RETURN
```

§

**SUPPORT YOUR LOCAL ADVERTISERS !**

This journal is expensive to produce and distribute, but is provided free to members as a benefit of membership.

In order to afford to continue this policy we need advertising to defray these costs.

So DEALERS ! - please advertise with us - the rates are low and we provide a select audience of active Apple users.

And MEMBERS ! - persuade your local dealer to advertise - and whenever you buy from advertisers tell them you "saw it in Hardcore"



# Reader's letters

Horfield  
Bristol

Lots of complaints that Reader's Letters didn't appear in the last issue of Hard Core, so here's a Bumper Crop to make up for the omission.

East Ham  
London E6

Dear Sir,

Having just received the May issue of Hard Core and before I am excommunicated (or should it be exorcised) by the Exidy Sorcerers Club, may I correct the statement on Page 10 that I am a current Sorcerer owner and not a 'former' owner? The Sorcerer is still slogging away, mainly on word processing.

That slight misunderstanding apart, I would be happy to introduce any Apple owner to the East London Amateur Computer Club with a view to forming a local Apple Group within this Club.

We have one ITT owner and my own Apple II but I am sure that there are many other local Apple or ITT owners who would welcome the opportunity to meet and talk to all sorts of interesting people about computers in general as well as Apples in particular.

There is an inherent danger in a 'one machine' club which shows in some of the contents of the magazine. While the Apple has a lot going for it one should not be blind to its shortcomings and to some of the less obvious but useful features of other micros which could be made available on the Apple.

If I might use the Sorcerer as an example. This is far superior for word processing and business applications compared with the Apple 'as bought'. It has a 64 column screen, just about right for an A4 page with margins, using a standard 10 character per inch printer, and without needing a high resolution monitor. It has upper and lower case on the screen as standard. It has a shift lock button and tab key. The repeat key works without holding down the key to be repeated (i.e. one finger operation). It has a separate 16 key pad which can be user defined to perform editing and control functions (again these are single key operations). It has a Centronics interface built-in and both serial and

parallel ports ready to receive a plug (no costly cards to buy). The Exidy word processor program is in ROM at about \$120 and plugs into a slot in the side of the case - instant availability.

It also has a good set of graphic symbols, plus the facility to define your own graphics and put them onto any key (up to 128 at any time and redefinable during the program run) and a screen resolution of 512 by 240 addressable points, which is not bad.

The cassette system uses conventional cheap cassette recorders, two of them under program control if required, and operates reliably at 300 (CUTS standard) or 1200 BAUD. It does not use an 8-track cartridge as stated in the article by Graham Rubens on Page 48 as his reason for rejecting the Sorcerer. The 'cartridge' he saw contains the ROMs for the BASIC interpreter, the word processor, an Editor/Assembler in ROM or a general purpose ROM card (up to 16K) which can be programmed by the user for any specific application and can be plugged in as required (I have yet to open the case of my Sorcerer during the five years it has been in use).

If you would like to know how I came to buy an Apple if I like the Sorcerer so much, well that is a different story, but I would caution against being too insular regarding the Apple and its deficiencies. Rather we should be pressing for improvements in what is really a rather basic machine for the price, for getting rid of bugs which still exist and the rather weird hangovers from the original 16K set up as described on Page 25 of the same issue of Hard Core.

The ELACC meets on the second and fourth Tuesdays each month, at Harrow Green Library, Cathall Road, Leytonstone E11. Contact is Fred Linge on 01-554 3288 or I would be pleased to hear from anyone who would like more information on 01-472 4307 most evenings except Tuesdays.

Yours

Ted Lepley.

... I have recently been trying some machine code programming but find I do not know how to access the extra 1k of memory that ITT tag onto the HIRES graphics screen and consequently always end up with the nasty "white lines". Could you tell me how to get at this area from machine code?

I've been trying on and off for some months to get Palssoft in ROM whilst keeping Integer available (as most of my programs are written in Integer). As there is no equivalent Palssoft version of the Applesoft Card I approached several dealers for a solution but none could come up with anything viable. I was also shocked to have quoted at me a price of \$160-180 for the Palssoft chips alone (I found this incredible as the Applesoft card only costs about \$100. Can you suggest anything? Will the Z80 Softcard work on the ITT and if not are ITT intending to produce their own version?

T. Goodrick.

(Ed. There are problems in using the Z80 card on the ITT. Ken Mace at ITT did promise us some help but we have not heard any more yet. On the graphics problem see Ian Trackman's article. If you had bought an ITT with Palssoft then you would have received a program to save ITT pictures on disk which shows how to separate the two parts of a HI-RES picture, which ought to give you more clues. This routine has been put in the software library by Ray Harris and he used it to print ITT graphics on the EPSON. A copy of the routines are in the library.

We have had a number of enquiries about updating from Integer machines (it is surprising how many are about) so it would be useful to add a few words to others as well. If you get a RAM card (i.e. the plug-in card normally supplied with PASCAL - now available separately from your Apple Dealer - or the Andromeda, Microsoft or other versions, you can load the language you haven't got on the motherboard (i.e. Applesoft if you have an Integer machine, or Integer if you have an Applesoft machine). This is done automatically if you boot up on a 3.3 master or use the Basics disk. A RAM card is cheaper than the corresponding Applesoft or Integer ROM board and less versatile. It should be possible to do the same for Palssoft. Anyone any ideas? We will pass them on.)

Vange  
Essex

I left ITT just before the 2020 project was drastically chopped, having spent two years writing documentation that was never published. Since then I have collected together several sources of information to derive a nearly complete DOS Disassembly which I am currently hacking into book form for a possible publication later this year, I would be interested in helping on the journal if there are any openings and shortly will be moving down to Sussex and could perhaps help out with a regional organization. By the way, is there anything in the pipeline regarding an ABBS in the near future?  
Regards,

Cliff Hooton

(Ed. Openings? You write, we have openings! Note: ABBS= Apple Bulletin Board System. Readers: any ideas on this?)

Carrigrohane,  
Co. Cork  
Eire,

We in CESI (Ed. Computer Education Society of Ireland) are attempting to set up a program library for our members and as APPLE is the majority micro in Irish schools we may in the future be able to exchange software with you... I would be interested in finding out more about the library you offer, source of articles, etc.

Yours sincerely,

Michael Moynihan  
Chairman, CESI

Roman Wood

I have read about BASUG in the Liverpool Software Gazette with great interest and I enclose \$10 membership fee... could you please let me know whether it is possible to obtain an Apple printed circuit board - it would then be possible to use most of the I.C's in the ITT to produce a machine to the full Apple spec.  
Yours faithfully,

G.H.Poulton

(Ed. Welcome a-board, Mr Poulton (please excuse the pun). Yes, you can get a board if you have the necessary \$405! But see about the conversion to graphics mentioned elsewhere in the magazine; by using this you will be able to run ITT or Apple Graphics and so have a more versatile machine.

Bromley  
Kent

My particular interests are learning all about the Apple/ITT, particularly all the monitor subroutines which are not included in the manuals I have.

I believe there is a book called "The Apple II Monitor Peeled" by William E Dougherty, which might be of great value to me. I would appreciate it if you could let me know where I could obtain this book or any other with similar information.

Yours faithfully,

S Novotny

High Wycombe

...I have been asked to do 3D graphics for crystallography!

Henry D Talbot

(Ed. There is room for a Graphics SIG! Any volunteers to coordinate one?)

Dundee

I am interested in being the focal point for a regional group in East Scotland - I have already exhorted fellow users in Dundee to join nationally.

Tony van der Kuyt

(Ed. Go to it, Tony! Anything we can do?)

East Goscote  
Leicestershire

I would like contact with people who can explain their experiences changing from a mainframe concept to Apple system, and eventually would like to get involved in capabilities of speech control and driving electronic equipment for domestic purposes i.e. home automation.

D.M Palmer

The Robert Pattinson School  
Lincoln

Thank you for the newsletters and disk. They have aroused great interest in the department. As a school we would be interested in reports and articles from users of CAL packages giving evaluations based on their experience of using them and details of where the more useful ones may be obtained.

Yours sincerely

Peter Lancashire

Twickenham

Delighted to receive your letter... No response from UKUG despite many enquiries, I was beginning to believe there was no one out there.  
Don Bothwell

Bristol

I would be pleased to receive ... information on any branches in Bristol where I can meet other users and enthusiasts.

M J Tait

(Ed. No branches of BASUG in Bristol area yet but there is another group - BAUD - see the Local Groups page.)

Stockton on Tees

I have joined the club to get some ideas on interfacing (A/D, D/A, stepper motors, non-Apple equipment etc.). Keeping up to date with Apple developments and programming tricks, guidance on software and hardware selection.

Don Riley

Repton School

Derby

I would like to know about the "Light Pen" - I have been looking around for details but couldn't gain much. I live in Hong Kong. Is your other member in Hong Kong called Woolf by any chance?

Dieter Yih

(Ed. No Mr Yih, his name is Sammy Chan. Do you know a Mr Woolf? If so get him to join. There are a number of light pens available, maybe someone would like to do a survey? Look in the ads in American magazines or try the Sales and Wants column in HARDCORE.)

North Harrow

I have an ITT but I am soon to switch to Apple because of the incompatibility problems I am experiencing and the lack of positive help from ITT to overcome them... One major project of mine is design of a system to analyse and store horse racing results, etc. Most of my work is concentrated on developing this system.

R.L. Heron

Up Nately Rectory  
Basingstoke

Livingston  
West Lothian

Renal Haemodialysis Dept  
The Royal Sussex county Hospital

Dear BASUG,

...I and no doubt many thousands of others (sic) are having a slight problems with the micro-brains at Micro(non)sense who are still refusing to release Ian Trackman's magic 'go-between'. This would be infuriating enough if there were some genuine reason (like, for instance, it didn't work) but consider the variety of reasons offered: taken in order of appearance, we have:-

Excuse 1: It isn't available yet.

Excuse 2: The software is available but we're waiting for the manual to be printed.

Excuse 3: The software and manual are available, but we won't release it because the little labels to stick on the disk are still being printed.

Excuse 4: The software and manual are still available, and the little labels to stick on the disks have come through, but we still won't release it because we didn't like the little labels and we're having them reprinted....

The mind boggles what Excuse 5 may be! meanwhile my sympathies lie with Ian who must be finding this even more infuriating than anyone else! One wonders also what Microsense's American masters would make of this sort of inefficiency.

... You asked in a recent issue for comments on the title 'Hard Core'. Imagine, if you can, a fairly crowded railway carriage out of London into which climbs a respectable-looking clergyman complete with clerical collar. He opens his briefcase and begins to read a magazine prominently displaying the word 'Hard Core' on the front cover. Now imagine the reaction of the other people in the carriage....

With all best wishes

Brian Cowell.

(Ed. Hm... about that title... must have a think. Would a plain brown wrapper solve the problem? As for the Go-Between - it's here, I bought one. If your dealer hasn't got one, he should.)

Carlisle

I believe BASUG should exert pressure on Apple to acknowledge that the north of England exists....

N.L. Sidaway

Thank you very much for Hard Cores Nos 1 and 2, very useful and enjoyable reading... For me though the several pages on games (especially in No 2) were wasted space. I didn't really understand the "Applesoft Input Anything" routine article in No 1. I just use:

```
50 A$=""
```

```
60 GET X$
```

```
70 IF X$ <> CHR$(13) THEN A$=A$+X$:GOTO 60
```

Instead of 50 INPUT A\$ when I want to input characters such as commas and colons. It also works from disk text files. Isn't this simpler than the version you printed? If you want to allow the back space arrow to function as 'input' just insert

```
65 IF X$ = CHR$(8) THEN A$ =  
LEFT$(A$,LEN(A$)-1)
```

Of course it is also more convenient to have the input appear on the string if input is from the keyboard:

```
63 Print X$;
```

accomplishes this quite easily, including back spacing.

The programs BASUG sent me to convert PET programs to Applesoft directly from a PET Tape are wonderful. It alone is worth a year's membership. There have been a couple of failures, but I strongly suspect that misalignment of the recorder head of the PET cassette unit is to blame.

Finally, there is nothing to do with educational use in HARD CORE. If anyone would like to share an interest in this, including software exchange - they are welcome to contact me at Dean's Community High School, Livingston EH54 8PS. About 30 secondary schools in the area have Apples.

Best wishes,

Graham Dane.

Streatham Hill  
London

Why doesn't BASUG organize simple electronics courses, for interfacing, etc.

Andy Dawson

...So far we have only purchased the CCA Data Management System software pack and the Microbase Mailing List...

We have two main applications for specialized software: 1) a selector/enquiry program to process information on about 3000 local patients being surveyed for raised blood pressure (hypertension), about 30 fields of data of 30 characters per patient to be sorted for about 6 different fields simultaneously;

2) a Stock/Budget control for dialysis machinery in the hospital and for patients who have their own equipment at home - probably will be able to get this out of the CCA DMS facility program.

3) in the long term we would like to be able to control the dialysis machines directly by micros using trend analysis of past and present treatment in progress.

So we would like to be put in touch with anyone on your lists involved in medical data processing....

And of course we are interested in developing games programs, teaching programs and the like to further our knowledge and interest in programming in Microsoft/Applesoft BASIC.

Yours sincerely,

Derek Fairbank

Senior Medical Physics Technician

(Ed. One way for a medical group to start would be for all those interested to write to Derek Fairbank and then decide amongst yourselves who is going to run the Medical SIG. The committee will help as far as they can but such SIGs need the initiative of members to both get going and look after themselves.)

South Mimms

I am trying to make contact with people who have modems - although I don't have one yet. I also run amateur radio (G3100) which is not yet interfaced.

Chris Evans



# Mail Order Software

Alien Rain (Galaxian) - Colour + hi-res version of pub game.	£13.95	M48K
Alien Typhoon - A much more difficult version of Alien Rain.	£13.95	M48K
Apple Panic - Chase little apples up and down ladders.	£16.95	M48K
Autobahn - Road race game with sound and hi-res graphics.	£14.95	M48K
Both Barrels - 2 hi-res action games on one disk.	£12.95	A48K
Cranston Manor - A new hi-res adventure like Wizard & the Princess.	£19.95	M48K
Cyber Strike - 3-D hi-res action adventure in space.	£19.95	M48K
Demon Derby (Hyper Head-On) - 4 skill level hi-res car race game.	£13.95	M32K
EXPEDITER II- THE APPLESOFT COMPILER AT A LOW,LOW PRICE.	£56.95	A48K
Galaxy Wars - Colour graphics + Sound effects + Hi-Res.	£13.95	M32K
Gamma Goblins - Yet another superb hi-res action game.	£16.95	M48K
Gobbler - Eat up the dots but watch out for the Gobblers.	£13.95	M48K
Gorgon - Another superb pub game for your Apple.	£21.95	M48K
Hi-Res Cribbage - The title describes it. Even hear the pegs move.	£14.95	M48K
Hi-Res Soccer - English football for 1 or 2 players in hi-res.	£16.95	M48K
Missile Defense - Hi-res animation and sound arcade game.	£16.95	M48K
Mission:Asteroid - Hi-res adventure in 21 colours.Save the World!!	£10.95	M32K
Mystery House - Hi-res adventure using over 100 pictures.	£12.95	M48K
Oldorf's Revenge - Hi-res adventure with 100 rooms.	£11.95	A48K
Operation Apocalypse - Four computer simulations of World War II.	£35.95	A48K
Orbitron - Fight off enemy forces and avoid meteor showers.	£14.95	M48K
Paddle Graphics - Draw your own pictures in 21 hi-res colours.	£21.95	A48K
Pool 1.5 - Hi-res colour graphics pool table simulation. 4 games.	£19.95	M48K
Pulsar II - 2 superb hi-res games on one disk.	£14.95	M48K
Sneakers - Waves of little creatures attack you in hi-res.	£16.95	M48K
Snoggle (Puckman) - Hi-res maze of ghosts. Great fun.	£13.95	M48K
Space Eggs - Hi-res super-fast arcade style game.	£14.95	M48K
Space Warrior - Hi-res pub game.	£13.95	M48K
SuperScribe - Word Processor. True upper/lower case on screen.	£48.95	M48K
Tarturian (Wizard) - Another hi-res adventure with 160 rooms.	£14.95	A48K
The Shattered Alliance - Swords & Sorcery on a far-flung world.	£35.95	A48K
The Warp Factor - 1 or 2 player hi-res Star Trek/Space War game.	£23.95	A48K
The Wizard and the Princess - Hi-res adventure in 21 colours.	£18.95	M48K

A: Requires Applesoft in ROM.

M: Will run on any Apple.

All software is available on disk only and boots under DOS 3.2 (Most software will also boot under DOS 3.3). Many of the games require paddles. PRICES INCLUDE VAT AT 15%. Add 50p P+P for orders under £30 totally. Please write for your free copy of our software list.

DEALER INQUIRIES INVITED.

MAIL ORDER ONLY.

SPIDER SOFTWARE

98, AVONDALE ROAD, SOUTH CROYDON, SURREY.

Tel: 01-641 1829



# Accutrack Disks . . .

Because data reliability  
is the important difference  
in disk construction.



## Anatomy of a disk.

Flexible disks are simple information storage devices consisting of a magnetic disk enclosed in a semi-stiff protective jacket. The disk rotates within the jacket while magnetic recording heads on your data or word processing systems "read" or "write" information on the disk's magnetic surface. Since disk operation is simple, it's relatively easy to make one that works. But building in reliability is something else again. It takes specialized technology to build disks that operate flawlessly over an extended period of time.

## What counts in disk construction.

Key design objectives for a disk are listed below. How well a disk measures up to these objectives relates directly to the throughput, accuracy and overall costs for your data or word processing system. No disk measures up better than Accutrack.

- The magnetic coating must be precisely formulated and uniformly applied. Imperfections as small as five millionths of an inch cause signal dropouts, data checks and wasted processing time as well as errors.
- The disk surface must be absolutely clean, totally flat and permanently lubricated to prevent excessive head wear with subsequent signal degradation and eventual loss of information. (This is the most critical objective and the one that's most often compromised

since poor operating results take a while to show up. It's also the area that most effects the long term reliability of your data.)

- The disk must be free to rotate within its jacket without internal drag to avoid further data checks, excessive processing times and errors.
- The jacket must protect the disk from external contamination and damage. It should also remove microscopic particles of debris from the disk surface before they can damage the disk.

## Why you'll never find the best disk bargain in the bargain basement.

While there's little apparent difference between other disks and Accutrack, the performance differences can be substantial. Simply stated, an Accutrack disk is premium priced. But the protection it gives your information, the reliability it provides to your operations, and its substantially longer life make it the best disk buy. After all, the real cost of your operations is constructing and processing the data stored on the disk - not the disk itself. It doesn't make sense to trust that data to anything but the best disk. Accutrack.

**B A SUG BULK BUYING - 500<sup>off</sup> PRICE FOR 10<sup>s</sup>**

**£17:50 incl FREE LIBRARY CASE !**

(ADD 50p POSTAGE / PACKING)

**BRITISH APPLE SYSTEMS USER GROUP**

P.O. Box 174, Watford WD2 6NF.

**MORE ON PROGRAM POINTERS**

By J. Rodger

- 1). THE SELF RELOCATING PROGRAM
- 2). HOW TO DELETE LINES AND KEEP RUNNING
- 3). HOW TO PREVENT UNNECESSARY BLOAD'S
- 4). HOW TO SPEED UP A PROGRAM

Firstly, why should we wish to relocate an Applesoft program? One reason might be that we wish to use graphics pages HGR or HGR2 and our programs or variables are running into the same space. As soon as we enter graphics mode this code is overwritten. We therefore place our program above HGR (decimal locn 16384) or above HGR2 (decimal locn 24576). Some users never use the graphics capabilities, however relocation may still be necessary in the following circumstances. Many useful utilities are written in machine code starting at around location 2048 (HEX \$800), for example FID, MUFFIN, UPDATE, HIGHER TEXT etc. These utilities would be compatible with Integer Basic programs but not with Applesoft and we need to relocate our program to sit above these routines.

In order to set up your program "almost" wherever you choose you could enter the monitor and change addresses \$67 and \$68 (see Hardcore No. 1), then return to Basic and LOAD/RUN your Basic program. Alternatively you could keep on your disk the following short program to do it for you.

```
100 REM PROGRAM RELOCATOR
110 HOME : VTAB 10 : HTAB 5
120 INPUT "START LOCATION DEC ";A
130 INPUT "NAME OF PROGRAM TO RUN ";N$
140 HZ = A/256:LZ = A - HZ * 256
150 POKE 103,LZ:POKE 104,HZ: POKE A - 1,0
160 PRINT CHR$(4);"RUN";N$
```

It is good practice to make your relocated Basic program exit via the following completion routine.

```
63999 PRINT CHR$(4);"FP"
```

This resets program pointers to normal.

## 1). THE SELF RELOCATING PROGRAM

Now for a neat little dodge to get the program to relocate itself. One way that is often used to do this is with the &LOMEM utility, as published in MICRO and used by several commercial programs, e.g. HIGHER TEXT. This requires you to have a 160 byte machine code routine sitting on the same disc which must be loaded and executed at the beginning of your program. Apart from using up two sectors you have to remember to include it on every disc with your program.

The same thing can be easily accomplished without any extra programs by including the following few lines at the beginning of your program.

```
10 A = 16384:N$ = "PROGRAM NAME"
20 HZ = A / 256:LZ = A - HZ * 256
30 IF PEEK (103) = LZ AND PEEK (104) = HZ THEN 100
40 POKE 103,LZ: POKE 104,HZ: POKE A - 1,0
50 PRINT CHR$(4);"RUN";N$
```

```
100 REM REST OF YOUR PROGRAM
```

Note that A will be your required start location and "PROGRAM NAME" will be the name to save the whole program under.

## 2) HOW TO DELETE LINES AND KEEP RUNNING

The Basic statement

```
300 DEL 100,200
```

can be included in your program but the program will terminate execution immediately after deleting these lines. In order to prevent this we can use our old friends the program pointers again together with two new pointers, the CURRENT LINE POINTERS. Insert the following line anywhere in your program and all line numbers up to this point will be effectively deleted, the program will continue to run.

```
1600 POKE 103, PEEK (121) + 1: POKE 104, PEEK (122)
```

If you halt your program and list it the first line will now be 1600 (if you inserted above line at 1600)

## 3). PREVENT UNNECESSARY BLOAD'S

Using the above technique we can prevent binary files which have already been loaded from being reloaded when we rerun the program in memory.

```
100 PRINT CHR$(4);"BLOAD BINARY FILE1"
110 POKE 103, PEEK (121) + 1: POKE 104, PEEK (122)
120 REM REST OF PROGRAM
```

Similarly, if you wish to display instructions or copyright notices when a program is first run but not when rerun in memory then use the same technique

Example

```
100 HOME:VTAB12:PRINT"BASUG IS A WORTHWHILE ORGANISATION"
110 FORS=1 TO 2000:NEXT S
120 POKE 103, PEEK(121)+1: POKE104, PEEK(122)
130 HOME:VTAB12:PRINT"HARDCORE IS AN INTERESTING MAGAZINE"
Run the above program twice
```

## 4) HOW TO SPEED UP A PROGRAM

We can use this idea still further to make frequently executed routines work at maximum speed. Because of the way Applesoft uses linked lists a GOTO 20000 in a program with 20000 line numbers will take considerably longer than a GOTO 2. For this reason many programmers put frequently used routines at the beginning of their programs. This unfortunately makes it difficult for others to read. (That's the idea I hear you say but wait till you try to modify your own programs months later!!).

All you need to do is insert the routine as in line 1600 above at the beginning of your routine wherever it is and any GOTO'S within this routine will be executed at maximum speed. When the routine is complete we must set locations 103 and 104 back to normal before continuing.



## ENHANCED MODE WITH THE MX-80

By John Sharp

If you have an EPSON MX-80 printer, you may or may not have felt too happy about the manuals. You may have seen that there is an enhanced mode, which gives a much darker print than the normal text. The means of getting this are not clear from the manuals. In fact most people only discover it by accident, if then. It is fairly easy to use from BASIC, but like all the other facilities, not accessible from APPLEWRITER or other word processors. Initially I will explain how to get at it from BASIC and then describe a routine to use it with APPLEWRITER.

If you send a CTRL-O to the printer it will cause the printer to output in CONDENSED mode, and

the output will now look like this.

This will continue until you send a CTRL-R, to return you to normal. If from normal mode you send a CTRL-N you will get ENLARGED mode and

the output will  
now look like  
this.

However, unlike condensed mode, whenever you send a line feed (i.e. usually a carriage return) the enlarged mode is switched off. Now what happens when you try the two together. (An unusual request, which is why I said the following is often found by accident).

condensed + enlarged =  
enhanced

The following short BASIC program will illustrate this.

```
10 PR#1:PRINT CHR$(0): REM
SWITCHES ON PRINTER AND CLEARS
BUFFER
20 A$="THE QUICK BROWN FOX
JUMPS OVER THE LAZY DOG "
30 PRINT
40 PRINT "NORMAL MODE": PRINT
A$
50 PRINT "ENLARGED MODE":PRINT
CHR$(14);A$
60 PRINT "CONDENSED MODE":
PRINT CHR$(15);A$
70 PRINT "ENHANCED MODE" :
PRINT CHR$(14);A$
80 PRINT CHR$(18):REM SET BACK
TO NORMAL
90 PR#0
```

This is all very well, but what if you want to use APPLEWRITER. If all the file is required in condensed mode, you just type PR#1, followed by RETURN and CTRL-O and press return, wait for the buzzer to sound because it is not a correct command and gives you a syntax error, then PR#0 and go back into the PRINTER routine. However, if you wish to use ENLARGED, or ENHANCED you are stuck because even if you could signal a CTRL-N, a line feed would cancel it. The only way to use APPLEWRITER to its full extent is to intercept the output before it gets to the printer and insert the CTRL characters in then. This is the principle of Ian Trackman's 'THE GO-BETWEEN' for the CENTRONICS 737 and Michael Mathison's OMNIFONT for the EPSON. The added bonus for Hi-Res versions of the EPSON, with OMNIFONT is the ability to print in any character-font you can define using the DOS 3.3 Toolkit. To use ENHANCE the following routine is possible. All we

need do is set the condensed mode, and intercept line feeds, so that whenever a line feed comes up it is followed by a CTRL-N and the ENHANCED mode is restored. The listing is at the end of the article. If you do not know how to put it in read BEGINERS PAGE in HARDCORE no 2. When you have typed it in, to save it type:-

```
BSAVE ENHANCE,A$300,L$2F
```

and

press return.

When you wish to use it, at the PRINT CONSTANTS STAGE, change

PRINTER ADDRESS to 300 and make sure that the sum of the right and left margins is less than 68, as the Enhanced mode will only support 68 characters.

You should be alright, except that I have found it gets lost sometimes in moving back and fore from the printer to the editor (does anyone know why?), with the result that it just hangs; so I adopt this procedure. From the PRINTER MENU, type CTRL-D to allow commands to go through DOS, and now BLOAD ENHANCE, then quit back to the menu and carry on as normal.

It is as easy as that. When the APPLEWRITER file is being printed, if you watch the screen then a row of inverse Cs will come up. Ignore these, they are not printed. Then after each line, as the CTRL-N is sent, there is a brief flash of an inverse N on the screen.

THIS IS NORMAL EPSON PRINTING TO ILLUSTRATE THE DIFFERENCE.

\*300.32F

```
0300- 48 A9 11 20 00 C1 A9 0F
0308- 20 03 C1 A9 0E 20 03 C1
0310- 68 48 C9 0D F0 04 C9 8D
0318- D0 05 20 03 C1 A9 0E 20
0320- 03 C1 68 A9 11 85 36 A9
0328- 03 85 37 60 00 00 00 00
```

```
0300- 48 PHA
0301- A9 11 LDA £$11
0303- 20 00 C1 JSR $C100
0306- A9 0F LDA £$0F
0308- 20 03 C1 JSR $C103
030B- A9 0E LDA £$0E
030D- 20 03 C1 JSR $C103
0310- 68 PLA
0311- 48 PHA
0312- C9 0D CMP £$0D
0314- F0 04 BEQ $031A
0316- C9 8D CMP £$8D
0318- D0 05 BNE $031F
031A- 20 03 C1 JSR $C103
031D- A9 0E LDA £$0E
031F- 20 03 C1 JSR $C103
0322- 68 PLA
0323- A9 11 LDA £$11
0325- 85 36 STA $36
0327- A9 03 LDA £$03
0329- 85 37 STA $37
032B- 60 RTS
```

## JAILBREAK

COME OUT FROM BEHIND THOSE HI-RES BARS

Now you can run any hi-res program written for your Apple on your 2020 without the annoyance of the 40 vertical bars. Chess and complex games are a joy to behold.

- \* Saves having to modify programs to 2020 standard (no mean task with complex games which can be over 16K bytes long).
- \* No soldering required - Fit in five minutes by following the simple instructions.
- \* Switchable between Apple and 2020 nodes.
- \* Compatible with Apple PASCAL and Microsoft Softcard BASIC.
- \* Gives 280X192 hi-res dot screen like an Apple (the 2020 normally gives a 360X192 hi-res dot screen). Text and lo-res are unaffected.
- \* Does not support hi-res colour. Colours produce various shades of grey on a b&w display.

**£33.95 inclusive.**

**MICROSOURCE**

1, BRANCH ROAD, PARK STREET, ST.ALBANS.

TEL: 0727 72917 & 09273 75093

## EAMON REPORT.

by David Row

This is a report on the Role - Playing game Eamon, details of which were given in Hardcore No 2. I am going to do this in two parts, first the game system and secondly how the game plays.

The first problem with the game system comes with determining the attributes of characters. The original role - playing game, Dungeons & Dragons, used six of these, and later versions introduced supplementary characteristics. Second generation games used more. Eamon tries to get by with only three, Hardiness, Agility and Charisma. This is, I feel, quite inadequate. The purpose of these is to build up a person but how many people can you describe completely in terms of these three attributes?

The main excuse for reducing the number of attributes in games is the amount and complexity of bookkeeping that it introduces. But in the Apple a bookkeeper is readily available - it should be utilised to the full.

The second problem occurs with the combat system. This is not the system as such, for although I personally don't like it, this type of system has its followers. My big objection is the lack of information given to the player when he is in combat.

Someone whose occupation is fighting things has a pretty good idea of how much damage the blow he has just struck has done, yet Eamon refuses to tell you this. It also doesn't say how good the blow was, i.e. did it miss by a mile or was it only deflected by circumstances which seem to indicate the unusual (perhaps magic armour) ?

While this complaint may seem petty, anyone who has played these games with a human gamesmaster will tell you otherwise. This type of situation is very rarely found in human - run games, and the main reason for this is that so much aggravation is caused by it. Arguments that this system is more realistic (now there's a leading statement in a fantasy game...) are, I believe, completely wrong - the opposite is the case.

My third major complaint regarding the game system is the use of the attribute Charisma. This is meant to indicate how well the character gets on with people (although this particular characteristic is notoriously difficult to pin down). My main objection is that is grossly overweighted in importance; it affects the price you pay for equipment by large percentages, how people react to you, and seems to have a tendency to dominate most of the game.

The odd thing about this is that although no allowance seems to be made due to attempted player actions, there is a permitted command 'Smile' which you would expect to have some

effect on reactions. In short it doesn't matter how you act, it's all fixed at birth.

To summarise my feelings towards the Eamon game system; I don't like it much. It is in general primitive, and not amenable to intelligent play. How open it is to expansion I am not yet sure - I have yet to see the design module - but at this moment I am not hopeful.

How does Eamon play? In spite of what I have just said, not too badly, as long as you do not expect too much.

The game appears to run without any untoward tendency to crash, and the bulk of the actions you are likely to try are included in the command set (a situation some commercial designers of Adventure games might do well to take note of).

Apart from the nice picture at the start, Eamon does not use any graphics at all. This seems a pity in view of the Apple's excellent capabilities in this respect, and once again you are reduced to scraps of paper to map your path on. The argument for this is presumably that it is traditional - D & D made you map, so all other games should. I do feel that it is unnecessary in general though - the object of these games should be to enjoy a creative fantasy, not to learn how to draw maps!

The behaviour of the monsters you meet in the cave (another traditionalism - what's wrong with an Adventure in the open air?) is depressingly random. It doesn't seem to matter what you do, it's all fixed by your charisma and a dice roll. Perhaps because of the lack of any means of affecting combat other than by running away or hacking at your opponent, the lack of any sort of display of the combat situation is not particularly oppressive, it's simply that it wastes the Apple's capabilities again.

My final complaint in the Cave is the existence of possible 'instant death' traps (I won't say what they are). I dislike these on principle, since they directly encourage either reckless play (characters who try and do anything) or play so cautiously as to spoil the fun (it's supposed to be a Game, not a crossword puzzle). These tricks are questionable enough under a human gamesmaster, where it is possible to find out a lot by asking general, then specific questions, but with a computer this is not really on. I'm sorry but I feel this type of effect hardly ever adds anything of value to the game, and normally does the opposite.

In conclusion I feel I cannot recommend Eamon unreservedly. It's not a bad game as such; partly it's just that the state of the art in this field is steadily moving on, and Eamon is decidedly 'first generation'. If you don't expect too much, however, it gives reasonable value for money at £4. It would in fact make quite a good introduction to this type of game, since it is simple to master.



**THE ANDROMEDA 16K RAM  
EXPANSION BOARD & THE I.T.T**

By Jeremy Ensor

Having acquired an Andromeda Ram Card primarily to be able to use and write Integer programs without having to boot the Relocated Integer disk each time, I rushed home to fit it and try it out. Imagine my acute dismay and frustration to find that, when I tried to fit it (as per the accompanying manual) in slot 0, the card wouldn't fit as the switch on the end didn't coincide with the vertical gap on the back of the I.T.T!

Having read the rest of the manual I decided to try fitting the card in either slot #2 or slot #4 as hinted at. My first discovery was that it wouldn't fit in slot #2 for the same reason as before but it did fit in slot #4. I then disconnected the relevant Ram chip from the motherboard, fitted it on the vacant slot on the card and connected the card to the now vacant Ram slot. This didn't work either as I soon found that I couldn't boot any of my disks.

At this point I was furious and was mad enough to give my I.T.T. a swift kick where it hurts most-why should there be such a stupid little difference in design between it and the Apple? Logic and visions of my bank manager finally won the day and I decided to ring the long suffering David Bolton who suggested what should have been blindingly obvious, that I remove the switch and put it back on the other side of the card. As I'm not the greatest wielder of a soldering iron I had to think long and hard about tackling the job as it would obviously void any accompanying warranty, but in the end my spirit of adventure prevailed.

It actually turned out a lot easier than I expected it to do though I should point out here that it would be much, much easier and quicker to have someone helping you. The problem lies in the fact that you have to try and unsolder the four feet all at the same time in order to remove the switch unit. I did by stages with a screwdriver used gently as a lever to free the unit. Once you have removed the unit you turn it over and solder it back onto the corresponding pins on the reverse side of the card. Remember that the switch positions are now reversed.

I would strongly recommend that you buy an expansion card rather than an Integer card as it gives you more memory, gives you Integer and allows for further use of Fortran and other languages. To make further use of the card I have loaded the hello program from the System Master, removed the "greeting" and replaced it with my own "greeting". I then initialized my new

disk with this amended program and then copied the "Intbasic" binary file from the System Master onto the new disk. I can now boot Integer with this new disk without having to use the System Master each time.

**DO YOU KNOW THAT?**

**1. RENUMBER PROBLEM**

There is a problem with RENUMBER on both 3.3 and 3.2 DOS masters. If you try to renumber a line like:-

10 LET A= B\* 10

it will come out as :-

20 LET A= B\* 20

To fix it :-

DOS 3.2

For RAM APPLESOFT:-  
LOAD RENUMBER  
POKE 14343,172  
POKE 14343,171  
SAVE RENUMBER

For ROM APPLESOFT  
LOAD RENUMBER  
POKE 4815,172  
POKE 4816,171  
SAVE RENUMBER

DOS 3.3

For RAM APPLESOFT  
APPLESOFT  
LOAD RENUMBER  
POKE 14316,172  
POKE 14317,171  
SAVE RENUMBER

FOR ROM  
LOAD RENUMBER  
POKE 4789,172  
POKE 4790,171  
SAVE RENUMBER

Do not run them before saving or doing the POKES of course.

**2. DISK DRIVES**

During 1980 there was a change in the part of the disk drive that holds the disk whilst it spins. A consequence of this was that the disk would slip whilst writing or reading ( especially writing) and you lost your data. Apple dealers should have told you and replaced the offending part. If you had disks with reinforcing rings in the centre there would not necessarily be a problem, but it could be intermittent. If you had a drive serial numbered between 96,000 and 148,200 (but excluding 123,000 to 132,000) have the drive checked.

Any more problems or solutions -> BASUG PO BOX 174 WATFORD WD2 6NF

# Pippin's Page ~~~~

~~~~~  
 Edited for younger readers by Vernon Quaintance  
 ~~~~~

Last month we took a look inside the Apple. This month we will start learning how to write our own programs. Most of you will already be familiar with some of the commands, since you will have loaded and run various games, however I will include these at an appropriate point in the course. What I hope to teach you will normally be true for any computer which you may meet - not just the Apple.

There are two ways of using the computer. The first is as a very large and expensive calculator; this method is known as 'immediate mode' and we will start there.

Try typing `2+3=` followed by a `<RETURN>`. (that is pressing the key marked 'return'). What happened? You didn't get the answer that you expected did you? In order to see the results of a calculation we must tell the Apple to `PRINT` them to the screen. Let's try that calculation again; type:

```
PRINT 2+3 <RETURN>
```

Note that `<RETURN>` means press the key marked return. We must do this every time we get to the end of an instruction. The Apple sees the `<RETURN>` as the indication that you have finished typing in something and that it is to go away and do what you have told it to. We will not tell you to press `<RETURN>` in future, just do it automatically when you finish typing a command.

The result, 5, was printed on the next line. Now try a few sums for yourself. You can add using `+`, subtract using `-`, multiply using `*` or divide using `/`. Notice we do not use `X` to mean multiply. If you understand squares and cubes then you can also try `PRINT 3^2` and other things like it. The `^` symbol means 'raised to the power of'.

Whilst you are trying these simple additions etc, try also doing more than one at a time, for example try `PRINT 2+5-4` and also `2+3*4`. You will probably have expected a different answer to this last one. The answer that the computer gave was 14 but did you think that it should have been 20? The reason for this difference was that the computer does multiplication and division before it does addition or subtraction. If you want the addition done first then you must use brackets, eg. `PRINT (2+3)*5`.

We have tried doing some simple sums now and have learned the command `PRINT`.

Supposing we want to print out some words on the screen, what should we do? Go on, try something. There is nothing that you can type at the keyboard which will harm your computer.

What was that? Did the Apple respond as you expected or did it reply with a 0 instead of the words you wanted? If you got a 0 (zero) then this means that you forgot something. What?

To print out words one must enclose them in quotation marks, eg. `PRINT "HELLO, THIS IS YOUR APPLE COMPUTER."`

Now let us try something using both of these. Type:

```
PRINT "2 + 5 * 6 = ";2+5*6
```

You will see both the sum and the result printed out together on the same line. The semi-colon (`;`) in the print command line means 'type the next bit immediately after the previous bit'.

Now try this one:

```
PRINT "PETER","FRED","MARY"
```

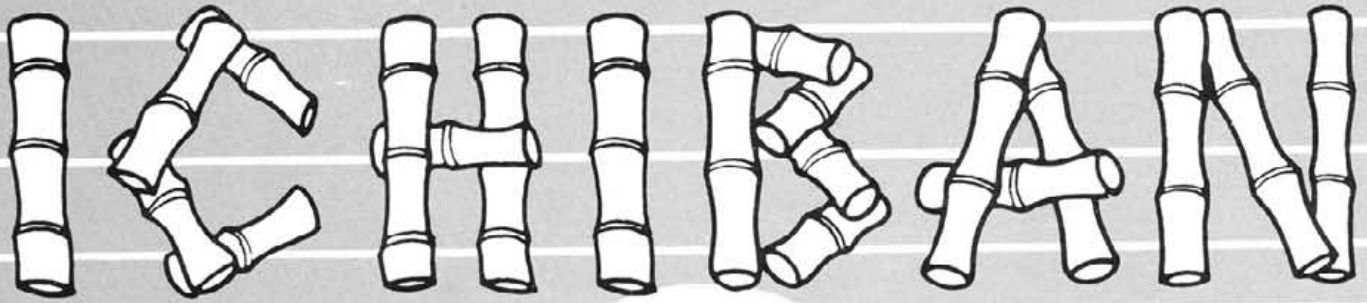
Notice how the three words are spread across the screen. This is because we used a comma (`,`) in the print command. When used in a print command, the comma means 'type the next bit in the next column'. Each column goes one third of the way across the screen.

I will leave you to experiment with the `PRINT` command. Try using it by itself without anything following it. Try various combinations of commas and semi-colons in the command. In the next issue we will learn about the other method of using the computer, known as 'deferred execution mode'.

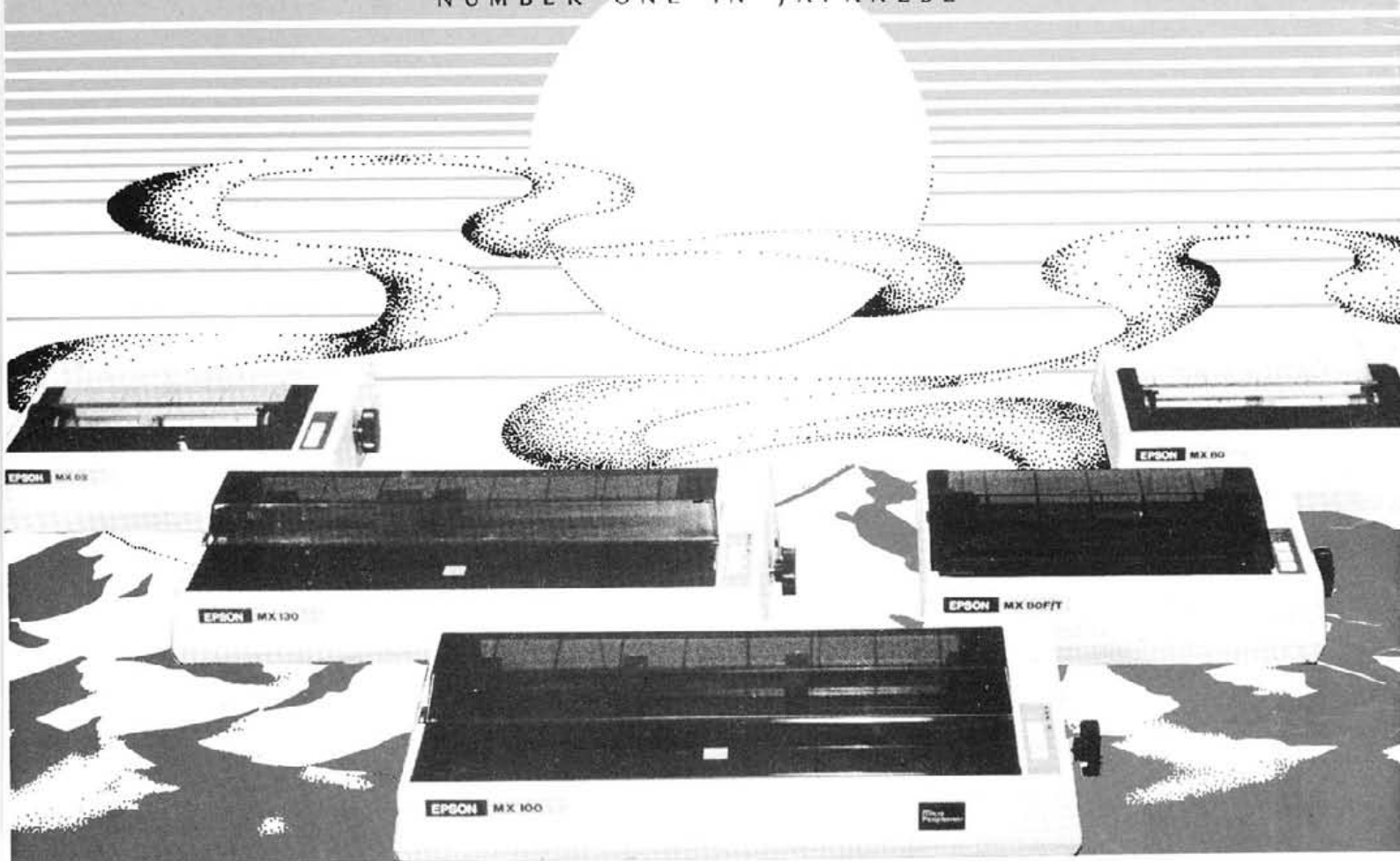
Happy Apple Programming!







NUMBER ONE IN JAPANESE



Highest possible quality + lowest possible cost = EPSON + MICRO PERIPHERALS  
for the best of both worlds.

	Matrix			Paper Handling		Speed		Max. Paper Width		High Res. Graphics & Italics
	7 x 5	9 x 7	9 x 9	Friction	Tractor	80 CPS	135 CPS	10"	15 1/2"	
MX80 T			●		●	●		●		
MX80 FT/1			●	●	●	●		●		
MX80 FT/New Type 2			●	●	●	●		●		●
MX82			●		●	●		●		●
MX100			●	●	●	●			●	●
MX130		●		●	●		●		●	●

The above machines have many more features including interfaces for Apple, PET (with PET Graphics), TRS80, Sharp, NEC, Hitachi, Nacom, Acorn, Super Brain, Video Genie, BBC Micro etc, some have correspondence quality printing and multiple character sets including international languages. Ring Ian today for full details and specifications and printout samples. All machines usually ex-stock with next day delivery plus 12 months no-quibble guarantee.

Wholesalers and Distributors of quality Japanese Micro Products.

# Micro Peripherals Ltd.

61 NEW MARKET SQUARE, BASINGSTOKE, HANTS. Telephone: 0256 56468

Europe's leading Epson Distributor and Supplier. Trade Sales - 0256 56469

JAPANESE OFFICE: 101 Abe Bldg. 4F, 2-42 Kanda Jinbocho, Chiyado-ku, Tokyo, Japan.