hard core

THE JOURNAL
OF THE
BRITISH APPLE
SYSTEMS
USER GROUP

APRIL 1982 £1 VOLUME 2 No. 2

P R E S T E L
GOOD AFTERNOON THU 25 MAR 1982 20:21
BASUG
ENTERPRISE COMPUTER
NEW MESSAGES FOR YOU,KEY *930#

Welcome to Prestel

YOU LAST USED THIS COMPUTER
ON SUN 21 MAR 1982 AT 17:02

KEY # TO CONTINUE

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

BOOKSbooksBOOKS

BOOLEAN ALGEBRA FOR COMPUTER LOGIC

by HAROLD ENNES £6.65

Not a book specifically for programmers, but also for digital circuit design. Deals with BOOLEAN ALGEBRA in a lucid way; since Boolean Algebra is only an extension of conventional algbebra, you need only start with such a general understanding and go on through this self instructional book through symbolic logic, algebra and Venn diagrams. There are many examples and excercises with answers.

ASSEMBLY LANGUAGE PROGRAMMING

by Inman and Inman £10.50

A beginners route from BASIC- not the final word but a gentle introduction.

ASSEMBLY LANGUAGE

by Randy Hyde £11.95

At the next level (not a beginners book) a book that helps you program and gives you some useful routines, as well as being one of the few places you can learn to use the SWEET 16 routines.

BUSINESS PROGRAMMING

BASIC BUSINESS SOFTWARE

by

E.G.BROONER £8.70

A general introduction to all aspects of computer software for business, with programs analysed, chapters on information storage, word processing, and simulation as well as inventory control payroll etc.

APPLE BASIC FOR BUSINESS

by Parker and Stewart £12.50

This is a practical guide to writing business software on the APPLE. The step by step problem approach leads you through all types of file access in a logical manner using problems you are likely to find useful in the programs you need. It is also useful for those needing to handle files in other contexts.

APPLE BASIC DATA FILE PROGRAMMING

by

Finkel and Brown £10

This is a self teaching book, even more modular and program based. It is one of the Wiley Self Teaching Guide series.

• MICROSOURCE

1, BRANCH ROAD, PARK STREET, ST. ALBANS, HERTS, AL1 4JR.

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
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 Editorial

5 Chairman's Corner

6 Working with the Apple /// — Roger Swaine

11 Mnemomic Puzzler — D.R.Daniells

12 Education Column — Norah Arnold

13 Another First for the Apple — Dr.B.E.Weller

14 More on Apple-Writer & Go-Between — Ian Trackman

21 Two Programs Co-Resident in Memory — Les Budgen

22 Pippin's Page — Vernon Quaintance

23 Local Groups

24 Repeat...Until — Bob Raikes

24 Basug Toolkit No.1 - Machine Code Print Routine — Cliff Wootton

26 The Nicelister — T.Tse

31 Compu-Read — Reviewed by Norah Arnold

32 CS Textmaster Word-Processor — Reviewed by G.P.Ognibeni

34 Type-Right Lower Case Modification — Reviewed by Fran Teo

35 Apple Pilot — Reviewed by Leo Crossfield

38 Jailbreak (ITT Graphics Modification) — Reviewed by Chris Chapman

41 Using 6502 Assembly Language — Book Review by Ian Trackman

42 Structured Basic and Beyond — Book Review by Michael Tickle

42 Astro-Apple — Reviewed by Chester Kemp

43 Reader's Letters

EDITORIAL

Once again I write these notes in haste, with the magazine at last ready for the printers, and the deadline, as usual, missed. From out there, it may be only once every two months that you receive the magazine, but from in here it seems as if we put one together every couple of weeks! Hopefully things will improve from next month, with some daytime effort available (see the Chairman's Report herein). One certain change is that we will be able to more positively 'market' Hardcore, so the advertising revenue should increase. This will enable us both to increase the size and quality of the magazine still further, and to start paying a (modest) page rate to contributors.

April will see BASUG at two shows - the 3rd London Computer Fair at the North London Polytechnic on 15 - 17th April, and the Computer Fair (a bit naughty of Practical Computing to use that name!) at Earls Court on 23 - 25th April. We also expect to be at Windfall's "Apple'82" exhibition in Slough on 4-6th June, although this has yet to be confirmed. Finally, on Saturday & Sunday 11th & 12th June, we shall be holding our A.G.M. and Convention at Wallingford. We propose that the weekend be made up of a large number of events, each catering for a small group of people. In this way people can better get to know each other, which is the main purpose of the weekend.

At the first of the above exhibitions (15-17th April) PRESTEL will be launching their TELESOFTWARE project, featuring software for the Apple, which has been supplied mainly by BASUG. Which leads me into the main news of the month.

One of PRESTEL's main 'pushes' of 1982 (Information Technology Year) is to develop Telesoftware, and as part of this they have approached the various national user groups for assistance. BASUG has reached what we think will be mutually very attractive arrangements whereby we can make available to our members a comprehensive range of PRESTEL services, as follows:-

a)Telesoftware- As much as possible of BASUG's software library will be put onto PRESTEL, to ease the cost and effort of distribution by more conventional (and old-fashioned!) methods. Although some programs will be available to all PRESTEL users free, to act as a 'shop window' for BASUG, in general there will be a small charge to cover the group's costs, as there is for disks. This charge will usually be higher for non-members. Although we can (and indeed in some cases will) restrict access to programs to BASUG members only, the committee feels that in general, provided our costs are covered, it is in our best interest, as well as in the spirit in which the programs are made available to us in the first place, to distribute programs as widely as possible. No doubt this issue will lead to some lively discussion.

b)Closed User Group- BASUG members will belong to a PRESTEL 'closed user group' giving access to various information, programs, ordering facilities, etc, to which access is barred for non-members.

c)Order Facility- Members will be able to order BASUG products, book places on courses, etc, via PRESTEL

d)Database- Details of forthcoming events, an index to the software and book/magazine libraries, an Apple bibliography (this latter provided some of you out there help to compile it!), etc, will be available on PRESTEL.

e)An electronic version of Hardcore ('Softcore', perhaps?) will be developed, both by providing information pages specially for BASUG members, and by cross-referencing to existing pages of interest.

f) We hope to be able to provide a 'BULLETIN BOARD' facility, whereby members can interactively use PRESTEL to communicate with each other.

g) We expect that PRESTEL will make available on loan to the group for a limited period a small number of PRESTEL ADAPTORS. The first priority for these will be that local groups have them available at meetings for demonstration and for downloading software. Within this constraint we intend to lend them out to individual members for short periods, subject of course to suitable undertakings concerning responsibilty for the safety and return in due course of the adaptor, and for the payment of the Prestel Access Charges incurred.

We should stae here that there will be no net cost to the group in respect of these facilities. The already established principle whereby any minority activity (Prestel, Equipment Sales, Courses, etc) is self-funding will apply.

If you're interested in these developments, either write in for a PRESTEL 'Information Pack' which will give you all the information you could want, or come along to one of the exhibitions. If you already have access to PRESTEL, then we need to know your user number, and which PRESTEL computers you are registered with, so that you can be included in our Closed User Group.

These developments are on the 'leading edge' of (to coin a phrase) information technology and all the people who've been involved so far are very excited at the implications of what's happening. PRESTEL has been rather slow to take off so far but this time, by tapping the "innovative, technologically-aware" population of microcomputer users (and there are some 400,000 of us in the U.K. compared with 15,000 PRESTEL users) it looks set to go somewhere.

Your editor, incidentally, has two vested interests in these developments. Firstly, the software to upload and download Apple Telesoftware has been written by myself and John Sharp. Secondly, I see this as a way to spread the load and widen the involvement (and if we are to be truly a national group this is essential!) in the production of 'Hardcore'. We can already transmit BASIC programs via PRESTEL, and it's just a matter of sorting out a protocol to do the same with Applewriter files. The next issue of Hardcore will contain articles that have been transmitted in this way! The much vaunted, futuristic concept of sitting at home, typing an article and editing it, and then 'sqirting' it down the 'phone to a central location for printing is no longer in the future. For us, its here, now!

BASUG's first course took place in March - a Beginner's Machine Language two-day course attended by ten people, and presented by Ian Trackman. First reports suggest that the participants got a lot out of the course, and we hope to repeat it from time to time. If you would like to attend, let us know and we'll keep you informed about dates. The PASCAL course that didn't take place last year will take place this year, in July. Members should all have received details already - if you haven't then write in.

Finally, an apology. 'The Last 0.8', an article in the last issue of Hardcore, was wrongly credited to Cliff Wootton who rang me up to say that he had no recollection of writing it, but it was good, wasn't it. It was in fact written by Ted Lepley. Sorry, Ted!

David Bolton

BASUG?

For those readers who aren't members, it stands for British Apple Systems User Group. Membership is £10 per annum, and there is a one-time joining fee of £2.50. Just send a cheque to our PO Box address - we'll do the rest.

CHAIRMAN'S CORNER

This is the first chance I have had to write in Hardcore this year, which is evidence, I hope, of the busy life that faces anyone who decides to enter the microcomputer world to make a living - and I am pleased to be able to report that even with the turmoil in the market place at present, with each manufacturer bringing out their latest offerings, our favourite machine is still offering a versatility which is among the best in the business. If you really must have the third-degree burns that can result from dabbling in the white heat of technology, you can plug the latest 16-bit microprocessor into your Apple anyway, and enjoy the best of all worlds!

I have been told by the Editor that space is limited this time, so straight to the main points - firstly, renewals of memberships have been coming in thick and fast, although despite our best efforts to inform everyone of the need to renew, there are still some latecomers! As you know, this will eventually cost you money, and I am sure that no BASUG member will want to miss out on the goodies that the Group provides.

Secondly, as has been evident over the past few months, BASUG has been so successful that the purely voluntary efforts of your Committee simply cannot cope with the flood of correspondence, and it must be said, the odd complaint about something or the other. Steps have been taken to do something about this! John Sharp, who since BASUG started has been putting in more than a full week's work in his spare time each week on the Group's behalf, has recently ceased gainful employment to concentrate upon developing his own business. The committee, keen to protect members' interests, have seized upon this chance, and have decided to pay John to work one day each week on behalf of the Group. He is not going to get rich this way, but it seems to the committee that this is a good way of ensuring that the management of the Group is put on a more businesslike footing, without the problems that ensue when one tries to do things during the day while actually holding down a job as well. I hope that you will all join us in wishing John success in his own enterprise, and in thanking him for the unselfish way he has helped to create BASUG. As a further step in the direction of efficiency, we have converted the BASUG membership records to a more sophisticated Database, and we hope that from here on in our publications and correspondence with you will be tidier!

Frank Kay

WORKING WITH THE APPLE ///

by Roger Swaine

Having had the opportunity to work with the Apple /// for some weeks now, I am beginning to get the hang of it, and this article is for those who are interested in this intriguing new machine

It must be pointed out right at the outset that this machine is not aimed at the hobbyist. It's very much a small business computer, and in this article I will try to look at it from this point of view.

Well, how does the machine compare with the Apple II? Lets look at the hardware first:-

(i) The Apple /// has an upper and lower case keyboard, with the case change being through the shift key i.e. a standard typists keyboard (advantageous if the machine is going to be used by typists as a word processor). There's also a numeric pad (with its own 'Enter' key) for fast data entry. All keys repeat, while the cursor movement keys, if pressed hard, repeat at treble speed.

(ii) The standard monitor is styled to sit on, and blend with the rest of the computer. It has 80 columns with genuine descenders on the lower case letters. The 80 columns makes the machine easier to use for word processing, and also makes for easier link-up to mainframes (where 80 columns is standard)

(iii) The machine can generate, and receive, the whole 128 ASCII character set (for easy compatability with other machines)

(iv) The processor is still 6502 - based but runs effectively about 40% faster, and can address more memory (128 kbytes at present, though a 128K add-on card is in the offing giving a total of 256K) However, about 64K of memory is occupied by the operating system and interpreter so that you have 60-70K available to the programmer.

(v) There is a built-in disc drive, and a socket at the back to plug in the first external drive. The second external drive plugs into the back of the first, and so on, up to a maximum of 4 drives in all. There is no provision for tape.

Now, you could probably get most of the above on an Apple II by hooking up various pieces of hardware, but by the time you had done so you would have paid almost as much as the Apple ///, and you would also have the risk of incompatability when using packaged software. In addition you would not have got the benefit of the really big change, which is in the software.

This machine is in no way a hotted-up Apple II; it is a very different animal since it is almost totally "soft". Tht is to say, when you switch it on there is no operating system nor language in it. In fact, the only thing in ROM on the machine is, I believe, about 4K of diagnostics. This has the disadvantage that your boot diskettes must contain not only your program but your operating system and your interpreter. Since the last two can take up about half the space on a diskette, your programs and data files are correspondingly restricted in size if you want to keep everything on the one diskette. Set against this, however, are the following advantages:-

(i) Since what you load in configures the machine, you can write your programs in any language you like, provided you load in the corresponding interpreter. Thus, for example, the standard utilities diskette supplied with the machine is in Pascal code.

(ii) As upgrades to the

operating system and/or languages become available, they can be incorporated by copying them into your boot diskettes: you do not have to change the hardware.

This "soft machine" approach means that the Apple /// is more like a mini/mainframe computer than the conventional micro, and this is carried through into the control of the hardware (i.e. the console, the printer, the disk drives, and any other peripherals), all of which is controlled through the use of software drivers.

Supplied with the computer is a diskette with a variety of drivers: you copy onto your boot diskettes the drivers that you need, reconfigure them as necessary (and there is a variety of options) and from then on you don't have to worry about them. You can send data to the screen, to the printer, or to a disk file - it is all the same command, just a different destination. This use of software drivers also means that you do not necessarily have to put cards into the machine to attach peripherals(which is probably why the machine only has four slots). In this way the software looks after the hardware, and the programmer does not have to worry about it. Farewell to the Control "D"s!

Launched at the same time as the Apple /// is the "Profile" hard disk, and the Apple /// operating system has obviously been designed with this in mind, as it is possible to group files into directories, and instead of specifying a file just by its name you specify its "pathway", which defines the directory, sub-directory (if required), and then the actual file. Since a hard disk could contain a large number of files, this facility for grouping related files together is obviously sensible. A nice point is that you do not have to tell the machine which drive a particular file is in - as long as it is in one of the drives the machine will find it!

Finally in this section, for those who have software written for the Apple II it should be pointed out that, by the use of a special program supplied with the machine, it's possible to turn the /// into a 48k Apple II & run your existing programmes. The only limitation is that you cannot use slots 1 to 4 in your fictitious Apple II, and you may have problems if your program uses the games paddles.

BUSINESS BASIC

The standard language supplied with the machine is Business Basic, which is differently orientated from Applesoft in that it reflects the requirements of the business user, who requires that his programs should be easy to write, easy to read and easy to maintain. To this end, variables can now have names up to 64 characters long, so that you can have variables with meaningful names e.g. "Itemcode", "Price", "Packsize". In addition to the usually string, integer, and real variable types there is also a new "long integer", that enables very large numbers (up to 19 digits) to be handled accurately. Conspicuous by their absence are PEEKs and POKEs: machine code programming is rather more complex on the Apple III with its enhanced addressing, and not advised, and (for the reasons given above) their use is not normally cost-justified for the business user.

File I/O is improved, you can have random access to binary data files to speed up file access. (Incidentally, there seems to be an error in the manual here; it appears to say that each successive "Read" statement reads from successive records, but in my experience with fixed record - length data files I have always had

to specify the record number to get it to read properly.)

Formatting commands have been much improved by the introduction of the "Print using.." and "Image" statements.

One particular bugbear of mine about the Apple II however still remains in the Apple ///; this is that there is no distinction between "saving" and "replacing" a file, so that if you accidentally save a file under a name that already exists, it will wipe out the old version without warning you (Incidentally, the new Visicalc that runs on the Apple /// does warn you, so it must be possible !)

VISICALC ///

This is pretty much the same as the DOS 3.3 Visicalc except that you can see 80 columns across. You also have 66K of available space, though the maximum number of rows and columns is no larger. In order to fit in with the new operating system, you have to read the drivers file and configure it to suit your hardware and then write the new file back onto the disk. Therefore the disk is not write-protected. However, you are provided with a back-up diskette.

APPLEWRITER ///

This is being sold as a word processing package with the implication that the Apple /// plus Applewriter /// might be a better buy than a purpose designed word-processor. Quite frankly, viewed in that light it is a bit of a disappointment.

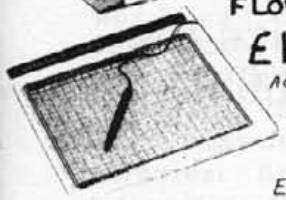
As with Applewriter I the final formatting of the document is carried out at the printing stage and although this is adequate for text that is blocked up against the margin, it is not adequate for most business typing, where more complex formatting is required, and although you can get this by embedding formatting commands in the text, they complicate the input of the text to the point where it would be quicker to do it on an ordinary typewriter where you can see what you are doing. You really ought to be able to see the result of any commands immediately reflected in the text on the screen. There is no horizontal scrolling, while vertical scrolling is jerky and unpredictable. The cursor displaces the line that it is on so that all text to the right of it is moved over one place, and when you try to move the cursor vertically up or down it jumps around like a demented flea. All in all, and bearing in mind the flexibility of the machine, it looks to me like a bit of a rush job. it must be possible to produce a much better w.p. package than this.

Now, if the above seems harsh, I should point out that I am comparing with a purpose-designed word-processor. If you have bought the Apple /// already howevever, and are looking at the cost of just the Applewriter software then it probably is worth having, though not necessarily for use a a word-processor. This is because, on the Apple ///, the Applewriter reads and creates straight forward text files, and therefore can be used to create and edit Basic EXEC files, Visicalc /SS files and (I am pretty sure, though I have not tried it) DIF files, as well as being used as a fairly rudimentary word-processor.

introductory offer:

BUSINESSMAN'S SPECIAL
• Apple II plus 48K
• Apple Disc Drive II and interface DOS 3.3
• 12" Green High Resolution Monitor
• Visicalc 3.3 Software
£1075

QUALITY FLOPPY DISKS
£14.50/10
ACCUTRAC/WABASH/MEMOREX etc

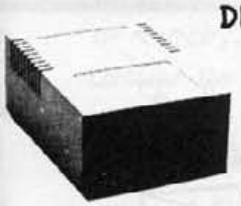


GRAPHICS TBLT £375
EUROCOLOUR CARD £69

SILENTYPE PRINTER £189
APPLE PASCAL LANGUAGE SYSTEM £235
SPECIAL DELIVERY SOFTWARE AT SPECIAL PRICES

BMC 12" GREEN MONITOR £119
VIDEX 80 COL CARD £190

DISCOUNT PRICES – NOT SERVICE!
ROMWRITER, ROMPLUS, CLOCK CARD, SPEECHLINK AVAILABLE



ADD-ON DISK £285
3.3 DISK DRIVE WITH CONTROLLER ONLY £325
MANY OTHER BARGAINS, CALL FOR DETAILS NOW

apple computer
Authorized Dealer
Accept No Less

48K £618

FULL WARRANTY ALL PRODUCTS

EPSON MX82 £325
MX80 F/T £330, TYPE 2 £365 MX100 £475

ALL PRICES EX-VAT & DELIVERY

MASS MICROS
TEL WELWYN GARDEN (07073) 31436/7
WELLSON HOUSE, BROWNFIELDS
WELWYN GARDEN CITY AL7 1AR

Wida Software

Specialists in Educational Software for Schools

APFELDEUTSCH.....£ 120.00
CAL course in German; Beginners to O'Level and beyond. Textbook: Workbook: 6 Language Lab cassettes: 9 Apple disk of teaching and testing routines.
Apple only. (20% discount for schools)

TEACHER'S TOOLKIT.....(Apple disk) £ 20.00
(Pet cassette) £ 15.00
Starter pack for building up library of tests. No knowledge of computing needed. Suite of five programmes; Tester, Testmaker, Editor, Multiple Choice specimen, Directory of tests

LERARENPAKKET.....(Apple disk) £ 25.00
Teacher's Toolkit as above but with on-screen instructions in Dutch

GERMAN ROUTINES.....(Apple disk) £ 15.00
(Pet cassette) £ 10.00
Stand alone teaching and testing routines – article and adjective endings, word order, pronouns, etc. Send s.a.e for details

SHAPE MANAGER.....(Apple kit) £ 59.65
The easy way to build up shapes and use them in your program. Does for shapes what a word processor does for words. From Sinta Software

TYPE-RIGHT.....(Apple kit) £ 46.50
Lower case word processing with Apple Writer. Full shift key operation. Plug-in fitting with disable switch. Manual, fitting instructions, software on disk.

APPLE PILOT.....(2 disks) £76.00
The ultimate language for teachers; mix sound, graphics and text on-screen in your teaching programmes. And much else. Disks and manuals

ARISTOTLE'S APPLE..... (Apple disk)£ 20.00
Authoring system; easy to use multiple choice matching tests. Includes alternative answers.

LUDINSKI'S LOGS..... (Apple disk)£ 20.00
O'Level revision of logs problems. Teaching, testing and scoring. Correct solution rewarded by 25 seconds of arcade game.

COMPUTERS FOR KIDS.....(book) £ 4.00
A Basic programming manual written just for kids. Special section for teachers and parents. by Sally Greenwood Larsen Apple II plus edition

TIMETABLING	Let Keith Johnson – England's
TIMETABLING	foremost authority on time-
TIMETABLING	tabling guide you through
TIMETABLING	next August's trauma!
TIMETABLING	Book with CBM listings £8.95
TIMETABLING	Apple software on disk £30.00
TIMETABLING	Software on cassette for
TIMETABLING	380Z, TRS80, CBM £30.00

All prices inclusive of VAT postage & packing free

Contact Wida Software
The Red House
2 Nicholas Gardens
London W5 5HY
Tel (01) 567 6941
or (062 882) 5206

VISA

kj

apple computer
Sales and Service

ZOOM GRAFIX™

kay-jay studios
135 COLES GREEN ROAD LONDON NW2 7HN
TEL: 01 450 6993

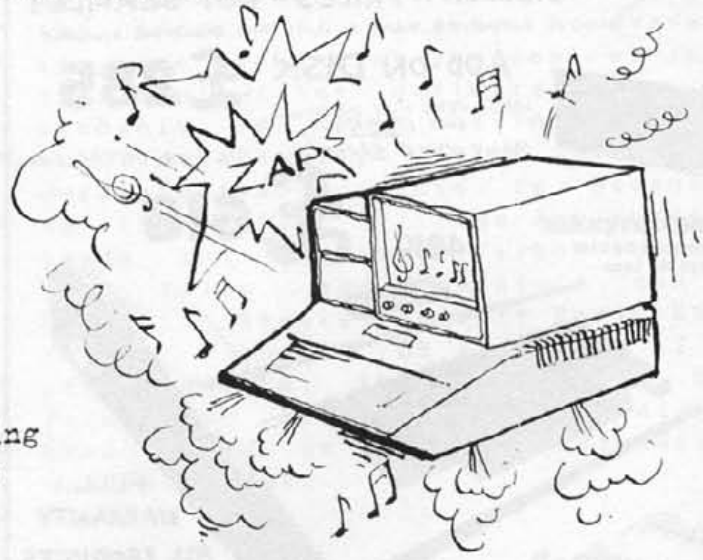
Finally you can realize the FULL graphic capabilities of your printer.

- Any area of the Hi-Res screen can be printed as large or small as you like.
- Zoom "Window" feature allows you to frame and see the specific area of the Hi-Res screen to be printed.
- Picture can be printed horizontally or vertically.
- Margins can be completely adjusted for custom placement of picture.
- Form feed and line feed can be controlled from the keyboard.
- Prints either Hi-Res screen 1 or 2.
- Prints charts, graphs or pictures.
- Menu driven . . . extremely easy to use.

ZOOM GRAFIX works with any of the following printers which have graphics capability: Anadex, Apple Silentype, Epson, IDS, Nec and Pro-Writer.

Only £29.95 inclusive of V.A.T., postage and packing.

The ZAPPLE



The latest mind-blowing (well, eardrum-blowing actually) board - the ZAPPLE for your Apple.

An "intelligent" sound effects/music board.

- On board Eprom means easy to use from Basic
- On board amplifier for volume control (also control of Bleep!)
- Uses simple "sound tables" (Compiler software supplied FREE)
- Once sound is set up no processor time to maintain - Doesn't slow down graphics
- Sockets for 3 AY-3-8910s (1 supplied, extra at £8 each)

Send cheque for £49.50 (including post etc.) or £44.50 to first 20 BASUG members (quote membership No.) or send S.A.E. for further details. State disk or cassette.

MEEKROSE

Meekrose Ltd
85 Waterdell
Leighton Buzzard
Bedfordshire
LU7 8PL

Telephone: Leighton Buzzard
(0525) 370621

MNEMONIC PUZZLER

by D.R. Daniells

Currently I am trying to write a machine code Editor/Assembler for the Apple, and in the process I came across an intriguing problem...
In order to save space in the source file the editor 'crunches' each input line by removing unnecessary spaces and replacing the three character mnemonic by a token, in much the same way as each line of BASIC is compressed before being stored in memory.
Now in order to tokenise the mnemonics, it is convenient to treat them as reserved words which must be stored in memory for comparison purposes. As standard there are 56 such mnemonics (listed in the Apple Reference Manual), to which I have added four "pseudo-ops":- ORG, EQU, BYT, and ASC, making a total of 60, 3-letter reserved words.
I have chosen to store this information as three tables, each having 60 entries, where the tables consist of a list of all the first letters of each mnemonic, all the second letters, and all the third respectively.
Now we get to the puzzle. Although each table is 60 characters long, the combination of three such tables can be made to fit into less than 180 bytes of memory.

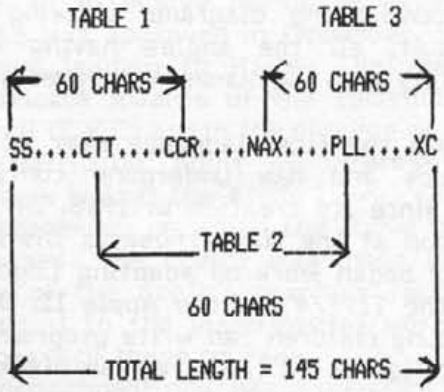
Consider the following list:-

0 STA	15 CLI	30 DEX	45 LDA
1 STX	16 ADC	31 INX	46 SEC
2 CLD	17 CPY	32 INY	47 ORG
3 BNE	18 RTS	33 JMP	48 INC
4 BPL	19 ROR	34 PHP	49 TAX
5 TSX	20 DEY	35 CMP	50 TXS
6 JSR	21 SBC	36 TYA	51 LDX
7 BEQ	22 BCS	37 TAY	52 DEC
8 CLV	23 BCC	38 LSR	53 PLA
9 PLP	24 BRK	39 NOP	54 TXA
10 ASL	25 BYT	40 PHA	55 ORA
11 ROL	26 ASC	41 SEI	56 EQU
12 BIT	27 AND	42 SED	57 BVC
13 RTI	28 BMI	43 EOR	58 CPX
14 STY	29 BVS	44 LDY	59 CLC

The numbers to the left indicate the order in which the words are stored. The three tables then look as follows:

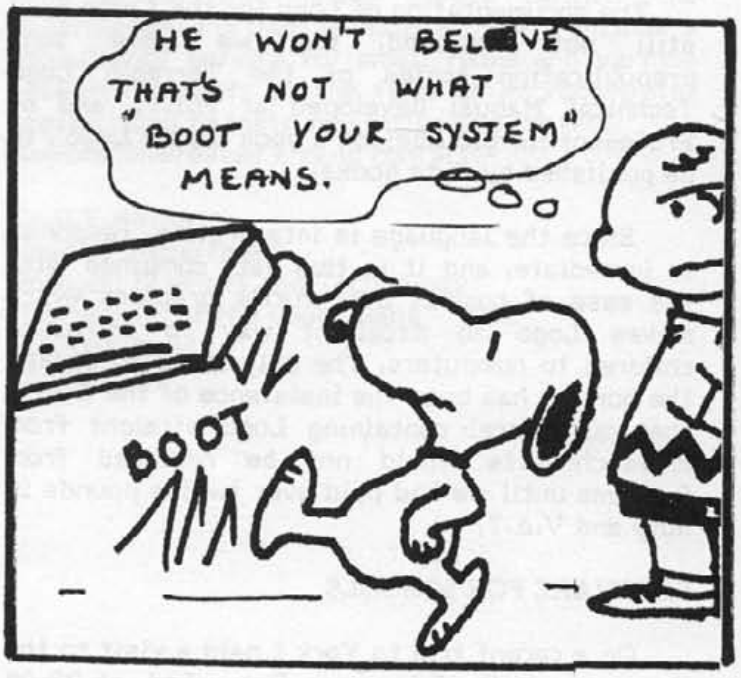
Table 1 - SSC.....BCC
Table 2 - TTl.....VPL
Table 3 - AXD.....CXC

Although the order in which the mnemonics have been written may seem strange, it enables the 180 characters to be stored in 145 bytes of memory.
Inspection will reveal that the last 24 characters of Table 1 (36-59) and the first 24 characters of Table 2 (0 to 23) are identical. At the same time, the last 11 characters of Table 2 (49-59) and the first 11 characters of Table 3 (0 to 10) are identical. Consequently the three tables can be overlapped in memory, saving 35 bytes, but without losing any information.



This is the way in which I have stored my data at present, but it is inevitable that a better solution exists - the question is then "Who can re-order the mnemonic list to make the greatest saving on overall length?"

I hope you find this entertaining. It certainly kept me engrossed for a few days.



EDUCATION COLUMN

Edited by Norah Arnold

TERRAPIN LOGO FOR THE APPLE II

During the last few weeks I have had the great pleasure of introducing some intelligent eight to nine year olds to the Terrapin Logo language. I have been inundated with turtle graphics programs, worked out in spare moments by little groups of children and neatly written out with accompanying diagrams showing the intended result, all the angles having been marked carefully with the number of degrees.

Logo was the first language ever to use turtle graphics and has undergone continual development since its creation in 1968. In 1979 the Logo Group at the Massachusetts Institute of Technology began work on adapting Logo for two micros, the TI99/4 and the Apple II. Using Logo even young children can write programs to draw complex figures on the screen in black and white or hi-res colour graphics.

The Terrapin Logo language is a procedure orientated language. Each procedure can take any number of arguments of any type, and can output a result. Procedures may be recursive and use local, global and free variables. There is a fully integrated screen editor and more than one hundred and twenty primitives including thirty graphics commands. Floating point arithmetic may be used and an assembly language interface includes an assembler written in Logo.

The documentation of Logo for the Apple II is still not finalised, but we were sent prepublication copies of the Terrapin Logo Technical Manual developed at M.I.T., and of Professor Harold Abelson's book 'Apple Logo', to be published by Byte books.

Since the language is interpretive, response is immediate, and it is this fact combined with the ease of control over turtle graphics which makes Logo an excellent way to introduce children to computers. The only cloud to darken the horizon has been the insistence of the G.P.O. that our parcel containing Logo straight from Massachusetts would not be released from Customs until we had paid over twelve pounds in duty and V.A.T.

SOFTWARE FOR SCHOOLS

On a recent trip to York I paid a visit to the Resources Unit of Longman Group Ltd. at 33-35 Tanner Row. Under the name of Longman Micro Software, they are marketing programs on biology, chemistry, economics, geography, mathematics and physics. Mr. Duncan Beal, the manager of the unit stressed that anyone writing educational software who would like their programs to be marketed by an established company, is welcome to submit a copy of their product to him.

Longman have also taken over from Edward Arnold the job of publishing the Schools Council 'Computers in the Curriculum' software. The Schools Council Economics package, which includes programs on the elasticity of demand, price fluctuations, price stabilisation, theory of the firm, the multiplier, fiscal policy, creation of credit, monetary policy and gains from trade, seems very good value to me at approximately twenty-two pounds, coming as it does with extensive documentation for both pupils and teachers.

This is in great contrast to some other software which has attracted my attention recently. Science Research Associates are offering several packages of Apple II computer software for primary schools. Traditional S.R.A. material tends to be well thought-out and produced but rather on the expensive side, and their computer software appears to be similar. A set of six S.R.A arithmetic games are priced at approximately one hundred pounds, while a program called Fact Track which provides timed practice in the four rules of number comes at sixty-five pounds. Other programs giving practice in arithmetic skills range from over one hundred to just above three hundred pounds. Having paid these amounts for the software, a school finds that the material is available under a licence agreement rather than through sale. Customers obtain the right to possession and use.

If any teachers have experience of using these programs I would be most interested in their comments. Most primary schools are short of money at present and the thought of spending over three hundred pounds for a maths tutor program which can be used by only one child at a time sounds like heresy to any headteacher.

S.R.A. are also offering Computer Discovery, a complete course in computer literacy for the secondary school and available on Apple II. This aims to introduce pupils to programming concepts and to the social issues related to computers.

Computer Discovery covers much the same ground as the Pedagog Programs (Aids to 'O' Level) from Pedagog Computer Services, but there is one vital difference. The Pedagog 'O' Level Aids consist of screen after screen of text which the pupils read, and hopefully, absorb. It

is hard to see the point of putting this material on the computer when it is not interactive and consists almost wholly of reading from the text screen. The Computer Discovery program from S.R.A., however, handles things in a far more lively and interactive manner. For example, in the section on analysis and programming pupils use a 'robot' whose movements they can 'program' themselves. They are also given an opportunity to create simple algorithms and flowcharts, and are able to practice using binary numbers.

Computer Discovery from S.R.A. is ninety-nine pounds and twenty student workbooks are included in the price. Although the Pedagog 'Aids' are cheaper at sixty pounds, in my opinion Computer Discovery is probably a far better buy.

Computer Games Review

COMPUTER GAMES REVIEW is a new bi-monthly magazine entirely devoted to evaluating, rating and reviewing new and established computer games software.

Most of us use our computers to play games - at least some of the time. But the range of games on the market is staggering; and as software isn't cheap its important to have a reliable idea of which games are good, which are merely indifferent, and which shouldn't even be on the market!

That's of course where COMPUTER GAMES REVIEW comes in: with detailed and informed analysis of games of all kinds. Whether your interests include Adventure, Arcade, Simulation or Educational games - CGR is for you.

The first issue of COMPUTER GAMES REVIEW is due out in June. CGR is available only on subscription, price £10 per year (6 issues) inc. p&p, from Computer Publications Ltd., 10 Star Lane, St.Mary Cray, Kent BR5 3LJ.

SPECIAL ADVANCE OFFER! *10% reduction to any BASUG member posting a subscription order by 10 May 1982! Send only £9 for one year.*

ANOTHER FIRST FOR THE APPLE

BASUG members, especially those concerned with education will be interested to hear of a new degree course in Microcomputing Applications. The Department of Physical Sciences and Technology at the Polytechnic of the South Bank is offering a BSc Honours degree in Physical Sciences with Computing starting in September, 1982. Degree courses in Polytechnics must pass a detailed scrutiny from a specialist panel of the Council of National Academic Awards (CNAA) and after an extensive period of development this course was approved in December, 1981. Related courses leading to Higher Diploma and Higher Certificate Awards of the Technician Education Council (T.E.C) are in the process of validation.

The particular point of interest for BASUG members is that the Apple II was selected as the microcomputer on which the above courses would be based. The choice was made partly on the basis of the Apple's graphics facility which is essential to the instrumental work which is at the heart of these courses.

The department now has a substantial number of Apples and many members of staff are involved in the hardware and software development necessary for such courses. The material in Hard Core has proved very useful in this connection and we look forward to contributing some of our own ideas which others may find helpful.

These courses are an exciting venture as they are the first of their kind in Britain and represent an important development in the field of education for microcomputer applications in industry. They are already attracting a a great deal of interest and if any BASUG member would like further information please contact the undersigned. Further developments will include a consultancy service for small firms and various research projects in the field of microcomputer applications to instrumentation and control.

Anyone interested should contact:-

Dr.B.E.Weller
Head of Department
Physical Sciences and Technology
Polytechnic of the South Bank
Borough Road
London SE1

MORE ON APPLE WRITER and GO-BETWEEN

By Ian Trackman

I was very interested to read the articles in the last edition of Hard Core (Vol. 2 No. 1) about relocating Apple Writer after moving DOS onto the language card. I would like to add some further suggestions for the switching routines and then explain how the same procedures might be adapted for use with Go-Between.

Memory expansion

But to start with, there is alternative advantage to relocating DOS. Instead of using the released memory for copies of Teditor and Printer, it could simply be made available as an extension to the text-file processing area. At the cost of having to Bload Teditor and Printer as before, the free memory is increased by about 30%. It's simply a matter of re-setting Apple Writer's standard "end of memory" address.

In Teditor, change $95 to $BD at these addresses :-

$836, $83F, $848 and $11B1.

In Printer, change $95 to $BD at $84D and $85E. Change $9B to $BE at these addresses :-

$F74, $FAD, $FC3, $117E, $1198, $119B, $11B0, $11BE and $11C9.

No guarantees that I'm right - but it seemed to work for me. Don't forget to move DOS before you try it !

Switch Routine

Rather than using Applesoft's "STROUT" routine, I prefer to keep Apple Writer completely independent of any Language Rom and so I would use COUT ($FDED) to send messages to DOS.

Other readers may also have their own ideas for enhancing the switching routine. To avoid having different Apple Writer patches for each variation, it would be better to operate the switch between the modules of Apple Writer by way of an indirect jump. For that, we will need two double-byte vectors. The ampersand hook at $3F6/7 is available for one vector. There are two free bytes in the DOS Page 3 area at $3ED and $3EE (they are normally set to NOP NOP) which we can use as the other vector. Having two vectors does away with the need to test for which module is active, since each module will call its partner's vector.

The switching code itself can be streamlined as suggested in the listing below.

As to the actual patches, Teditor's patch now becomes an indirect jump command JMP ($3DE), i.e. :-

128C : 6C ED 03

I think that a better place to patch Printer is at $AB0, where it first responds to the "R" command, viz :-

AB0 : 6C F6 03

I would not patch out the Save Constants routine at $B54, as I assume that it would be desirable to save new constants. I think that the correct place to patch out the "load constants" routine is at $AFB with a simple RTS (60).

I also suggest moving the maximum possible size of each Apple Writer module. This should therefore cover all other patches (e.g. lower-case, Chain Mail) which are "tagged on" after the end of the main code.

Go-Between

Patching Go-Between is slightly more complicated because it contains internal routines to enable the program disk and text-file disk to operate automatically on different drives. I have not done any programming using relocated DOS and so I cannot guarantee that the patches which follow will work infallibly. They are unofficial and not fully tested ! Please let me know of any problems or - better still - the solutions !

Since I wrote the original Centronics version of Go-Between, I have made a few minor adjustments to the code, which means that care must be taken to patch the correct addresses. One of the changes put right a wrong calculation in the "continuous paper mode" routine. I have also recently discovered that the firmware on the Aristocard and Grappler interface cards does not handle "Control I 255 N" properly for unlimited line length. Version 1.2 of Centronics Go-Between deals with the first bug and Version 1.3 also covers the second problem. The version number is shown in a REM statement in the HELLO program. I would be happy to up-grade any earlier versions on receipt of the original disk and 50p to cover the return postage. There is only one version of the Standard Go-Between.

The patching addresses that I will now give will be for Centronics Version 1.3. The addresses for the Standard Version are shown in brackets. Because of the change mentioned above from "255" to "0", the changes to Centronics Version

1.2 will be offset by two bytes. Owners of Version 1.2 should disassemble the code to ensure that they are making the correct changes. As an aside - I think that whenever a patch to any program is suggested, the bytes in the original code should be indicated. This will help to confirm that you are patching the bytes intended by the author.

After initialisation, the program begins by loading the constants and the correct character set. This should now be done by the DOS Patcher and is simply a matter of giving the correct commands in the "Text" section. Note that loading addresses are needed in the Standard Version.

The loading code in Go-Between is at $816 ($816), starting with a LDA, LDX. This can be skipped over with a jump to the next routine at $827 ($81D) (i.e. enter 4C 27 (1D) 08 at $816). Bear in mind the warning given in the original article - the constants conflict with Basic's use of zero page. If you leave Apple Writer and then return via the ampersand, you must reload the Constants. In the Centronics Versions, you can do this either with a Control D command from Teditor or by entering the Change Constants program (you needn't make any changes, of course). In the Standard Version, you must "BLOAD CONSTANTS, A$50" and "BLOAD USER CONSTANTS, A$1780" (these are the commands to put into the DOS Patcher program.

At $831 and $83A ($836 and $83F), I pick up the current slot and drive from $B7E9 and B7EA. These should now be taken from $BFE4 and $BFE5.

At $85F ($864), there is a JSR to $A851 to re-hook DOS. This should now be vectored through $3EA.

At $86B, $870, $916, $91B, $9FF and $A04 ($870, $875, $91E, $923, n/a, n/a), I use $AA6A and $AA68 to reinstate slot and drive references. These should be changed to $EA6A and $EA68.

To prevent DOS from intercepting any output, I disabled it at $A6A ($A78) with a call to $9ED1. Cornelius Bongers (the author of DOS MOVER) does not appear to have provided us with a simple call to do this and so I suggest that the easiest method is to patch out my JSR $9ED1 with three NOPs (EA EA EA) and hope that there are no Control Ds in your text-file !

The error trap vector is set at $E52 and $E55 ($E39 and $E40) and, as with the original PRINTER, should now refer to $DD5A and $DD5B instead of $9D5A and $9D5B.

The routine to call Teditor starts at $900 ($908) (with a LDA #0 STA $200) and that is the place for JMP ($3F6) i.e. 6C F6 03.

In the Centronics Version, "Change Constants" also has to be patched. At $813, remove JSR $9ED1 with NOPs (as above). At $B84, re-hook DOS with a JSR $3EA (which is strictly unnecessary if you don't unhook it). Return to Go-Between at $BA2 with a JMP ($3F6) - 6C F6 03.

Finally, if you want memory expansion (as explained at the beginning of this article) with Go-Between, you shouldn't need to make any changes to it.

```
SOURCE FILE: A-W RAMCARD PATCHER
0000:              1 ********************************
0000:              2 *
0000:              3 * APPLE WRITER SWITCHER
0000:              4 * FOR LANGUAGE CARD DOS
0000:              5 *
0000:              6 * By Ian Trackman
0000:              7 *
0000:              8 * After ideas of Michael Clark,
0000:              9 * John Rodger & John Sharp
0000:             10 *
0000:             11 ********************************
0000:             12 *
003C:             13 FROMBGNL EQU $3C
003D:             14 FROMBGNH EQU $3D
003E:             15 FROMENDL EQU $3E
003F:             16 FROMENDH EQU $3F
0042:             17 DESTL    EQU  $42
0043:             18 DESTH    EQU  $43
03ED:             19 JMPVECTL EQU $3ED
03EE:             20 JMPVECTH EQU $3EE
03F6:             21 ANDVECTL EQU $3F6
03F7:             22 ANDVECTH EQU $3F7
```

```
0803:                 23 PROGRAM EQU  $803
9D03:                 24 TEDSAVE EQU  $9D03
ADFF:                 25 TEDEND  EQU  $ADFF
AE03:                 26 PRNTSAVE EQU $AE03
BEFF:                 27 PRNTEND EQU  $BEFF
FDED:                 28 COUT    EQU  $FDED
FE2C:                 29 MOVE    EQU  $FE2C
0000:                 30 ;
------ NEXT OBJECT FILE NAME IS A-W RAMCARD PATCHER.OBJ0
9C00:                 31         ORG  $9C00
9C00:                 32 ;
9C00:                 33 ; Bload all files
9C00:                 34 ;
9C00:A2 00            35         LDX  £0
9C02:BD 47 9C         36 GO.LOAD LDA  TEXT,X
9C05:F0 06            37         BEQ  LOADDONE
9C07:20 ED FD         38         JSR  COUT
9C0A:E8               39         INX
9C0B:D0 F5            40         BNE  GO.LOAD    ;Always
9C0D:                 41 ;
9C0D:A9 9C            42 LOADDONE LDA £<GET.TED ;Lo-byte
9C0F:8D F6 03         43         STA  ANDVECTL
9C12:A9 21            44         LDA  £>GET.TED ;Hi-byte
9C14:8D F7 03         45         STA  ANDVECTH
9C17:A9 9C            46         LDA  £<GETPRNTR
9C19:8D ED 03         47         STA  JMPVECTL
9C1C:A9 27            48         LDA  £>GETPRNTR
9C1E:8D EE 03         49         STA  JMPVECTH
9C21:                 50 ;
9C21:                 51 ; Set up high bytes
9C21:                 52 ;
9C21:A9 03            53 GET.TED LDA  £>TEDSAVE
9C23:A2 FF            54         LDX  £>TEDEND
9C25:D0 04            55         BNE  DO.MOVE    ;Always
9C27:                 56 ;
9C27:A9 03            57 GETPRNTR LDA £>PRNTSAVE
9C29:A2 FF            58         LDX  £>PRNTEND
9C2B:                 59 ;
9C2B:85 3D            60 DO.MOVE STA  FROMBGNH
9C2D:86 3F            61         STX  FROMENDH
9C2F:A9 9D            62         LDA  £<TEDSAVE ;Lo-byte is same for Printer
9C31:85 3C            63         STA  FROMBGNL
9C33:A9 AD            64         LDA  £<TEDEND  ;Lo-byte is same for Printer
9C35:85 3E            65         STA  FROMENDL
9C37:A9 08            66         LDA  £<PROGRAM
9C39:85 42            67         STA  DESTL
9C3B:A9 03            68         LDA  £>PROGRAM
9C3D:85 43            69         STA  DESTH
9C3F:A0 00            70         LDY  £0         ;As required by Move
9C41:20 2C FE         71         JSR  MOVE
9C44:4C 03 08         72         JMP  PROGRAM
9C47:                 73 ;
9C47:8D 84            74 TEXT    DFB  $8D,$84
9C49:C2 CC CF         75         ASC  "BLOAD    TEDITOR,A$9D03"
9C4C:C1 C4 A0
9C4F:D4 C5 C4
9C52:C9 D4 CF
9C55:D2 AC C1
9C58:A4 B9 C4
9C5B:B0 B3
9C5D:8D 84            76         DFB  $8D,$84
9C5F:C2 CC CF         77         ASC  "BLOAD    PRINTER,A$AE03"
```

```
9C62:C1 C4 A0
9C65:D0 D2 C9
9C68:CE D4 C5
9C6B:D2 AC C1
9C6E:A4 C1 C5
9C71:B0 B3
9C73:8D 84        78          DFB  $8D,$84
9C75:C2 CC CF     79          ASC  "BLOAD    PRINT CONSTANTS"
9C78:C1 C4 A0
9C7B:D0 D2 C9
9C7E:CE D4 A0
9C81:C3 CF CE
9C84:D3 D4 C1
9C87:CE D4 D3
9C8A:8D 00        80          DFB  $8D,0

SOURCE FILE: G-B RAMCARD PATCHER
0000:              1 ***********************************
0000:              2 *
0000:              3 * APPLE WRITER SWITCHER
0000:              4 * FOR LANGUAGE CARD DOS
0000:              5 * GO-BETWEEN 737 VERSION
0000:              6 *
0000:              7 * By Ian Trackman
0000:              8 *
0000:              9 * After ideas of Michael Clark,
0000:             10 * John Rodger & John Sharp
0000:             11 *
0000:             12 ***********************************
0000:             13 *
003C:             14 FROMBGNL EQU $3C
003D:             15 FROMBGNH EQU $3D
003E:             16 FROMENDL EQU $3E
003F:             17 FROMENDH EQU $3F
0042:             18 DESTL    EQU  $42
0043:             19 DESTH    EQU  $43
03ED:             20 JMPVECTL EQU $3ED
03EE:             21 JMPVECTH EQU $3EE
03F6:             22 ANDVECTL EQU $3F6
03F7:             23 ANDVECTH EQU $3F7
0803:             24 PROGRAM EQU  $803
9D03:             25 TEDSAVE EQU  $9D03
ADFF:             26 TEDEND  EQU  $ADFF
AE03:             27 PRNTSAVE EQU $AE03
BEFF:             28 PRNTEND EQU  $BEFF
FDED:             29 COUT    EQU  $FDED
FE2C:             30 MOVE    EQU  $FE2C
0000:             31 ;
----- NEXT OBJECT FILE NAME IS G-B RAMCARD PATCHER.OBJ0
9C00:             32          ORG  $9C00
9C00:             33 ;
9C00:             34 ; Bload all files
9C00:             35 ;
9C00:A2 00        36          LDX  £0
9C02:BD 47 9C     37 GO.LOAD  LDA  TEXT,X
9C05:F0 06        38          BEQ  LOADDONE
9C07:20 ED FD     39          JSR  COUT
9C0A:E8           40          INX
9C0B:D0 F5        41          BNE  GO.LOAD    ;Always
9C0D:             42 ;
9C0D:A9 9C        43 LOADDONE LDA £<GET.TED ;Lo-byte
9C0F:8D F6 03     44          STA  ANDVECTL
9C12:A9 21        45          LDA  £>GET.TED ;Hi-byte
9C14:8D F7 03     46          STA  ANDVECTH
```

```
9C17:A9 9C       47          LDA  £<GETPRNTR
9C19:8D ED 03    48          STA  JMPVECTL
9C1C:A9 27       49          LDA  £>GETPRNTR
9C1E:8D EE 03    50          STA  JMPVECTH
9C21:            51 ;
9C21:            52 ; Set up high bytes
9C21:            53 ;
9C21:A9 03       54 GET.TED  LDA  £>TEDSAVE
9C23:A2 FF       55          LDX  £>TEDEND
9C25:D0 04       56          BNE  DO.MOVE    ;Always
9C27:            57 ;
9C27:A9 03       58 GETPRNTR LDA  £>PRNTSAVE
9C29:A2 FF       59          LDX  £>PRNTEND
9C2B:            60 ;
9C2B:85 3D       61 DO.MOVE  STA  FROMBGNH
9C2D:86 3F       62          STX  FROMENDH
9C2F:A9 9D       63          LDA  £<TEDSAVE ;Lo-byte is same for Printer
9C31:85 3C       64          STA  FROMBGNL
9C33:A9 AD       65          LDA  £<TEDEND  ;Lo-byte is same for Printer
9C35:85 3E       66          STA  FROMENDL
9C37:A9 08       67          LDA  £<PROGRAM
9C39:85 42       68          STA  DESTL
9C3B:A9 03       69          LDA  £>PROGRAM
9C3D:85 43       70          STA  DESTH
9C3F:A0 00       71          LDY  £0         ;As required by Move
9C41:20 2C FE    72          JSR  MOVE
9C44:4C 03 08    73          JMP  PROGRAM
9C47:            74 ;
9C47:8D 84       75 TEXT     DFB  $8D,$84
9C49:C2 CC CF    76          ASC  "BLOAD     TEDITOR,A$9D03"
9C4C:C1 C4 A0
9C4F:D4 C5 C4
9C52:C9 D4 CF
9C55:D2 AC C1
9C58:A4 B9 C4
9C5B:B0 B3
9C5D:8D 84       77          DFB  $8D,$84
9C5F:C2 CC CF    78          ASC  "BLOAD     PRINTER,A$AE03"
9C62:C1 C4 A0
9C65:D0 D2 C9
9C68:CE D4 C5
9C6B:D2 AC C1
9C6E:A4 C1 C5
9C71:B0 B3
9C73:8D 84       79          DFB  $8D,$84
9C75:C2 CC CF    80          ASC  "BLOAD     CHARACTERS - COUNTRY 2,A$BE80
9C78:C1 C4 A0
9C7B:C3 C8 C1
9C7E:D2 C1 C3
9C81:D4 C5 D2
9C84:D3 A0 AD
9C87:A0 C3 CF
9C8A:D5 CE D4
9C8D:D2 D9 A0
9C90:B2 AC C1
9C93:A4 C2 C5
9C96:B8 B0
9C98:8D 84       81          DFB  $8D,$84
9C9A:C2 CC CF    82          ASC  "BLOAD     CONSTANTS"
9C9D:C1 C4 A0
9CA0:C3 CF CE
9CA3:D3 D4 C1
9CA6:CE D4 D3
9CA9:8D 00       83          DFB  $8D,0
```

MICROSOURCE

1 Branch Road, Park Street, St. Albans.
Tel: Park Street (0727) 72917

are pleased to announce their appointment as distributors of

BLUE CHIP SOFTWARE

SUPER EDITOR £22.50

Super Editor is an essential tool for anyone writing programs in Applesoft or Palsoft. Using a simple command format, this machine-code routine will speedily list out every program line which contains a Basic command, function, variable or string designated by the user, highlighting it in inverse characters. Super Editor also enables the user optionally to substitute any other command, function, variable or string for the item found, whatever their respective sizes.

* Remains in memory whilst programs are loaded and saved
* Reduce long descriptive variable names in final versions of programs
* List every occurence of a designated variable for debugging
* Check all GOTO, GOSUB and THEN references before deleting a line
* Change integer arrays to real arrays and vice versa
* Convert Integer programs to Applesoft and vice versa

SUPER TRACE £22.50

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

THE PACKING SUITE £22.50

The Packing Suite is an essential series of high-speed machine-code utility programs for anyone working with Applesoft / Palsoft programs.

UNPACKER separates lines consisting of multiple statements into new consecutive line consisting of single statements, to assist with editing and debugging.

REM STRIPPER removes REM statements and, if required, lines consisting of a single colon (which might be used for spacing).

PACKER puts statements together to form maximum-size multi-statement program lines for faster operation and memory saving.

All the routines observe Applesoft's rules of syntax (e.g. multiple statements after an IF are protected) and all line-number cross-references to deleted lines are properly adjusted to a valid line-number.

SYMDIS £22.50

Symdis will take a block of 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. Optionally, the source file can be produced with the hexadecimal or ASCII values of the object code displayed. The source file, in standard ASCII format, is saved directly to disk so that the only additional memory area required is for the operational symbol table. The source file can be reloaded into any Assembler/Text Editor which handles standard ASCII format source-files. Alternatively, by using the conversion program provided, the source file can be presented to the 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.

SPECIAL OFFER TO BASUG MEMBERS - VALID TO 31st MAY only!!!

Any two - £40 ****** All Four - £ 75**

Dealer enquiries welcome.

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 Accutrak, 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.

BASUG BULK BUYING – 50% off PRICE FOR 10s

£17:50 incl FREE LIBRARY CASE !

(ADD 50p POSTAGE / PACKING)

BRITISH APPLE SYSTEMS USER GROUP

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

TWO PROGRAMS CO-RESIDENT IN MEMORY.

By Les Budgen.

Have you ever wished that you could have two Applesoft or Palsoft (FP) programs in your micro and be able to access either easily.If you have your wish is granted.With the aid of a short machine code program and the Apple ampersand function you can have two readily accessible programs.

FP basic is loaded into memory begining at a location determined by the value of two zero page memory locations (called pointers from the fact that they direct basic to some other location).These pointers are located at $67and $68.FP basic normaly sets these to $01 and $08 respectively.This sets the start of a program loaded or entered from the keyboard at $801.If these pointers are altered before loading a program then the program will be loaded at a new location.For example if they are set to $01 and $60 the program will be loaded at $6001 ie.just above hires graphics page 2.The next important pointers point to the end of the FP program and are located at $AF and £B0.These pointers tell basic where the end of the program is for the purpose of ...

1)Saving the program correctly.

2)Where to start if a new line is added to the program.

If you look on pages 140 and 141 of the Applesoft Manual and in Hardcore Volume 1 Number 1 you will find a list of zero page usage,these pointers are explained there.

O.K.- now for the two program bit...

Enter the following machine code program from the monitor.

```
0300-   A9 4C      LDA   #$4C
0302-   A2 18      LDX   #$18
0304-   A0 03      LDY   #$03
0306-   8D F5 03   STA   $03F5
0309-   8E F6 03   STX   $03F6
030C-   8C F7 03   STY   $03F7
030F-   60         RTS
0310-   01 60      ORA   ($60,X)
0312-   01 60      ORA   ($60,X)
0314-   04         ???
0315-   60         RTS
0316-   04         ???
0317-   60         RTS
0318-   A9 00      LDA   #$00
031A-   8D 00 60   STA   $6000
031D-   A5 67      LDA   $67
031F-   A6 68      LDX   $68
0321-   8D 10 03   STA   $0310
0324-   8E 11 03   STX   $0311
0327-   AD 12 03   LDA   $0312
032A-   AE 13 03   LDX   $0313
032D-   85 67      STA   $67
032F-   86 68      STX   $68
0331-   AD 10 03   LDA   $0310
0334-   AE 11 03   LDX   $0311
0337-   8D 12 03   STA   $0312
033A-   8E 13 03   STX   $0313
033D-   A5 AF      LDA   $AF
033F-   A6 B0      LDX   $B0
0341-   8D 14 03   STA   $0314
0344-   8E 15 03   STX   $0315
0347-   AD 16 03   LDA   $0316
034A-   AE 17 03   LDX   $0317
034D-   85 AF      STA   $AF
034F-   86 B0      STX   $B0
0351-   AD 14 03   LDA   $0314
0354-   AE 15 03   LDX   $0315
0357-   8D 16 03   STA   $0316
035A-   8E 17 03   STX   $0317
035D-   60         RTS
```

Having entered the program check it against the listing above. Then save the program- BSAVE(name),A$300,L$5F

To use the program...

1.BRUN(name) before loading your programs
2.Load or enter your first program.
3.Press & and <RETURN>.
4.Type NEW <RETURN>.
5.Load or enter your second program.

You may now switch from one to the other when you wish just with &<RETURN>.

Explanation.

0300 to 030F sets the & pointer to 0318. 0318 is the begining of the change-over routine.

The numbers stored between 0310 and 0317 are the new values for the pointers.These will be exchanged with the actual pointers each time you press & and RETURN.

If you wish to have your second program at another location alter the value $60 at 0311,0313,0315,0317 and 031C to your own value.E.G.if you alter them to $40 your second program will be located at $4001.

It is possible for the two programs to share the same variables. Having set them with one program you can switch to the other one with & RETURN.Do not run the second program as this will set the variables to zero,instead you should type GOTO (first line number).

If you are using hires graphics you may need to set LOMEM to keep the variables out of the graphics screen memory.If you do, remember to set it above the second program.You may find the end of a program with...

PRINT PEEK(175)+256*PEEK(176)

Pippin's Page ~~~~

Edited for younger readers by Vernon Quaintance

Welcome to another Pippin's Page. I would welcome your letters saying how useful you find this page, and what changes, if any, you would like to see here. Your programs would also be welcomed. You can write to Pippin's Page c/o P.O Box 174, Watford, WD2 6NF.

In the last Hardcore we saw how to use the GOTO command to cause a program to run indefinitely. At some time, however, we will want to stop the program from running. One way is to switch off - but this loses the program. Another would be to use the RESET key; this will be OK if you have an Apple][Plus, but not with an Apple][.

There is another way to stop a program without risk of it being lost. Enter a program with a GOTO placed at the end to cause it to keep on running. Now RUN the program.

Next press the key marked CTRL and hold it down whilst pressing the letter 'c'. Has the program stopped? Has the] symbol and the flashing cursor appeared? If not, you may have stopped at an INPUT statement. press the RETURN key. The] and cursor should now be on the screen.

Notice that the program will not stop on CTRL-C alone if it is waiting for an INPUT it will then also need a RETURN, at any other point in the program a CTRL-C alone will stop it.

Supposing now that we wish to control the number of times that a program repeats itself. We will need some means of counting the number of repeats and also of testing to see if this is the required number.

Type in the following program: (Don't forget the NEW first)

```
10 REM LINES GENERATOR
20 TEXT : HOME
30 INPUT "HOW MANY LINES ARE TO BE
WRITTEN? "; N
40 HOME
50 PRINT "I MUST PAY ATTENTION IN CLASS"
60 LET COUNT = COUNT + 1
70 IF COUNT < N THEN GOTO 50
80 END
```

Now RUN it, choose a small number of lines to start with (say 5). Your 'lines' should have been printed on an otherwise blank screen. Did you get the right number of them?

The two new ideas in this program occur in lines 60 and 70. line 60 should be read as "let count become count plus one". that is, take the present value of the variable COUNT and add one to it, then store this new value in the variable COUNT.

Line 70 performs a test to find out if the value of COUNT is less than the value of N. all the time that COUNT is less than N, the program loops back to line 50. Let us assume that you asked for ten lines, therefore N = 10. As soon as COUNT reaches 10 (which is after 10 lines have been written) then the test is no longer true (COUNT is not less than N) and the program does not loop back but carries on to line 80 where it ends.

You should note that when you RUN a program, the value of all numeric variables is set to 0. COUNT did not become 1 until after we had written the first 'line'.

Let us now change the program slightly. Re-type lines 70 and 80 to read:-

```
70 IF COUNT = N THEN END
80 GOTO 50
```

Now RUN this program. It should do exactly the same as the previous one. Depending upon exactly what we want to do, we can test for greater than (>), less than (<), equal to (=), less than or equal to (<=), greater than or equal to (>=), or not equal to (<>). Try writing some simple programs of your own using each of these comparisons, see how you can change the program to do the same job with a different comparison.

I expect most of you have Applesoft or Palsoft BASIC. In these articles I will use the name Applesoft to mean both Applesoft itself and also Palsoft. I will only mention Palsoft specifically if I know it to be different.

If you have Applesoft then try this program:-

```
10 TEXT : HOME
20 PRINT N
30 N = N + 1
40 IF N <> 11 THEN PRINT ", " : GOTO 20
50 END
```

When you RUN this program you should find that it prints the numbers from 0 to 10 in a line, separated by a comma and a space. There will be no comma after the 10 though. You will see that the program tests to find out if N is not equal to 11. Whilst this is so it will do two things (a) it will print a comma and a space, (b) it will loop back to line 20. In fact, every command which follows the THEN in line 40 will be obeyed if, and only if, the test is true. As soon as the test is not true then none of the remaining commands in that line will be obeyed.

If you have Integer BASIC and try this program you will see a distinct difference. The program will loop indeffinitely and continue writing numbers to the screen until you stop it with CTRL-C. There will be a comma and a space after all the numbers except the 10 (which is when N = 11). With Integer BASIC only the command immediately following the THEN depends on the test. All other commands in the line will be obeyed anyway.

In some versions of BASIC (as used on other computers or with the Softcard) you can write a line like:

```
90 IF N = 15 THEN PRINT : PRINT "RUN FINISHED" ELSE GOTO 20
```

Note the new command ELSE used here. If the test is true then all that is between the THEN and the ELSE will be obeyed. If the test is not true then only what follows the ELSE will be obeyed.

Next time we will learn how to control looping in another way, which is the one we will then normally want to use.

THE HERTS LOCAL GROUP

Meetings of this group are held on the first Tuesday of the month at The Old School, 1 Branch Road, Park Street, St. Albans. Forthcoming meetings have been planned as follows:-

Tuesday May 4th - VisiCalc
business applications, etc.

Tuesday June 1st - Terrapin Logo
powerful turtle-graphics for the Apple

Provisional topics for July/Oct:-
Tues. July 6th - Amateur Radio
Tues. August 3rd - Games evening.
Tues. Sept. 7th - Music on the Apple
Tues. Oct. 5th - Z80 Board/CPM

All members are welcome, whether beginners searching for encouragement and help or those with more experience. If you have any suggestions regarding the topics you would like to see covered in future meetings, contact Bob Raikes ([illegible]) or have a word with Bob or myself when you come along to a meeting.

Norah Arnold.

LEICESTER APPLE USER GROUP FOR HELP AND SUPPORT
(L.A.U.G.H.S.)
(AFFILIATED TO B.A.S.U.G.)

This is a thriving group of lively Apple users which started nine months ago. With our increasing membership we have now outgrown our present premises at the "Leicester Computer Centre", and from 7th April will be meeting at

The Winstanley Arms,
The Glade,
Narborough Road,
Leicester,
(Near M1 Junction 21)

Our meetings vary in content and include interesting discussions and speakers, demonstrations of software, and 'Help Forums'. Member facilities include embryo software and text libraries, and discounts on Computer products.

If you are a BASUG member or an Apple user living in the East Midlands area you would be welcome at our meetings which start at 7.30 pm on the first Wednesday of each month.

If you require further information this can be obtained from the secretary

Hazel Bown,
[illegible]

or from the chairman

Mike Preston
[illegible]

Hazel Bown

REPEAT....UNTIL

By Bob Raikes

In Hardcore 3 there was an interesting article by Ian Trackman about the uses of Apple's ability to return Boolean values for tests of equality (Hard Core 3, p12, "The Disappearing 'If' trick").

In the September issue of BYTE there is a short routine which demonstrates the use of this ability to simulate the "Repeat....Until" statement available on some machines.

The routine is:-

```
10 FOR D=0 TO 1
20 action
30 D=CONDITION
40 NEXT D
```

As an example:-

```
10 FOR D=0 TO 1
20 PRINT "X";
30 D=(PEEK(-16287)>127)
40 NEXT D
```

The first time the program performs the loop D=0 until line 30. At that point the condition is evaluated either as true or false. If it is false D remains at 0, but if it is true D becomes 1.

At line 40 D is incremented by the BASIC default value of 1. BASIC will continue with a FOR....NEXT loop until the index is greater than the value after the 'TO'.

If the condition was false then D becomes 1 at the line 40. This is not greater than the highest value (1) so the loop repeats. If, however, the condition is true, D becomes 2 at line 40. As this is greater than the highest loop value, control will pass to the line following 40.

The example program will keep printing asterisks until the button on paddle 0 is pushed.

This could, of course, be done with a routine such as:

```
10 PRINT"*";
20 IF PEEK(-16287)<128THEN 10
```

The author of the BYTE article claimed that the REPEAT....UNTIL method was quicker. I haven't, as yet, fully tested this.

BASUG TOOLKIT NO.1

MACHINE CODE PRINT ROUTINE

By Cliff Wootton

Have you ever wanted to put lots of text into a program written in assembly language, but been daunted by the writing of an indexed loop? Well I have used this routine for the last 3 years and it has been possibly the most used of all my subroutines. The comments on the source listing should be self evident. The program should be included with the other subroutines in your source code and should be used as follows:-

```
Print Statement  JSR PRINT
                 DCI "THIS IS YOUR TEXT"
Next line of
     code        LDA £$FF
(Need not be LDA £$FF)
```

The construct DCI "string" will assemble the ASCII values of each character into the object code. Bit 7 of each character except the last will be clear. The last character has its Bit 7 set to indicate the end of the string. It's just as easy to use for one character since that character is assumed to be the last character of the string.

```
Thus JSR PRINT
ASC "A" consumes 4 bytes of memory
While LDA £ $C1JSR COUT
consumes 5 bytes of memory.
```

The operation is quite simple and relies on the fact that when a JSR is performed, the return address minus 1 is stored on the stack. This can be pulled and used as a pointer to the string. When the string is completed the Y register holds its length. This is added to the pointer value which is then pushed back onto the stack. When an RTS is next encountered the program execution carries on with the first instruction following the string of text.

There are several advantages to be gained from this technique. One of the most useful being that it is well nigh impossible to disassemble the object code quickly and easily. This only makes it harder for the would be copyright notice remover. The other main advantage is that the text appears in the source code where required and not elsewhere. Be careful not to use this function with strings longer than 256 characters. It could be modified to handle more but would probably be of limited appeal were it so modified. It is also possible to modify it to use some encription algorithim to make it harder to find text messages within the object code.

```
:ASM
0000:                  1 ********************
0000:                  2 *                  *
0000:                  3 * MACRO PRINT RTN *
0000:                  4 *                  *
0000:                  5 ********************
0000:                  6 *
6000:                  7         ORG $6000      ; START IT HERE
6000:                  8         OBJ $6000      ; JUST FOR CONVENIENCE
6000:                  9 *
6000:                 10 ************************
6000:                 11 *                      *
6000:                 12 * PRINT EQUATES TABLE *
6000:                 13 *                      *
6000:                 14 ************************
6000:                 15 *
6000:                 16 PTRL        EQU $00    ; ZERO PAGE POINTER CAN BE ANYWHERE
6000:                 17 *
6000:                 18 COUT        EQU $FDED  ; MONITOR PRINT CHAR IN ACC
6000:                 19 *
6000:                 20 ****************************
6000:                 21 *                          *
6000:                 22 * MACRO PRINT SUBROUTINE *
6000:                 23 *                          *
6000:                 24 ****************************
6000:                 25 *
6000: 68              26 PRINT   PLA            ; PULL THE RETURN
6001: 85 00           27         STA PTRL       ; ADDRESS OFF THE STACK
6003: 68              28         PLA            ; AND STICK IT INTO
6004: 85 01           29         STA PTRL+1     ; THE ZERO PAGE POINTER BYTE PAIR
6006: A0 00           30         LDY #$00       ; SET THE Y REG OFFSET TO 0
6008: C8              31 LOOP    INY            ; AND POINT AT THE FIRST CHARRACTER
6009: B1 00           32         LDA (PTRL),Y   ; THEN GET THE NEXT CHAR
600B: 48              33         PHA            ; SAVE IT ON THE STK FOR LATER
600C: 09 80           34         ORA #$80       ; SET THE BIT SEVEN HIGH
600E: 20 ED FD        35         JSR COUT       ; AND PRINT THE CHAR
6011: 68              36         PLA            ; RETREIVE THE ORIGINAL VALUE
6012: 10 F4           37         BPL LOOP       ; AND LOOP IF BIT 7 WAS A 0
6014: 98              38         TYA            ; ELSE GET OFFSET
6015: 18              39         CLC            ; CLEAR ANY CARRY OVER
6016: 65 00           40         ADC PTRL       ; AND ADD THE OFFSET TO
6018: A8              41         TAY            ; THE BASE ADDRESS TO
6019: A5 01           42         LDA PTRL+1     ; GENERATE THE NEW RETURN
601B: 69 00           43         ADC #$00       ; ADDRESS FOR THE RTS
601D: 48              44         PHA            ; SAVE THE HIGH BYTE
601E: 98              45         TYA            ; THEN GET THE LOBYTE BACK
601F: 48              46         PHA            ; AND SAVE THAT
6020: 60              47         RTS            ; WHEN YOU RTS YOU GO BACK AFTER STRING
6021:                 48 *
6021:                 49 ************************
6021:                 50 *                      *
6021:                 51 * END OF MACRO PRINT *
6021:                 52 *                      *
6021:                 53 ************************
    0   ERRORS IN THIS ASSEMBLY
```

THE NICELISTER

(c) By T Tse

One day ALL Applesoft programs will be nicely listed!!

The Applesoft Manual, on page 48, specifically warns you that the LIST command adds spaces liberally. The Basic Interpreter LIST routine always adds two extra spaces for each Applesoft token disassembled - one immediately before listing the Basic command and one immediately after listing the Basic reserved word. This method produces a perfectly readable listing, is rather wasteful of spaces and is somewhat irritable to the eyes when most of your Basic lines tend to list on more than one line. The degree of irritation is dependent upon the type of printer/interface being employed.

The solution is The NiceLister routine presented here. Some of its obvious advantages are:

1. Simple to use

The NiceLoader program, listed further on, will BLOAD The NiceLister routine and set up the &-Ampersand jump vector for user interface. After that, NiceLister is invoked simply by typing &nnn, where nnn is a numeric argument in the range 7 to 255, to specify the column width of the listing required. (Note: the printer has to be initialised first via a PR£n if printer output is desired).

2. A really Nice Listing

All Basic line numbers are right-justified on the 5th listing column, while all Basic statements listed are left-justified on the 7th listing column.

3. More compact listings

The NiceLister is very mean on spaces, yet does not suffer any consequential loss in program readability. A single space is added before dislisting a token ONLY if that token will disassemble into more than one character. Another space is added after dislisting the token ONLY if the next Basic byte to be listed is not a token!! This logic works out quite satisfactorily.

4. Virtual printer/interface independence

NiceLister is believed to work on ANY printer with ANY interface card. To list on, say, 80 column printer width, all you need to do is to type in the sequence PR£n and &80. There is no longer the need for controlling commands such as CTRL-I 80n or POKE 1913,80.

Finally, an apology, folks! The whole NiceLister routine was typed in, debugged, tested and eventually patched using the old monitor and mini-assembler supplied with Integer Basic. So you will have to key it all in again!! To save the routine onto diskette, enter BSAVE NICELISTER, A$8000,L$260.

```
8000- 4C 60 81 00 00 00 00 00
8008- 00 00 00 00 00 00 00 00
8010- D0 D3 D6 DA DE E3 E6 E9
8018- ED EF F3 F6 F9 FD 01 05
8020- 09 0D 10 17 1C 20 25 29
8028- 2D 31 37 3D 42 49 4F 56
8030- 5B 61 64 68 6E 74 79 7F
8038- 85 8A 90 93 97 9A 9C A3
8040- A4 A9 AF B2 B6 B8 BC C0
8048- C4 C7 CB D0 D4 D8 DD E0
8050- E3 E7 E9 EB EF F3 F5 F8
8058- FC FD FE FF 00 01 04 06
8060- 07 08 09 0C 0F 12 15 18
8068- 1D 20 23 26 29 2C 2F 32
8070- 35 38 3B 3F 42 46 49 4C
8078- 50 55 5B 5F 00 00 00 00
8080- D0 D0 D0 D0 D0 D0 D0 D0
8088- D0 D0 D0 D0 D0 D0 D1 D1
8090- D1 D1 D1 D1 D1 D1 D1 D1
8098- D1 D1 D1 D1 D1 D1 D1 D1
80A0- D1 D1 D1 D1 D1 D1 D1 D1
80A8- D1 D1 D1 D1 D1 D1 D1 D1
80B0- D1 D1 D1 D1 D1 D1 D1 D1
80B8- D1 D1 D1 D1 D1 D1 D1 D1
80C0- D1 D1 D1 D1 D1 D1 D1 D1
80C8- D1 D1 D1 D1 D2 D2 D2 D2
80D0- D2 D2 D2 D2 D2 D2 D2 D2
80D8- D2 D2 D2 D2 D2 D2 D2 D2
80E0- D2 D2 D2 D2 D2 D2 D2 D2
80E8- D2 D2 D2 D2 00 00 00 00
80F0- 03 03 04 04 05 03 03 04
80F8- 02 04 03 03 04 04 04 04
8100- 04 03 07 05 04 05 04 04
8108- 04 06 06 05 07 06 07 05
8110- 06 03 04 06 06 05 06 06
8118- 05 06 03 04 03 02 07 01
8120- 05 06 03 04 02 04 04 04
8128- 03 04 05 04 04 05 03 03
8130- 04 02 02 04 04 02 03 04
8138- 01 01 01 01 01 03 02 01
8140- 01 01 03 03 03 03 03 05
8148- 03 03 03 03 03 03 03 03
8150- 03 03 04 03 04 03 03 04
8158- 05 06 04 00 00 00 00 00
```

```
8160- 20 F8 E6 86 E0 20 8E FD
8168- A6 67 A4 68 86 E2 84 E3
8170- A0 01 B1 E2 D0 02 60 00
8178- 20 8E FD A0 02 B1 E2 85
8180- 9F A0 03 B1 E2 85 9E A0
8188- 04 84 E4 A2 00 A5 9E C9
8190- 28 B0 2F C9 27 90 08 A5
8198- 9F C9 10 B0 25 A5 9E C9
81A0- 04 B0 1B C9 03 90 06 A5
81A8- 9F C9 E8 B0 11 A5 9E D0
81B0- 0C A5 9F C9 64 B0 06 C9
81B8- 0A B0 01 E8 E8 E8 E8 20
81C0- 4A F9 20 28 ED 20 39 82
81C8- 20 54 82 A4 E4 E6 E4 B1
81D0- E2 F0 3C 30 07 20 3F 82
81D8- 4C CB 81 00 29 7F 85 E5
81E0- A8 B9 10 80 BE 80 80 85
81E8- E6 86 E7 B9 F0 80 C9 01
81F0- F0 09 A6 E1 E0 06 F0 03
81F8- 20 3D 82 A2 00 86 E8 A4
8200- E8 E6 E8 B1 E6 48 20 3F
8208- 82 68 10 F3 30 11 00 A0
8210- 00 B1 E2 AA C8 B1 E2 86
8218- E2 85 E3 4C 70 81 00 A4
8220- E5 B9 F0 80 C9 01 F0 A3
8228- A4 E4 B1 E2 30 9D C9 28
8230- F0 99 20 3D 82 4C CB 81
8238- 00 A9 00 85 E1 A9 20 09
8240- 80 20 ED FD E6 E1 A5 E1
8248- C5 E0 90 0C 20 8E FD A2
8250- 06 20 4A F9 A2 06 86 E1
8258- 60 00 00 00 00 00 00 00
8260- 00
```

```
]
]&100
```

```
  1 : REM  APPLE II 48K
  2 : REM  THE NICELISTER
  3 : REM  WRITTEN BY  T TSE
  4 :
  5 : REM  (C) COPYRIGHT 13-OCT-1981
  6 : REM  ALL RIGHTS RESERVED
  7 :
100 TEXT : HOME
110 VTAB 3: PRINT "NICELISTER"
120 VTAB 6: PRINT "WRITTEN BY T TSE"
125 VTAB 7: PRINT "****************"
130 VTAB 9: PRINT "(C) OCTOBER 1981"
140 VTAB 13: PRINT "TO USE :-"
150 VTAB 15: PRINT "]PR#1    - FOR PRINTER OUTPUT"
155 VTAB 16: PRINT "           PRINTER ASSUMED IN SLOT 1"
160 VTAB 18: PRINT "]& 39    - TO LIST ON 39 CHRS PER LINE"
165 VTAB 19: PRINT "           NUMBER RANGE : 7-255"
200 PRINT CHR$(4)"BLOAD NICELISTER"
210 POKE 1014,0: POKE 1015,128
220 PRINT : PRINT "READY !": END
```


apple computer
Sales and Service

kay-jay studios
135 COLES GREEN ROAD LONDON NW2 7HN
TEL: 01 450 6993

COMPUTERS	BY APPLE
MONITORS	BY KAGA
PAPER	BY THE THOUSAND
FINANCE	BY BOWMAKER LTD
DISKETTES	BY BASF
CASSETTES	BY AUDIO MEDIA
PRINTERS	BY EPSON & QUME
SOFTWARE	BY MANY
BUGS	BY MISTAKE
PLASTIC MONEY	BY ACCESS & BARCLAYCARD
PRICE LIST	BY REQUEST
PERSONAL SERVICE	BY KEN JONES & HIS CREW

applefan™ by hiteck

Are you overheating?

Cool things down with the APPLEFAN!
Your Apple will not only run more efficiently, with less risk of breakdown, but will be well organised, with easy access to all input/output ports.

Simple to fit, no soldering or drilling – the APPLEFAN still allows access to the lid of your Apple.
APPLEFAN may be removed at any time.

Basic model only **£68.60** +VAT.

Ask your Apple dealer for a demonstration, or post the coupon for our brochure!

Hiteck Products 21 Station Rd, Knebworth, Herts. Tel 0438 812137

Please send me your brochure of APPLEFAN!

NAME: _____

ADDRESS: _____

_____

_____

FIRST DIVISION COMPUTERS.

Snodene , Style Loke , Barford , Norfolk NR9 4BE. (060-545-481)

The Following Items are for sale to B.A.S.U.G. Members :-

1 ANADEX DP8000 Printer. Secondhand £200.00

1 PAPER TIGER 445 Printer with Parallel Interface. NEW & UNUSED. £320.00

2 MICRO-CONNECTION Telephone interfaces with Diplomat Card & Computech software (used for evaluation purposes only). £250.00 Each.

2 C.I.C.C. Mercury Operating Systems (used for evaluation purposes only). £250.00 Each.

1 Set TABS Software including hardware card OFFERS?

50 Walters BD136 High Speed Matrix Printers (240 c.p.s. Logic Seeking Bi-Directional 10K buffer) £450.00 Each.

The BD136 Printers are still selling for £1,200.00 , and are Brand New. We have purchased them as a result of a take-over and at £450.00 they must be the bargain of the Decade. These are the printers which First Division Computers install at leading Football Clubs for ticket printing. FIRST COME FIRST SERVED.

All the above prices EXCLUDE V.A.T. and are for C.O.D. or Pro-Forma only.

SPECIAL OFFER TO BASUG MEMBERS

HIGH QUALITY FANTASTIC PRICE

5¼" DISCS FOR APPLES WITH **2 YEARS WARRANTY**

OUR PRICES INCLUDE VAT

1+ Boxes (10 DISCS)	£16
10+ Boxes	£15
100+ Boxes	£14

+ Postage £1 for up to 5 boxes £2 for 6 boxes

CS

COMPUTER SCIENCE
Beacon House, P.O. BOX 46
London N20 0YA
Telephone: 01-368 3290
Telex: 83247 INTLAB

NOW YOU HAVE AN APPLE MICROCOMPUTER YOU'LL NEED

NEW

OMNIS®

All you've ever wished for in an information management system.

OMNIS sets new standards in database programs and levels of performance that you never believed were possible on a microcomputer.

- **OMNIS** is written in UCSD Pascal+, this means a better structured, faster running set of programs than could ever be possible using Basic — We believe that UCSD Pascal+ is the best microcomputer language available — OMNIS proves it —
- **OMNIS** is structured around powerful file handling modules. These modules give you the flexibility to store and retrieve your information in the way that you want. Full multi-key indexed access is available to all your database files, you say what you want — OMNIS does the rest.
- **OMNIS** provides you with a versatile report generating module that enables you to define your own reports, lists, mailing labels etc.
- **OMNIS** has unparalleled search facilities to allow you to be selective. Those hours of fruitless searching through rows of card indexes becomes a thing of the past.
- **OMNIS** lets you design your own screen layouts for data entry and inspection — you may have up to 10 screens per file.

OMNIS has an application waiting for it in every business, school and laboratory and workshop. Wherever information needs to be stored and retrieved, OMNIS is available for both APPLE II and APPLE ///. We can also supply OMNIS for use on APPLE microcomputer networks (yes, with true multi-user record locking). Trade enquiries welcome.

All registered users of OMNIS will be sent **FREE BACKUP** disc and you will be kept informed of all updates and upgrades. Free help will be given to all registered users via an OMNIS hotline.

OMNIS — All you ever wanted

APPLE II* version - £174.00 (incl VAT & pp)
APPLE ///* version - £225.75 (incl VAT & pp)

BLYTH COMPUTERS LIMITED
Wenhaston, Halesworth, Suffolk
IP19 9DH

☎ 050 270 565
24 hour phone service

*trademarks of APPLE Computer Inc.
+ trademark of the Regents of the University of California, San Diego.

We welcome Access | BARCLAYCARD VISA

® Registered Trade Mark

To: Blyth Computers Ltd., Wenhaston, Halesworth, Suffolk IP19 9DH.
PLEASE SEND ME ☐ **APPLE II VERSION** ☐ **APPLE /// VERSION**
Cheque enclosed or please charge my Access*/Barclaycard no:
*Delete as applicable
Name (Block Capitals)
Address
Tel. No

H1

The biggest Apple event ever held in Britain!

apple '82

® 1st National Apple User Convention

Whether you're an active Apple user, or just fascinated by the rapid development of microcomputing generally, you won't want to miss the action-packed weekend that will make up Apple '82.

From Friday, June 4, to Sunday, June 6, the whole of the ultra-modern Fulcrum Centre in Slough will be completely devoted to the onward march of the micro, when some of Britain's top computing experts will be revealing their secrets.

FRIDAY
is education day – the staging of the first National Apple Education Forum and a chance for teachers and lecturers to exchange ideas, evaluate software and listen to a series of lectures covering every aspect of computer-assisted learning. Some 25 Apples will be on show, demonstrating a wide range of applications in the school environment.

SATURDAY and SUNDAY
will be for users generally – the first National Apple Users Convention. So many leading figures in microcomputing want to take part that presentations will be given simultaneously in two adjoining theatres throughout the weekend.

The full timetable of events covers database systems, graphics, music and speech synthesis, Pascal, Cobol and other languages, commercial and industrial applications, hardware and software troubleshooting and micros in medicine.

A central feature of the convention will be a communications workshop, to explore latest developments in linking Apple to Apple, Apple to mainframe, remote information retrieval systems and bulletin boards. It will give a unique insight into a subject that is rapidly becoming one of the most exciting aspects of computing today.

And for light relief, there will also be the national finals of the nail-biting Apple Olympics.

Mail the coupon below for full details of plans for Apple 82 – and about the first-class accommodation that can be reserved for you at some of the best hotels in the area for a modest £17 a night – far below their normal rates.

This major event in the Apple world is attracting users from all parts of the British Isles and overseas. But tickets are limited, so early booking is advisable.

Please send full details of Apple '82 to:

Name _____

Address _____

_____

_____

POST TO: Apple '82, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

ozwise

OZWISE WISE BUYS

PRINTERS?
OZWISE GIVES YOU THE BEST PRICES!

Epson MX80 F/T 2 only £339
Epson MX100 only £479
NEC 8023 only £339
Integrex CX80 colour only £729

Best prices on Anadex, Anacom, Centronics, Epson, Mannesman Tally, Nec, Oki, Olivetti, Olympia, Paper Tiger, Qume, Ricoh, Tec

WE SUPPLY OVER 40 PRINTERS!
12 DIFFERENT MONITORS!

12" green screen monitors from £95
14" colour monitors from £229

OVER 60 INTERFACE & EXPANSION CARDS!

Ramex 16K RAM card only £69
Videx 80 column card only £159
CPS multifunction card only £119
Z80 card only £89

OVER 100 UTILITIES, COMPILERS & LANGUAGE SYSTEMS!

Tasc Applesoft compiler only £89
Lisa Assembly Lang Develp Sys £59
Pascal language system only £209

OVER 100 BUSINESS PROGRAMS!

DB Master only £109
Zardax word processor only £139
Visicalc only £94
Visipak (calc/trend/file) £329

OVER 230 EDUCATIONAL & GAMES PROGRAMS!

Wizardry adventure game £25.95
Zork adventure game only £23.95
Lettermind mastermind game £12.95

ALSO DISCS, RIBBONS, ETC AT UNBEATABLE PRICES!
OVER 700 APPLE ITEMS!

Write or phone for free catalogue
Please add 15% VAT to orders
Access/Barclaycard holders telephone your order

VISA

Write or 'phone for full details

ozwise computers

28 CROFTS ROAD, HARROW, MIDDX. HA1 2PH.
01-863 2309 24 hour service

COMPU-READ

A Review by Norah Arnold

The set of programs known as COMPU-READ was written by Steven W. Pederson and Sherwin A. Steffin and are distributed by Edu-Ware Services, Inc. of California. They consist of four skill-building programs which are designed to increase certain aspects of the user's reading ability. Also provided are two programs which are specifically for creating new word-files to be employed in two of the skill-builders.

When booted the disk gives a colourful introduction then asks whether instructions are needed. If you opt for the instructions you are presented with several well arranged screens of information which make good use of inverse for added clarity and emphasis.

According to the objectives given in the instructions the programs are aimed specifically at increasing a person's reading speed and also for improving retention of words read. It should be obvious from this that they are definitely not programs to help with the initial acquisition of reading skills by very young children but may be used after this period to increase motivation and to remedy difficulties such as letter reversals.

The first of the skill-building programs is called Compu-Read I and is concerned with letters only. The user can control the number of attempts and the starting speed of the test (1 to 21). Three single letters are displayed spaced accross the screen horizontally. The user must retain the three letters and type them in after they have disappeared from view. If the attempt was successful then the next three letters will be shown on the screen for a shorter period of time. However, if you reach a point where you are unsuccessful then the program reacts and slows down the display until a correct response is achieved. The slowest display time appears to be 1.33 seconds giving a reading rate of 2.25 letters per second. This may seem fast but a teen-ager with no reading problems can get down to a display time of 0.06 seconds giving a rate of 50 letters per second within a few minutes of using the test.

Compu-Read II moves on to present not letters but words, phrases or short sentences at very high speed. This is the only Compu-Read program where the progress of the user does not control the display speed. Each time a correct response is made new words or phrases are presented. If the response is incorrect the same word is repeated. A file of 100 words to be used with Compu-Read II is provided on the disk but new files can be compiled by using File-Builder II. The screen prompts controlling the entry procedure for the new file are very clear and easy to follow. However, thorough preparation of material is necessary before starting to enter.

Compu-Read III appears to be the most demanding of the programs. Again the number of attempts to be made and the display time can be controlled by the user. One word is displayed on the left of the screen for a short time. On the right side a list of four other words is shown. Hidden in this list is a word which is either a synonym or an antonym of the word shown on the left. All five words have to be taken in simultaneously, then one must decide which word means the same or the opposite of the original shown on the left.

Correct responses are confirmed and the display time is decreased accordingly. Corrections are given if the response was wrong and the display time is increased. Display speeds start at 1.26 seconds giving a reading rate of 238.09 words per minute. Both my teen-age son and I achieved success at a display time of 0.06 seconds which according to Compu-Read gives a reading rate of 4999.8 words per minute, which seems rather high to me.

Again a file-builder program is provided so that new source lists can be built up at any difficulty level which you choose.

In Compu-Read IV sentences are flashed on the screen for shorter and shorter times. Four different types of sentence construction are used and individual words in the sentences are altered to give variety. The user controls the starting speed and the number of attempts. After each sentence is displayed a question about it is presented. A word from the sentence usually has to be reproduced as the answer. This program also responds to the standard of the user, becoming faster or slower as necessary.

The slowest display time for each sentence is 1.26 seconds, giving a reading rate of 380.9 words per minute. The fastest time we have achieved gave a rate of 48000 words per minute! Most of the questions asked seem to require only part of the sentence to have been read, nearly always the first few words. This appears to lead to highly inflated scores.

A clear, concise performance report can be obtained after using any of the four programs. Children with whom I have used the programs have seen them as a challenge and responded accordingly. The programs certainly increased motivation and led to a great interest in achieving a higher score than previously.

The Compu-Read programs present some interesting possibilities for research, especially in view of their adjustment to the user's performance. Personally, I am hoping to conduct a small research program based on certain aspects of Compu-Read. The Compu-Read documentation states that Edu-Ware will support research efforts with listings or other information.

CS TEXTMASTER - A NON-TECHNICAL VIEW OF THE LATEST WORD-PROCESSOR FROM AUSTRALIA

By G.P.Ognibeni

Textmaster by Computer Solutions is the latest in the long line of Apple word-processors and is quite clearly aimed at the professional market. At £160 + VAT it is rather more expensive than most and must certainly be measured against such an established competitor as Supertext II which is priced at £125 + VAT but can be bought cheaper than that.

Textmaster sports quite an impressive array of features. It has LOWER CASE in software, SHIFT ON THE SHIFT BUTTON (where it belongs - have you ever seen a car which had its clutch on the passenger side?) works on the ordinary Apple 40-column display but can also be used with any of the better known 80-column cards. It is a nice touch to have all the necessary software for the various cards already provided on the program disk - not all of us (myself included) are adaptation wizards and in any case the quality of professional software should in no small degree be measured in terms of user friendliness. This must indeed be Textmaster's strongest area. As I have learned to use both in the last 6 weeks, I feel confident to predict that anyone who has struggled through the Supertext-jungle will find the logical structure of Textmaster an absolute joy. There are no numbers to remember, no sequences to be got right, no distinctions between "add mode", "change mode", and "cursor mode" (Supertext) which, in order to complicate things, each use different controls but all the editing can be done simply from any point in the text without any fiddling about. Both arrows are destructive which makes forward and backward deletions easy whilst movement around the text is effected by CTRL-plus D= down, U= up, L= left, R= right, B= beginning of text, E= end of text. The first four of these can be used in conjunction with the repeat key and while I find the stretch for CTRL-R awkward, I defy anyone not to remember the necessary commands. In addition, CTRL-V will move the cursor down 10 lines, CTRL-shift-N up 10 lines.

THE MENU: Textmaster uses a main menu and an inner menu. The main menu displays the files on disk and the options available. You choose between C= create a new document, R= retrieve a document, i.e. load an existing file, P= print a document from disk, T= transfer a file from one disk to another, L= lock, U= unlock a file, G= glossary: this decides which glossary (see below) you want to use, N= newdisk, initialize storage disk, and I= index, print a copy of all file names on the disk. All options are accessed by their initial letters, all files by a 2-character code, so that all you do to load a file is to press RA4. If you have changed disks in between working on Textmaster and you want to see the directory of the new disk, all you need to do from the main menu is to press the space bar and the directory will be displayed. Very useful in this context is the inclusion of a free sectors figure in the top left corner. Obviously useful in an office environment is the option of including a 3-character "author" code in the file name. Once inside the document, you can approach an inner menu by pressing the escape key. You then have the following options: C= change (or examine) the text of the document in memory, D= draft: print a single copy of file in memory, M= go to main menu, P= print document in memory, R= rename document in memory, S= save onto disk, V= videoprint file on screen as it will be printed on paper. You can change freely between the various commands of the inner menu except for option M= go to main menu which will delete the file from the computer's menu. Escape will get you out of all sorts of mistakes and simply return you to whichever menu you are working from, again very nice.

Once INSIDE THE TEXT, you get all the usual functions that you would expect except for the number of bytes left available. Here, sadly, you are left to guess. Textmaster does display the cursor position inside each paragraph which can be useful if you are tabbing, etc. There is, of course, a global find and replace function built in which asks if you want to change "Y/N/A", i.e. change just this one word, leave it as it is or change every occurrence in the text. Pressing the space bar will drop you out of the find and replace mode at any point. Supertext at this point allows you to "Autolink" your files so that you could search through an entire disk but Textmaster does not have this feature. It can, however, link documents for printing very easily, and will allow you to "cut and paste", i.e. to combine paragraphs of text in any way you like. You can also insert (CTRL-I) other documents into the file you are creating although with the cut and paste option there is less need to do this than with other word processors.

TEXT MOVEMENT: This to me is Textmaster's weakest area. The only way of moving text is by moving paragraphs up or down. This in itself could not be easier, however as every carriage return is interpreted as a new paragraph, moving text in large documents can be a pain. The use of markers would have made such operations a lot easier and I think these should be included in future versions of Textmaster.

GLOSSARY: Textmaster has a wonderful little gimmick which I find extremely useful. It can store up to 26 words or phrases which can be up to several lines long and retrieve these at any point in the text by a simple CTRL-G plus code letter A-Z. This can reduce the typing of repetitive phrases, book titles, formulae etc. very considerably. In addition, several glossaries can be used on the same disk and could even be swapped in the same document. A great idea!

As already indicated, Textmaster appears to be strongly business/ office orientated and has some very impressive functions for writing circulars, etc. by defining "labels" and then filling them with "info sets", e.g. addresses, titles, etc. You can also print "footers" at the bottom of pages, either throughout or selectively. The documentation provides quite a few examples (in plain English, what a relief).

FORMATTING: There is a comprehensive range of formatting commands which can be inserted into the text to override the set of built- in default values. The possible options, like justification, page length, width, paper length, margins, page numbering, spacing, should be quite sufficient, although I should have like an option for setting top and bottom margins. As it stands, Textmaster will center the text according to the page length that has been set. Tabbing and indentation are very easy, and a real joy on an 80- column card (I wish I could afford one).

PREVIEW: Like Supertext, Textmaster has an on-screen preview facility (which it calls Videoprint) but unless you have an 80- column card you can only ever see the leftmost 40 columns. Supertext will let you determine which 40 columns you want to see, I suppose Textmaster is essentially an 80- column processor which I suspect some kind-hearted programmer has allowed to be used on a 40- column machine by paupers such as myself.

TEXT LENGTH: The maximum file length for a 48K Apple (or ITT) is 14K. The capacity is increased to 22K if a language/ RAM card is present and to 30.5K if you have both RAM and 80- column cards. The "Multiprint" text- chaining mode allows documents of max. 130K to be printed in one go.

LOWER CASE: Textmaster has a very (!) readable chunky character set with descenders in software. To get Textmaster working at all, you need a "dongle", i.e. a 16- pin header plug in the game I/O socket. This is not a security dongle (as in the Hayden compiler) but merely serves to sense whether the shift button is being used. The price includes fitting but anybody with a new Apple should be able to fit one himself, all you need to do is to connect pin 2 of the game socket to pin 2 of the keyboard and game socket 3 to keyboard socket 24. That's all. Instructions for fitting to older keyboards are included in the documentation. I use an ITT 2020 with a Cherry keyboard and fitting should be within most people's ability (but don't overcook your electronics). ITT users should either clear the Hi-res pages before using Textmaster (just build "HGR" and "HGR2" into the "HELLO" program - Textmaster lets you play around with it and displays a full catalog of peripheral programs). You will then get the odd dot or very small line on your screen but this should never be a problem. Or else, here's your justification (at last) for fitting Jailbreak and enjoying a nicer (chunkier) character display with no interfering dots on Textmaster AND have fun playing games (in spite of this mention, I shall still have to pay the full price for my Jailbreak).

SPEED: Textmaster is remarkably fast for a word processor which provides lower case in software. I should say that it is definitely faster than Supertext and in any case delays are only noticible when backward- deleting. Scrolling is not as smooth as on Apple Writer but about on a par with Supertext. The screen goes blank briefly when the 10 line jump in the text is implemented and goes altogether blank during printing.

VERDICT: Textmaster is a very good word processor with outstanding user friendliness. It could have been even better if one or two other functions had been built in but this little niggle pales into insignificance once you start using Textmaster. Should you splash out 180 odd pounds? Yes, if you can afford it and have a lot of word processing to do, otherwise I should put up with shift/ escape schizophrenia and stick with Apple Writer.

TYPE-RIGHT

Reviewed by Fran Teo

Have you ever cursed, sweated blood and ended up with your fingers in knots manoeuvering the 'ESC' key around when using Apple Writer? I have. I have also gone nearly blind trying to read the text on the screen when checking to see if I had all my capital letters in the correct place, so I was very eager to get my hands on Microsource's Upper and Lower case conversion.

I was not disapointed! To be able to use my Apple Keyboard as I use an ordinary keyboard has proved a great advantage to me and, I feel, to anybody who has a lot of typing to do. Also, as Upper case is now generated using the shift key, I found that I save a lot of time editing as I only have to press the 'ESC' key once to get into editing mode (I cannot stress the benefit of this enough). Luckily all the old files that I created using the original version of Apple Writer are still usable. They appear as upper and lower case on the screen although one or two odd characters are still in inverse, but this has no adverse effects.

Type-Right consists of two plug-in adaptors, one to enable the keyboard to recognise Upper AND Lower case letters, and one to correct the display accordingly.

The manual explains how to connect the software onto your own disks although my secretary felt that it was a little too technical for her and had to refer back to me for explanation of some of the terms, so if it is being aimed at the 'user' market the documentation should perhaps be simplified. The manual is well written and easy to follow.

In addition to the word processing, you can also utilise the Upper and Lower case ability within your own Basic programs. Having delved into this aspect I came to the conclusion that there are no distinct advantages (or disadvantages), it is a matter of personal taste. Although it is nice to know that you have the choice!

A couple of minor points. It would be useful to be reminded in the manual that you have to FID the Print Constant programs from your original disk. I know it sounds stupid but I forgot! Also if you dive straight out of editing into Basic, the Apple will not accept ESC I etc. unless the switch is off, or the Shift key is pressed. I have sworn countless times when forgetting that I have a manual switch to manoeuvre around. But, as I said, these are minor points. The advantage of an upper and lower case keyboard far outweigh these disadvantages.

In reviewing Type-Right I have to admit I have not explored the French, German or Scientific notation. I am sure that if you need these applications they will be very worthwhile.

CONCLUSION

If you do a lot of typing in connection with Apple Writer and/or work with other languages Type-Right is a must! Aesthetically it looks good in your own programs but, as I said before, it is a matter of personal taste.

Type-Right is available from Microsource for £46.50 inclusive of V.A.T. This price includes the Software and the conversion kit for the Hardware.

AUTOCRAT COMPUTERS

SPECIAL OFFERS

SAVE £295
FOR PROFESSIONAL
WORD PROCESSING RESULTS
TEC DAISY WHEEL PRINTER
ONLY £995
FAST-40cps, Quiet and efficient

Change up to 3.3
Update your 3.2 APPLE SYSTEM for only
£32.00

SAVE ON 5.25" Floppy Disks
MEMOREX Hub reinforced S/S, S/D disks,
ONLY
£14.95 (pack of 10)

STOP PRESS.....

*** EPSON MX-80 F/T PRINTER ***
ONLY
£339

ACCESS BARCLAYCARD

VAT EXTRA

AUTOCRAT COMPUTERS,
264 Preston Road,
Harrow, Middx., HA3 0PY.
Tel: 01 908 3636

APPLE PILOT

Reviewed by Leo Crossfield

Introduction

Each and every educator, whether teacher or employer who is responsible for the training of his or her staff, realise the need for individual and interactive instruction. The reasons for the need of interactive learning are numerous, but the main reason is that interaction focuses interest and focused interest aids both attention and consequently retention. There is a great deal of psychological research within the area of learning and retention, but I feel that we do not need to become too involved with detail, for it appears to be sufficient to suggest that the main elements that all educationalists are concerned with are firstly maximum understanding of content and secondly maximum retention of detail. Indeed learning curves are an interesting phenomenon, for research shows that over time a student's recall, of learned material, grows progressively worse; with approximately 75% retention after a one hour lecture decreasing to less than 30% after a five hour duration. This retention can of course be increased by personal interaction between teacher and student but in this age where time is so precious how can we cope?

The development of 'Programmed Instruction' (P.I.), has aided many educationalists, via self instruction manuals wherein a student is tested at set stages of the text, and dependent upon whether the test objectives have been achieved or not, the student will be told to branch to a more complex area of the manual or to branch back to a more detailed study of the problem area. Many companies have used this method of training in which the student will only progress upon the measurement of his understanding and retention of the subject matter. This method of instruction 'The Programmed Instruction Format' has been highly successful, but the individual student is still bound to a set text. The fact is that any instruction via manuals and books are less exciting because of the lack of both visual and audial stimuli.

2. Pilot Overview

The above trend of thought has led us to an almost inevitable conclusion; everything points towards the extensive use of computers in education. Indeed, Computer Assisted Instruction (CAI) is developing fast, for with the aid of micro technology, we are now being offered small portable machines that are robust in construction and flexible in application, as well as offering sophisticated colour graphics and complex sound generation, all at a reasonable price. These developments have given CAI a whole new dimension. CAI has developed a long way since Dr. John A. Starkweather in 1968 (while at the University of California, San Francisco) decided to write an interpreter for a high level language that was designed specifically for teachers.

The aim was for teachers, without any prior programming experience, to develop their own educational and instructional programs, both easily and quickly. The Original Pilot language was developed in the early 1970's, in an attempt to develop an easy method of program development so that instructors could concentrate upon their own specific disciplines rather than spending vast amounts of their time and energy learning a somewhat obscure high level programming language. The Original Pilot had only eight instructions and was very limited in its application, but since that time it has been revised and extended many times.

Today we have a version of Pilot that is the leader of all it's contemporaries. It has been made available on one of the most popular micro computers, namely the Apple II. Apple Computer Inc., have developed Apple Pilot in order to take advantage of the Apple's superb abilities to generate both sound and colour graphics, while still maintaining the basic tenent of the Original Pilot: Teachers, not programmers, will be Pilot's intended market. With this in mind Apple Computers Inc., have developed the language to contain a maximum of twenty-two instructions, of which the user will generally use only nine for writing the largest portions of a lesson.

3. Apple Pilot: The System

When you purchase Apple Pilot you will receive two manuals and three 5 1/4" floppy diskettes, the "Author" diskette (with a backup copy) and the "Lesson" diskette. The 'Author' diskette contains four editors to enable the Author to create sound, text and graphics files. It also encompasses a compiler to enable the Author and student to execcute the final program. On the 'Lesson' diskette there are a number of lessons that serve as examples to the user. By running these programs, one can get an idea of how one can use CAI to maximum advantage, and by looking at the source file one can see how the program was written.

The two manuals are beautifully presented, keeping up with the Apple tradition. They are clear and to the point. The 'Editors Manual' gives an overview of the system and explains how to use the four editors. And finally it leads through graphic animation citing the demonstration lessons as an example. the 'Language Reference Manual' explains the use of different instructions, special effects using games paddles, graphics and sound, file handling, hints for beginners and Advanced Programming. Each section has plenty of examples and is outlined in a very clear manner.

In order to use Apple Pilot you will require a 48K Apple System, a monitor and 3.3 DOS. Although Apple Pilot is written in the language Pascal you do not need either the Apple Language System, or a Ram card, to use it.

4. Apple Pilot: The Language

(a) Text Instructions.

The text editor is the main key to writing a program with Apple Pilot. All of the program instructions are created from within this editor. The Language does not require line numbers, like some other high level languages.
As an example, if we wish to print the word "HELLO" to the screen the instruction that we would insert, using the Text Editor is:

```
T: HELLO
```

This instruction will print 'HELLO' to the screen.

The format of Pilot is generally this simple. 'T:' represents 'Text'. If we require documentation of each instruction embedded within the program text, the command to use is 'R:' which represents 'REMARK':

```
R: This section is documented.
```

Variables are dealt with by first dimensioning their length:

```
D: F$(30)
```

This instruction dimensions the string variable 'F$' to a length of thirty characters maximum.

One can accept a user's input response with the instruction 'A:',which represents 'wait and accept an input response from the user'.

```
A: $N$
```

This instruction accepts the user's input and stores it in the variable 'N$'.
The response, which is stored in the variable N$ can then be compared to a list of possible correct inputs which is formulated under "the instruction 'M:' for 'Match'. For example:

```
M: YES! OF COURSE! YEAH! WHY NOT! POSSIBLY.
```

Each possible response is seperated by a 'shriek' (exclamation mark) and can then be compared to each element of the match list and tested.

(c) Flow Of Control Instructions.

The use of 'Jump' instructions are also available within Apple Pilot. One can label a section by first inserting an asterix '*' and then the desired label. For example:

```
* START
```

If the match is found to be an incorrect response one can issue a conditional jump with the instruction 'JN:'.

```
JN: START
```

This instruction will jump to the label "start" if the match was not successful.

There are many different jump instructions including an unconditional jump instruction 'J:'.

```
J: POINT
```

This instruction enables the Author flexibility within the program's flow of control. The 'J' instruction will jump to the label "POINT" when incountered within the program's flow of logic.

One can also time the student's responses with the following instruction:

```
C: T2=TIM (0)
```

This instruction will store the student's response time in the numeric variable T2. To display this time to the student one can output the response time with the following instruction:

```
T: You took £T2 seconds
 : to answer the question.
```

(d) Computational Instructions.

Apple Pilot includes the dimensioning of numeric and string variables, as well as arrays. The following instruction is a computational instruction.

```
C: N = 2 + 3/X
```

The 'C:' represents 'COMPUTE', and the above instruction assigns the variable 'N' the value of 2 + 3 divided by the variable 'X'. As one can see, all the arithmetic operators are available for computation.

(e) Graphics Instructions

One can use the graphics editor to create complex and quite dynamic high resolution colour pictures. These pictures can be easily utilised from within the Pilot program by using the 'GX:' instruction.

```
GX: Picture
```

This instruction executes the graphics display called 'PICTURE', where "picture" is the name of the graphics screen previously created using the Graphics Editor.

Apple Pilot's graphics commands are very powerful, for instance one can plot points, lines, draw, change colour and move the graphics cursor, all with simple commands from the Text Editor. For example to plot an X, Y co-ordinate, the command is:

```
G: Px,y
```

One can give a single "G:" instruction followed by numerous commands, as long as each command is separated by semi-colons.

A clever detail that Apple Computers Inc., have obviously well thought out is that they use the Graphics screen for all graphics and text display. This enables the Apple to produce both upper and lower case text, mix text and graphics and lastly it enables the user to define his own character fonts and to use them as if they were standard from the keyboard. There are a number of graphics commands enabling the Author to display graphics, text and mixed mode. Therefore one can limit the screen to a specified text size whilst using the rest of the screen for a mixture of text and graphics. A good example of this would be to give a text label to a graphic picture of a drawn map, with a section of the screen dedicated to direct interaction with the student.
Different graphic colours are easily set and one can choose black, white, green, violet, orange, blue or a reverse mode of the above.

(f) Different Character Sets.

Apple Pilot also encompasses a Character Set Editor which enables the educator to define, and design, their own character sets.

TX: GREEK

This changes the present character set to the Greek characters (which Apple supply with the package). Apple also include "Katakana" and a character set called "Maxwell". Katakana is obviously Japanese while Maxwell is a special set of well defined pieces of a little man. By manipulating these characters one can get Maxwell to walk and run about the screen.

As I have mentioned one can easily define one's own character fonts thus creating a very powerful and flexible system. The systems limitations are one's own imagination.

(g) File Handling

If one needs to maintain file structures then Pilot enables the 'Author' to easily create, open, write and generally access data files. Many Authors will not require any file handling but it is quite nice to maintain a past-record of a student's ability on diskette. Such a record can be easily accessed each time that particular student uses the program and can be used to inform that particular student of his past success rate.

(h) The Sound Instructions

So far I have covered most of the different aspects of Apple Pilot, although each area has much more depth than I am able to cover. Nonetheless the last, and for some, the most relevant editor contained within Apple Pilot is the "Sound Editor". Within the Sound Effects Editor the author can compose a tune and when it is correct it can be saved to disk under a specified name. One can also create silly jingles that can be used to encourage correct responses, and of course less polite noises for incorrect responses.

In order to execute the tunes once they have been created the command 'SX:' will execute the relevant sound file.

SX: BEETHOVEN's 5th Symphony

This instruction will execute a sound file called "Beethoven's 5th Symphony".

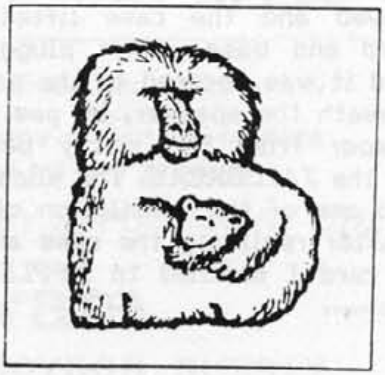
In conclusion, I have tried to give reasons why any institution that is involved in the process of education could benefit from the use of Apple Pilot. It offers a simple means in which Educators can access Computer Assisted Learning.

Within one half hour of using Pilot I was pleasantly surprised to find myself comfortably writing simple interactive programs. One can argue that the interactive programming languages such as Basic or Pascal can be just as easily used for CAL and indeed they can, but Pilot is written and intended to be used by the layman with little or no programming experience. It is deliberately structured with an understanding that the best people to design and write educational courseware are educationalists with specialist knowledge in their specific fields, which does not usually encompass computer science and skills in programming.

Pilot is modifiable and easily extended to encompass later developments. It's flexibility is almost limitless whilst it's use of menu driven routines give little chance for confusion.

Finally, each editor and instruction maintains a consistent format, which means that once one has learnt to handle the text handling routines, it does not take much further learning to handle sound, graphics or text files, for all are handled in a similar manner.

All in all I have found this package well presented, well written, and what is more, an excellent tool for use within all forms of educational environments. It is a superb step forward for CAI.



JAILBREAK

Reviewed by Chris Chapman

I have been using a JAILBREAK adaptor in my ITT2020 for 3 months at the time of writing. The following notes are intended to be a review of my experiences.

The need for a hardware modification to the ITT's graphics becomes obvious when one tries to run commercial Hires programs such as Space Invaders. If such programs are written in machine code (and the best ones are) , then the 9th bits of the ITT's Hires display wont be accessed. This is quite reasonable because the programme was originally written for an APPLE! The effect is that the game is played through a set of thick vertical lines like looking through a prison cage. Soon after buying an ITT most people learn that accessing the switch in memory position $C0FE turns the 9th bits off and the display is then much more acceptable. The bars are now black and against a black background the bars are invisable until a shape passes through the line when it will loose its middle. In the past people have had to live with this feature but recently a one wire hardware mod was published in HARDCORE which fills the gaps by copying in the dots from the sides. This is great BUT once installed running PALSOFT programs gives problems so you take it out then put it in etc.

JAILBREAK is the most complete answer to the problem that I know. For about $30 one acquires a small printed circuit board of about 4 x 1 , a ribbon cable and header plug , a 2 position switch, and instructions.

Installation is said to involve removing the IC at A6 next to the keyboard plug, plugging in the header socket from JAILBREAK and inserting the IC you took out into the empty socket on the little board. I found that in order to reach this area of the main board I needed to remove the case from my ITT. I therefore removed the 6 screws from the base and the 4 screws under the front of the keyboard. The keyboard plug was then removed and the case lifted from the motherboard and base. After plugging in the small board it was secured to the base next to and underneath the speaker, by peeling off the backing paper from the sticky pads on the reverse of the JAILBREAK. The slider switch is bolted into one of the ventilation slots by the speaker. After replacing the case and the disc controller card I swiched to APPLE mode and powered up.....

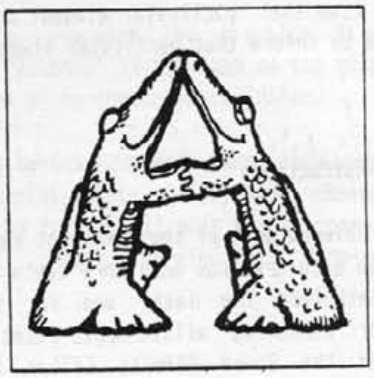
The result was that SPACE INVADERS (both the Appleware and Softwarehouse versions) ,APPLE GALAXIAN , SARGON , and MICROCHESS displayed perfectly. The display was just like an APPLE and occupied the full width of my monitor.

The instructions say that the colour information is lost leaving a monochrome picture. I usually use a b/w monitor but i plugged in our TV and ran APPLE 21 which has Hires playing cards on a green background. In the APPLE mode the background was green/black/red stripes ! I must add that the display is great - the cards are like on an APPLE with no black lines down the screen however a plain background does become striped.

I have a programme called L.A.MONOPOLY which has a Hires picture of the playing board and was written for the APPLE. The player's counters are DRAWen on the screen in the wrong place when run on an ITT and don't land on the correct squares ! The dice pips are also HPLOTed on the wrong part of the board. The 9th bit problems are solved BUT to change the dice/counter positions access is neccessary to the programme in order to modify the X,Y coordinates. This is rather awkward since the disc is somewhat protected and This took me some time to do. Another incompatability between PALSOFT and APPLESOFT is HPLOT x TO y TO z etc.

JAILBREAK on its own cannot help with the problems in the previous paragragh since they are caused by the BASIC interpreter. However, if one loads APPLESOFT (as opposed to PALSOFT) onto a RAMcard then this problem disappears, and one has to all intents and purposes a genuine APPLE II, albeit without the possibility of Hi-Res colour.

The instructions point out that by plugging in a second JAILBREAK in A1 you can fatten up your text characters. I haven't tried this since my text seems OK and so I cannot comment but I can really recommend one for Hires use.



Spider Software

OH NO – NOT ANOTHER DATABASE?!!

Some Questions and Answers on ACCESS – A new database management system from SPIDER SOFTWARE.

How many records can I have?
This depends on the size of each record. The maximum record size is 1560 characters. The maximum number of records per disk volume is 7936 but this is dependent on the record size. As an example, if your records are 200 characters long, you may have a maximum of 671 records per volume. A maximum of 40 fields per record is available.

How long will it take to find a record?
A powerful advanced IRAM (Indexed Random Access Method) is utilised for major record retrieval purposes giving an access speed of either instant recall or within 3 seconds. Any field (or combinations thereof) with multiple search criteria will either give instant recall or will take a maximum of 23 seconds. On the SyMBfile hard disk everything is at least 7 times faster.

How long will it take to sort a disk full of information?
All sorting is done on an index. If the sort is on the primary index it will take 0.2 seconds regardless of the number of records. To sort on any field which is not indexed involves first creating an index for that field which is then sorted. The time taken depends on the record size (generally less than 3 minutes). Any index can be saved for later use or made into a primary index. Sorting a disk need not involve creating a sorted version of the database.

How many disk drives do I need?
ACCESS will ideally run on 2 drives. However, it will support a single drive system and a version is available for the SyMBfile 5 megabyte hard drive.

How about report formats?
Reports are user-configured and can contain report headings, column headings, column sub-totals, brought forward totals, grand totals, computed fields, page numbering etc. Reports can be on selected and/or sorted data.

What if the dog chews my program disk?
We provide copy routines for backing-up of the program disk and the data disks as many times as you require. The ACCESS system is a combination of hardware and software.

Is the program menu-driven?
YES. ACCESS constantly displays prompts indicating the options available wherever you may be in the program.

How is the data stored?
ACCESS creates and uses its own data disks. However, facilities are provided to enable you to produce standard DOS 3.3 text files in either sequential or random access format using any sorted or selected fields. Because of ACCESS's own data storage techniques a very large database may require more than one disk to store the text file(s) produced.

How easy is it to create records and edit them?
ACCESS has a powerful word processor style screen editor enabling insertion and deletion of characters, etc., full cursor control across fields and pages of a record. A maximum of 40 screen pages are available. Password protected fields are supported as are computed on-screen fields.

What if I delete a record by mistake?
ACCESS only marks a record as deleted. Facilities are given to either "un-delete" deleted records or purge deleted records from the database.

My current database takes ages for me to add and save records because it needs to re-structure the entire file to keep the "primary key" in alphabetical order. Will this happen with ACCESS?
NO!!! ACCESS uses logic and technique to handle your data; there is no reason (should you have the stamina) why you should not fill an entire disk with information as fast as you can type and immediately retrieve all the information in sorted order or order of entry, etc. All complex and time-critical functions including disk input and output, indexing, sorting, searching, screen display and editing are performed by ACCESS using powerful machine-code routines.

What hardware do I need?
48K Apple II Plus with DOS 3.3 and 1 or 2 disk drives. Most makes of printer are supported.

Why should I buy ACCESS and how much is it?
Most facilities in ACCESS are available in other comparably priced database managers. However, ACCESS is more powerful and faster than its competitors in each function. ACCESS has gone beyond the boundary of merely complex facilities, it is powerful and "intelligent" enough to make itself extremely simple to use. The retail price including VAT is **£199.95.**

SPIDER SOFTWARE

3-D Supergraphics – A 3-D game development system in colour **£23.95**
Apple Panic – Chase little apple up and down ladders **£16.95**
AppleWorld – Allows you to produce 3-D animated graphics **£38.95**
Autobahn – Road race game with sound and hi-res graphics **£14.95**
Beneath Apple DOS – The programmer's guidebook to Apple DOS **£11.95**
Borg – Can you out run and outshoot the dragon's henchmen? **£16.95**
Copts & Robbers – An adventure in the tombs of ancient Egypt **£19.95**
Cranston Manor – A new hi-res adventure like Wizard and the Princess **£19.95**
Dark Forest – An adventurous game of strategy for 1 to 6 players **£16.95**
David's Midnight Magic – Hi-res pinball at it's best **£19.95**
EXPEDITER II – THE APPLESOFT COMPILER AT A LOW, LOW PRICE **£56.95**
E-Z Draw 3.3 – The poor man's graphics tablet. Very easy to use **£28.95**
Galaxy Wars – Colour graphics + Sound effects + Hi-res arcade game **£13.95**
Genetic Drift – A departure from the normal arcade style game **£16.95**
Jawbreaker – Eat up the dots but watch out for the Gobblers **£16.95**
Missile Defense – The Apple version of Missile Command **£16.95**
Orbitron – Fight off enemy forces and avoid meteor showers **£14.95**
Outpost – Save your space outpost from enemy fighters **£16.95**
Pascal Graphics Editor – Professional graphics editor for Pascal **£49.95**
Pegasus II – The latest in pub games now available for the Apple **£16.95**
Pulsar II – 2 superb hi-res games on one disk **£14.95**
Space Quarks – Shoot down the quarks before they get you **£16.95**
Space Warrior – Protect your shields from the dreaded RAM ships **£13.95**
Star Cruiser – Save yourself from the swooping aliens **£18.95**
SuperScribe II – 70 col. word processor with true upper/lower case **£73.95**
The Arcade Machine – Design your own machine-language arcade games **£25.95**
Threshold – Super fast arcade style game **£21.95**
Time Zone – 1,182 room, 21 colour hi-res adventure **£54.95**
Track Attack – Steal gold from a moving train in this hi-res game **£16.95**
Ulysses & the Golden Fleece – Another adventure like Cranston Manor **£19.95**
Wizard and the Princess – Hi-res adventure in 21 colours **£18.95**
Wizardry – The highly acclaimed game reviewed in Windfall No. 6 ..... **£28.95**

SEE US ON STAND 29 AT APPLE '82

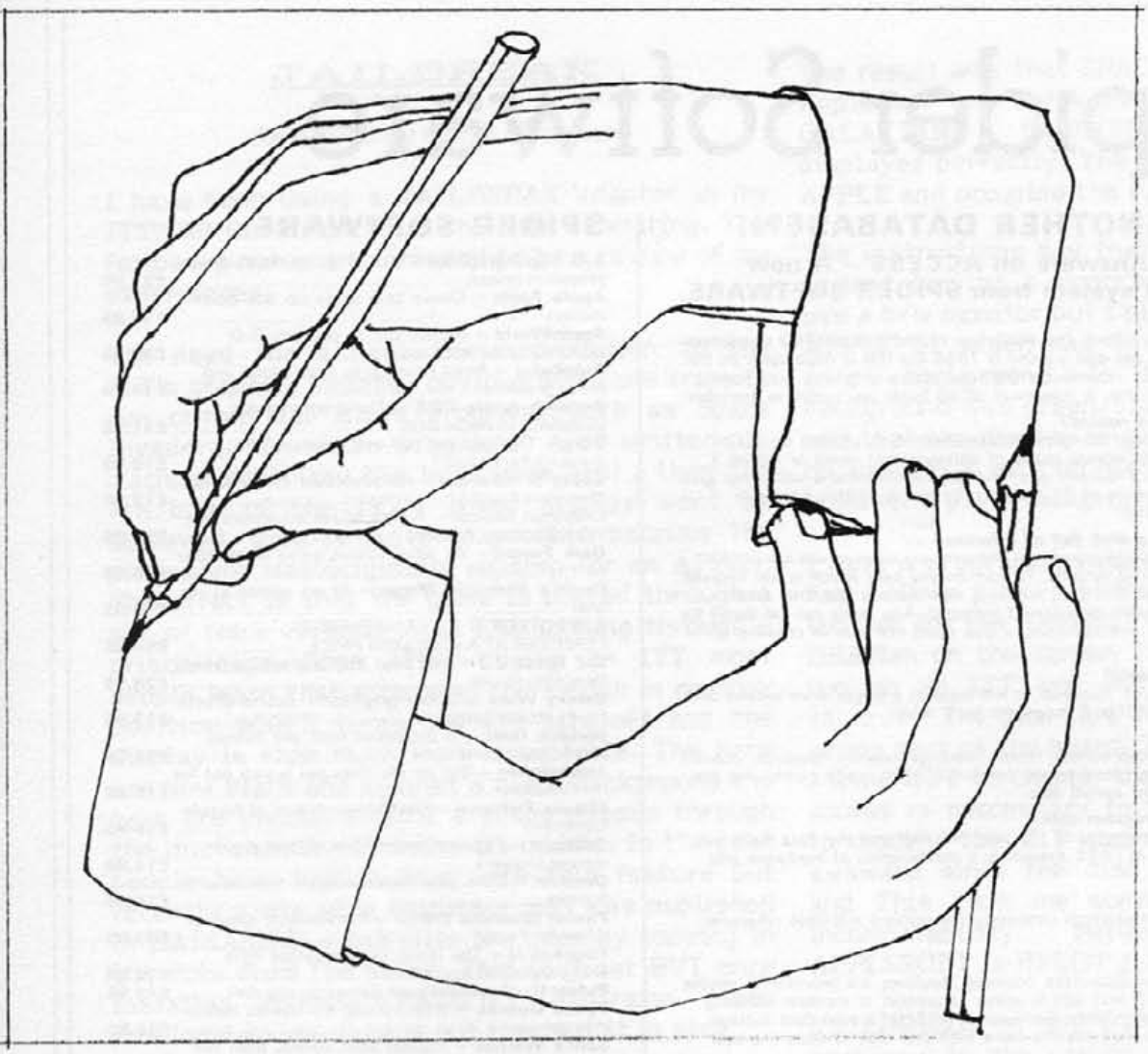
Prices include VAT at 15% Add 50p p&p for orders under £30 totally.

Please write or telephone for your free copy of our up-to-date price list or more detailed information on any product.

DEALER INQUIRIES INVITED PERSONAL CALLERS BY APPOINTMENT ONLY PLEASE

Spider Software

98, AVONDALE ROAD, SOUTH CROYDON, SURREY.
Telephone: 01-680 0267 (24 hours a day — 7 days a week)



THE

NEXT

ONE

The Scientific program generator

THE NEXT ONE is a program which takes the drudgery out of program development. It eliminates the need for the the programmer to write all the tedious I/O subroutines between keyboard/disc, disc/core memory.
* The target (or generated) program is automatically structured and interactive with many diagnostic features built in.
* The target code is ready for use within a few minutes of creation by THE NEXT ONE.

THE NEXT ONE will write to a file on disc (named by the user) the program statements which comprise the program to be created.
After the file name has been chosen the data base array name is requested. This array is used for disc Input/Output.
Next; command, variable or array names are requested (up to 100 may be defined for the target program).
THE NEXT ONE goes on immediately to write code for the default commands and housekeeping subroutines for data handling.

Also available is a large range
of Structural and Engineering Software
Dealer Enquiries Welcome

£50 + VAT (15%)

Tel. Southampton 583857

THE NEXT ONE

Logical Computing,
26, Wide Lane,
Swaythling,
SOUTHAMPTON.
SO2 2HH.

Book Review

USING 6502 ASSEMBLY LANGUAGE

Reviewed by Ian Trackman

My immediate reaction on seeing "Using 6502 Assembly Language or How Anyone Can Program The Apple II" (referred to from now on as "HACPTA" !) by Randy Hyde (Datamost) was one of shock at the price - £11.95 (converted from $19.95) and so, from the start, the book had to convince me that it justified the money. It is perhaps worth reflecting that we expect good software to cost anywhere between £20 and £100 and the effort in producing it is comparable to that of writing a book. Perhaps it is because I still remember 5/- Penguins ....

HACPTA is approximately 250 pages long and consists of 17 Chapters and an Appendix. It starts with a gentle introduction to the architecture of the 6502 and the principles of binary and hexadecimal notation. It continues with a detailed explanation of all of the processors' instruction codes, moving on to mathematical and string handling operations and ends with short chapters on Sweet 16, Input / Output and Debugging.

The book is a thorough study of its subject and, as a result, requires careful and close reading. Regrettably, Mr. Hyde is no Frederick Forsythe and his writing style makes no concession to anything less than total concentration. This means that you must be prepared to work hard for Mr Hyde but, in return, the end result will be rewarding. It is true that HACPTA contains a number of so-called cartoons, purportedly illustrating the text, but I found them decidedly unfunny and, in at least two cases, completely misleading.

Mr Hyde's work is accurate, although his printers might have been a bit more careful in their type-setting. I did not find any factual errors although I might have quibbled with some of the statements concerning hardware and about programming style.

There is one major criticsm to be made of HACTPA. Randy Hyde is the author of the Assembler known as LISA and several associated assembly language utility programs. Assemblers use what are known as "pseudo-opcodes" or "assembler instructions" which, unlike the actual 6502 instruction codes, vary from assembler to assembler, there being no accepted industry standard. These pseudo-opcodes are used, for instance, for data definition. Right from the start, HACPTA is used as a blatant advertising medium for LISA, which doesn't improve the book's readability. However, what is worse is that many of the examples are related specifically to LISA's pseudo-opcodes without a general explanation of how other assemblers deal with the problem and with the implication that, in any event, only LISA can cope - which is wrong.

Whilst HACPTA is meant to relate directly to the Apple, much of the book deals, as it of course needs to, with the programming of the 6502 processor in general. It would have spoilt the logical step-by-step approach of the chapters but, nevertheless, it might have made for lighter reading to have introduced the Apple's input and output routines early on. In that way, the reader could be given working programs which produced actual results even in the opening chapters. My own suggestion is that the reader should study pages 61 - 63 of the Apple Reference Manual (reproduced in HACPTA's Appendix) and use some of the Apple Monitor's sub-routines to experiment with the author's example programs, particularly in the beginning sections of the book.

The Chapter on Sweet 16 is stated to be "an introduction". A deeper look at the topic would have been welcome, including some suggestions as to the kind of situations in which Sweet 16 could be used to advantage.

Mr Hyde tells us that he intends following HACPTA with further volumes on advanced subjects and these are eagerly awaited by me.

Personally, I would have liked more example programs and in a lighter vein (some low resolution graphics or sound effects ?) since I suspect that most of Mr Hyde's readers will be hobbyists out to increase the enjoyment that they get from their computers.

Well, to answer my opening question, is HACPTA worth its price ? Indeed it is. This is a book that really will give you a thorough grounding in assembly language programming, providing you put in the effort.

STRUCTURED BASIC AND BEYOND

reviewed by Michael Tickle

I wish I'd had this book two years ago at the start of my struggle to learn Basiic programming. Unfortunately, it's not as simple as that.

Wayne Amsbury writes clearly, concisely, with obvious enjoyment on the way to program effectively and, by default, shows me all the bad habits I have got into. The Basic used is non-specific to any machine, but variations are discussed where appropriate. The first third of the book describes Basic with examples using flowcharts, listings, and more importantly Pseudo Code. The rest of the book uses mainly Psuedo Code to explain the more complicated algorithms and is none the worse for that. The Psuedo Code as defined, forces the programmer to think clearly about what the program is required to do, and at the same time imposes a logical skeletal structure on the Basic Coding.

Some of the chapter headings give an idea of the range of algorithms covered eg. Loops and Structures, Files, Strings and linked lists, Stacks and Queues, tree like Structures. Each chapter has self review, problems to solve and projects before moving onto the next chapter. Answers to the self review segment are at the end of the book.

Structured Basic and Beyond is aimed at University students and lecturers, to help produce professional programmers. But it is invaluable for 'do-it-yourselfers' and is the best book on programming without a doubt. I must alter my existing programs, if only to make them more understandable. If, of course, I had the time.

REVIEW

by Chester Kemp

AstroApple

Author: Bob Male
Publisher: Quality Software

AstroApple is, as the name suggests, an astrology package. It offers the following facilities: a horoscope of birth, a thirty day forecast, and compatibility ratings with other people. There are two programs, each of about 12k, the first one doing calculations for a number of people and forecasts, whilst the second gives interpretations for horoscope, forecast and compatibility.

We are told that the author is "a former teacher of astrology with over 10 years of programming experience" - well let us look to the results of this education; I look at this with the eyes of 18 years astrology and a decade of computing experience! (Stated for the record, but in itself not necessarily a recommendation of either myself or of this package's author).

Astrologically, this calculates the celestial positions for the birthdate, and uses a "house system" called Morinus, and a predictive system called Tertiary progressions. The house system is rarely used by astrologers and is somewhat eccentrically based, but the predictive system seems soundly based - although the mathematics only give the planetary positions at the outset of the forecast month (which is not the basis for tertiary progressions). The horoscope positions can be listed, but it is not given the usual astrological representation. The interpretations are trite and simplistic, and as there is little variety, the entertainment value is low. He gives compatibility ratings on a scale of 0-10 for physical, mental and emotional - the astrological justification is slight, and the result is ridiculous!

The elegance of the package is barely noticeable; I found it irritating to have to keep loading in another program to go back and get another chart done. The programming was no more than adequate, and the resultant displays were uninspired, to say the least.

Well, in terms of astrology, it is neither representative nor well done. In terms of entertainment value, it may temporarily (half an hour?) keep you occupied, but after that, forget it.

reader's letters

Killarney Heights
Australia

Dear Friends,
... I have just been elected Librarian for the Sydney Apple User Group and as such I have to search for new software in the public domain area for inclusion in the club library.
It occurs to me that we could exchange software to our mutual benefit. We would give credit to the originators.
Whilst on this subject, can I please beg you to refrain from sticking BASUG in the midst of programs without verifying that the program isn't fouled up as a result. May I also request that you keep disk catalogs simple and unadulterated. My printer goes berserk when I try to list some of the fancy inverse video catalog headings which are currently being distributed.
Thanking you for the continued excellence of your magazine and in anticipation of your participation.

Yours sincerely

Don Riley

Yes, Don we would be interested in exchange.
I hope you noticed the apology in the magazine last year, about the insertion of BASUG in a batch of introductory disks. This was a long time ago, and will not happen again.
On the inverse video side I complained about this to the person concerned. He now knows it isn't just me. I think sometimes one can do too many fancy things, which not everyone can cope with.

Bury St Edmunds

John,
I am an American living in England. I have been an APPLE owner for a long time and so have a large collection of software. I would like to contact BASUG members close to me. Would they please phone me on 0284 828666.

Bob Hirst

Paignton

Dear Sirs,

My main problem is with the ITT2020, as I have bought some games which are not compatible - the ITT literature does not make this clear.
Some of the games require paddles or joysticks and I am not quite sure which kinds are suitable. It is necessary to have <u>both</u> paddles <u>and</u> joysticks? I would also be very interested to know of any of your members with ITT's who might live in this area, if this is possible- or form a local group.

Yours sincerely

A A Bassett

The contact between BASUG members is an important aspect of the group. We decided in the early stages not to release a membership list. This was for two reasons. There were, when we held the inaugural meeting, lots of microcomputer thefts. We felt as a meeting that lists should not be sent out as a matter of policy, so that the criminal fraternity would know just where to go. The other factor is many people do not like their name being on a multitude of mailing lists. This is a subject which needs to be brought up at the AGM, but if you are willing to have your details published as Bob Hirst was we will do so. Mr Bassett did not give permission so we have not added the full details.

xxxxx

Keighley

Dear Sirs,

Congratulations on a magazine which is a credit to all involved. I was very interested to read the article by Neil McFerran in the August 81 issue. Is there any chance that you might publish a summary of errors of various types found by users, preferably with a note on how to correct them? For example, in the article (p 11) Mr McFerran refers to another "Append" error which I would like to know about.
Keep up the good work!

Yours faithfully,
A R Woolhouse

The APPEND error refered to was covered by Neil McFerran. It exists separately as an APNOTE. Write to John Rodger for the Library List (don't forget the SAE).

xxxxx

Sassenheim
Holland

Dear Basug,
I was interested in Mike Norfolk's letter (BASUG Dec 81) about using the Integer Card in another slot on his ITT. I tried this technique successfully on my Apple by putting my Applesoft card in Slot 2. I then booted DOS using the Disk Master and after getting into Monitor changed the following memory location - A5B8:A0;A5C0:A1. Any disk now INITed will run Applesoft from Slot 2. Having done thus can I now ask BASUG what advantages the configuration of Language Card in Slot 0 and Applesoft Card in Slot 2 gives me?

Yours faithfully
Alan Weinberg

The answer to you problem is that any program, or for example PASCAL, expecting to find or use a language card uses the standard slot 0. So it is best to have your language card there. If you have an APPLESOFT (or INTEGER) CARD as well then it will need to go into another slot, otherwise you will need to keep changing them, which is a problem unless you have a RAMEX or DIGITEK card, which do not have flying leads. Now also most printers use slot 1, so the best place to have your APPLEOFT/INTEGER card is slot 2.

Muir of Ord

Dear Sirs,
I have been trying unsuccessfully to scroll the HI-RES pages using machine code. The main problem is discovering the 'magic' relationship between machine code and the rather odd memory mapping of the screens.
Of course, I know the mathematical relationships that convert X,Y coordinates to the appropriate screen addresses and have produced these conversions in assembly language but these routines are exceedingly lengthy.

I was wondering, therefore, if any of the members could assist me with this problem. It could make a good article for Hard Core, particularly as I have never seen any articles explaining why the apple uses such strange screen mapping!

Yours faithfully,
A.J.Jameson

The Hi Res scrolling routines were dealt with in a recent issue of CALL A.P.P.L.E..
The reason for the odd sequence of numbering is to make it easier for calculating when you are in machine code. But this doesn't stop someone writing up a version for BASUG. There were a number in the magazines in 'the old days' about two years ago, but it would be useful to see the explanation again, amplified if necessary for the new wave of owners. Any offers?

xxxxx

Manchester

Dear Sir,
When I entered the machine language TAG SORT from NIBBLE vol 2 no 6, into my ITT 2020, attempts to run the sort produced a hung system. This is because the PALSOFT routine equivalent to the APPLESOFT "GETARYPT" resides at $F7C5 not at $F7D9. With this change the sort works perfectly on the ITT.
Hope the information will be of use.
Yours faithfully,
Jennifer Gamston

xxxxx

Raynes Park

Dear John,
Here's my £10! I'd like to propose that all national and local committee members are granted free membership with no meeting charge; not so much because of the cost, but quite simply the aggarvation gets in the way of more important things.
Cheers,
Warren Avery

We have thought about it, and decided some time ago that even committee members would pay just the same as everyone else. At this stage we feel it would be best left for the AGM to decide. It would perhaps need to be retrospective for work done rather than work to be done. It often doesn't materialise as it should. Others have brought up the same idea so there should be plenty of support. For your local meetings you can have whatever rules you want, but remember everyone will want to 'help' so they don't pay and you still have to fund the meeting.
John Sharp

xxxxx

Derby

Dear John,
With reference to the compiler review (vol 1 no 6):-
The RUN TIME program of TASC can be BLOADED together with the compiled program if they are both bloaded and saved together. However, the statement 4C B0 17 should be added to the beginning of RUNTIME. The address $17B0 is the starting address of the default values of the compiled program. Hence the final machine code program can be BRUN directly.
Yours,
Dieter Yih

xxxxx

A PLEA AND NOTE

If you have sent a letter and not got a reply, it was printed here because NO SAE was sent with it. If you have sent a SAE then it will gradually get done, although I am fairly up to date on those.
Any letters sent other than to the PO BOX, e.g MICROSOURCE are likely to be severely delayed and or lost in the process. Likewise any sent personally to the committee at their homes. Even more so in the latter case.

High Wycombe

Dear Sir,
The following tip may be useful for those members who have a language card and load INTEGER BASIC up when they boot the disk. On re-booting a slave disk they will then find the INTEGER has been lost. It is a modification of the greetings program.

```
First BLOAD MASTER CREATE  <CR>
      CALL -151            <CR>
      *080D: 4C            <CR>
      *800G                <CR>
      *1BCD: 20            <CR>
      *82DG                <CR>
```

This is not my idea but comes from Dec Issue of SOFTALK.

Yours faithfully,
B P Walker

xxxxxx

London SW16

Dear David,
I refer to 'MICROSOFT TASC COMPILER, A REVIEW' in the February issue of HARDCORE.
A drop out has occurred in the paragraph following the small table towards the top of the right hand column on p 33. The paragraph should have read:-

"However, LISTING 4 is not in optimum form for TASC which works faster if variables and arrays are, so far as is possible, declared as Integers. If a mixture of integer and FP values is required, the integers have to be decalred individuallyl but for LISTING 4 a blanket declaration can be made in the form:- "

I would also add that in the last full paragraph on P 33, about the middle 'was' should be replaced with 'uses';
'....small program which uses. a relatively large array.'
Yours sincerely,
Peter Broadbent

Sydney, Crewe

Dear David,

The emergence of APPLE SPIEL into the software library coincided with my completion of APPLE SPIEL IC. As a result of the experience gained in producing the new version, I should like to make a few comments about the versions in the library. The original BASIC version should be easily modifiable for any set-up, and the following remarks all refer to the hybrid version.

First, if anyone has the disc controller in a slot other than 6, line 9010 of the BASIC part of the program must be removed. The same is true if there is any card which is always active (e.g. a clock card) or which expects its scratchpad to be preserved while it is inactive. The offending line causes ALL the scratchpads to be over-written. Second, the hybrid version will not work with any card which uses the scratchpad in a different manner from the Apple card or which expects the high bit set on each character.

APPLE SPIEL IC, for which I am hoping to get a commercial release, has double the capacity and many more facilities. It comes complete with a configuration program which should enable it to work with any system, including lower case character generators, unless somebody has an interface card which uses areas outside its own ROM and scratchpad. I hesitate about offering a copy for review, as it would give Tony Williams sleepless nights (no search facility).

On a different topic, I see I can only access the library if I quote my membership number. Memory has never been among my strongest attributes, but I really cannot recall ever having been given one.

Yours sincerely,

Neil Lomas.

New Romney

Dear Sirs,

I am particularly interested to know if there is a good 2-pass assembler on tape for a 48k Apple with tape only. Also I purchased a Mountain Hardware Rom Writer only to find the program to run it is on disk. Is this available on tape? A 6502 development system or assembler in hardware would also be of interest. I hope you can can help in the above points.

Yours sincerely,

P.S. Blatchford

There doesn't appear to be an assembler which works just for tape. There was a book reviewed in a recent NIBBLE which had a free Tape assembler. However, as is usual in these cases, I can't find it. Has anyone else any ideas?

Scunthorpe

Dear Sirs,
At College we have a PAPER TIGER 440 for our printouts. Using the disk supplied we can transfer HI-RES pictures to the printer but not LO-RES. Is there a program in the library which can do this if so which?
If not please supply a solution to the problem.

Yours
Tony Gravil

No there isn't a program in the library.The way to solve the problem is to define a 16 colour shape table, which matches the LO-RES screen and using the SCRN function copy across. Then the HI-RES page can be dumped. The details of how to make a 16 colour block shape table is relatively easy IF YOU KNOW HOW, but requires a full article to show how and why it is done. I did ask someone to do it who wanted something to do but no results have been forthcoming.
John Sharp

xxxxx

London N.W.3

Dear David,

I'm putting cursor to VDU (well, "pen to paper" isn't quite accurate) to send you a miscellany of thoughts arising out of the last edition of Hard Core.

Firstly, may I congratulate you and all others involved on continuing to maintain the excellence of the magazine. As Henry Root would say - "Keep it up !"

In response to Chester Kemp's review of Super Editor, I have decided to meet his criticism by reducing the price of Super Editor, Super Trace and Packing Suite from £27.50 to £22.50.

FID doesn't copy random-access files properly because of the "null data" bytes. Before I try to make it work, has any BASUG member already solved the problem ?

Les Bugden claimed a first for information on ITT graphics entry points. May I modestly remind you that you published my article on the same topic in October 1981 ?

And whilst on the subject of back numbers, would authors please give references when they mention other articles ?

Finally, a request that you should always publish the source code of any assembly language program appearing in the magazine. With only a hex dump, it is impossible to re-create the author's labels and comments and, therefore, his original design. Entering code through an assembler also traps typing (and publishing) errors. And reading source code is so much more interesting than wading through hex bytes !

Kind regards,

IAN TRACKMAN

Cambridge

Dear BASUG,
How can I save all the CPU registers including the program counter in User RAM and then reload the CPU.

K Pandey

There is a subroutine in the APPLE monitor at $FF4A which saves the 6502 registers, and a further routine at $FF3F which restores these registers. The save uses zero page locations $45-$49. If you wish to save the program pointers we suggest you write your own routine in machine code based on the above routines- they are very simple and very short. The best books for the details of the locations to save and where it is safe to save them are "THE APLE MONITOR PEELED" and "WHAT'S WHERE IN THE APPLE".

xxxxx

LONDON NW11

Dear Sir,
We was interested in the article on ITT 9th bit addressing in the issue of December last year. However, although the suggestions about changing bit 9 appeared to work (i.e had a perceptible effect on the graphics screen) the suggestions for examining bit 9 did not. THere were some obvious mis-prints in that part of the article, and I wonder if there were some less apparent bugs.
Any help on this, and any help generally as to how to find out how the ITT graphics work, would be gratefully appreciated.

Yours faithfully,
Mrs S P Skemp

I'm afraid this is one I will have to ask Ken Gordon and Ian Trackman to answer.

xxxxx

London W11

Dear John,
I am writing a tailor made stock control program for a company whose employees have absolutely no experience of computers, and are therefore liable to press any key, including the RESET key. I have put as many validation routines as I can think of to ensure that only correct data is input, but I have yet to discover how to disable the RESET KEY.
Please can someone tell me how this can be done?

J.P.Yong

The first solution is to make the RESET key inactive, by setting the switch inside the APPLE (underneath, and to the left of the keyboard). This means then that it is necessary to press the CTRL and RESET keys at the same time in order to reset the Apple.
The second solution is to fix a washer or rubber band under the key so that it is very hard to press. This was the standard fix before the switch was introduced.
Then to be more sophisticated, you can alter the RESET VECTORS. This is mentioned in the APPLE REFERENCE MANUAL. By poking into two locations you can make the program start a machine code program by jumping to these locations. Your obvious choice would be to RUN the program. This resets the variables, so something more sophisticated would be needed in this case, which no doubt now they have been prompted someone will come back with the answer. At the moment I have to finish typing the letters. In order to make it RUN do the following POKES at the start of the program:-
POKE 1011,102: poke 1012, 213
Then it wil reset to the RUN command in APPLESOFT.

John Sharp

xxxxx

Lytham St Annes

Dear Sirs,
Please send me details of BASUG.
My main interest is in using a WEGO Mark sense card reader to mark objective homeworks, but can you tell me of anyone who is able to help with lower case subscripts etc on printers e.g

ClO_4^{2-}

This would be particularly useful as I am a chemist.

Yours faithfully
Dr J.A.Gudgeon

As you can see it is possible. This was written with APPLEWRITER and Ian Trackman's Go-Between for the CENTRONICS 73 -available from your local APPLE dealer.

xxxxx

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"

Time Machine II™

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 are 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.

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: ± .00%.
- 24-hour military format or 12-hour with AM/PM format.
- Six date formats and seven time formats including DD-MMM-19YY and MCI Apple Clock formats.
- Four software selectable interrupts (1/1024, 1, 60 and 3600 seconds).

CONTROL ROM
- 2K driver executes in the $C:N00 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.

PROTOCOL COMPUTER PRODUCTS

Apartment 3
114 Widmore Road
Bromley
Kent BR1 3BE
Telephone: 01-460 2580

5¼" DISCS BASF

PERFECT FOR APPLE II AND APPLE III

1 box (10 discs) **£16** per box
10+ boxes **£15** per box
100+ boxes **£14** per box

Post to: **SYSTEMICS LTD., 21-23 The Bridge, Harrow, Middlesex HA3 5AG.**

Order form Please send me:

Qnty.	Phone orders: 01-863 0079	Price	VAT	TOTAL
	Boxes of 10 5¼" DISCS			
	add £1.00 Delivery			£1.00
			VALUE OF ORDER £	

NAME _____

ADDRESS _____

_____

_____

Signed _____ Phone _____

My Access/Barclaycard number is:

BARCLAYCARD VISA

TREE SORT

TREE SORT is a high speed utility to allow you to add fast sorting to your programs from data in APPLESOFT arrays.

FAST means 1000 items in 5 seconds
1000 words in 8 seconds
SORTING can be Ascending or Descending
On whole or part of an array
Rewritten or stored as sorted in another array.
Zero values may be included or excluded.
Can be on single or multiple dimension arrays
PLUS MORE refinements

COMES WITH EXTENSIVE DOCUMENTATION DEMONSTRATION PROGRAMS complete with explanation in the manual.

TREE SORT costs £22 including P & P

"THE POWER OF VISICALC"

VISICALC is one of the most exciting developments in computing. The manual is thick and meaty, but when you have worked through it and mastered the use of this new language, you have a powerful tool at your disposal.

Now with THE POWER OF VISICALC, there is a book which takes you through a number of exercises on various aspects of VISICALC, which allows you to follow the logic of VISICALC functions, and apply them to specific problem solving situations.
You may find uses you had not thought of, or find out how to use parts of VISICALC you didn't know existed.

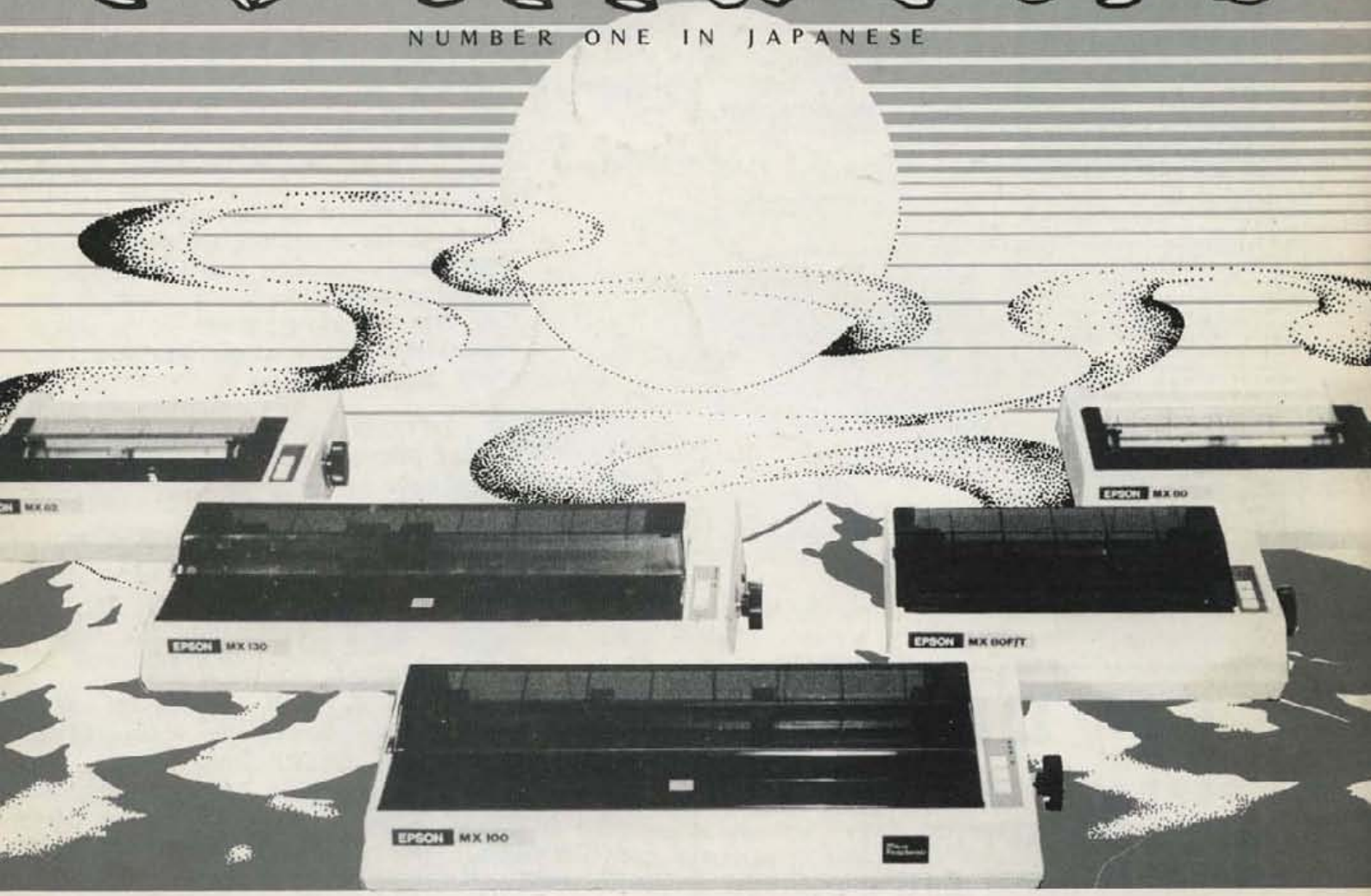
All exercises are step by step with explanations as well as printouts of the screen as you should see it.

THE POWER OF VISICALC costs £8.50 incl post & packing FROM

MICROSOURCE
1, BRANCH ROAD, PARK STREET, ST.ALBANS.
TEL: 0727 72917 & 09273 75093

ICHIBAN

NUMBER ONE IN JAPANESE



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

	Mutrix			*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"*	*15⅛"*	
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.