

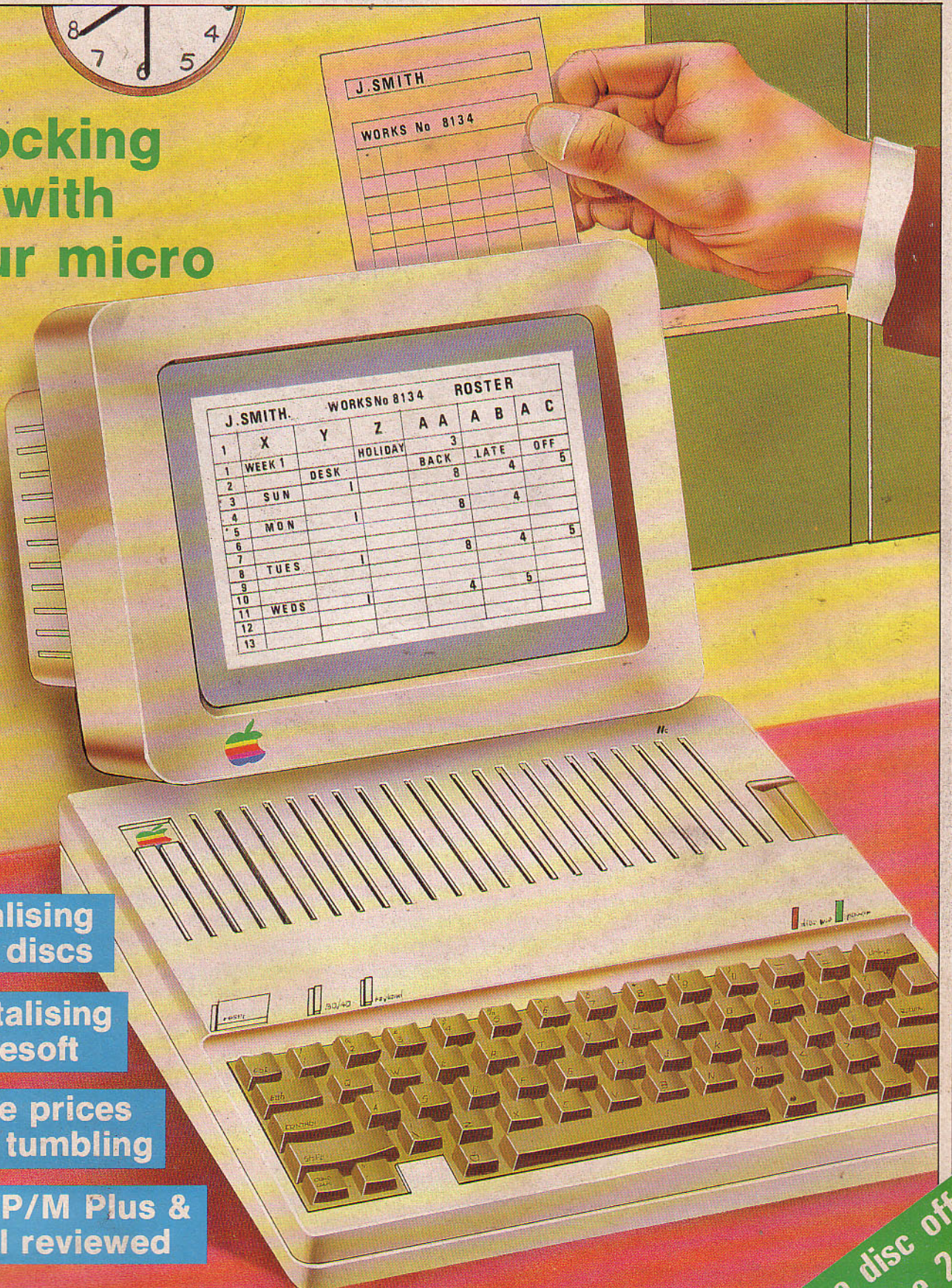


A Database Publication

apple user

Vol. 5 No. 11 November 1985 £1

Clocking on with your micro



Initialising data discs

Revitalising Applesoft

Apple prices start tumbling

IIc CP/M Plus & Excel reviewed

Free disc offer:
Page 26

News

● New drives and memory card for the II+ and IIe, new price for the IIc bundle, and more.

5

MicroLink

● The latest update from Britain's national electronic mail service.

9

Skyfox offer

● This month we've got a special offer on Skyfox, the great 3D flight and fight simulation.

15

MEMDOS

● Jon Parry enthuses over the operating system that's transformed his database programs.

17

ProDOS

● Chris Payne explains how ProDOS gives Applesoft a new lease of life.

21

Spreadsheet

● If it's your job to work out duty rosters, Bill Hill's suggestions could save you a lot of brain-ache.

25

Disc offer

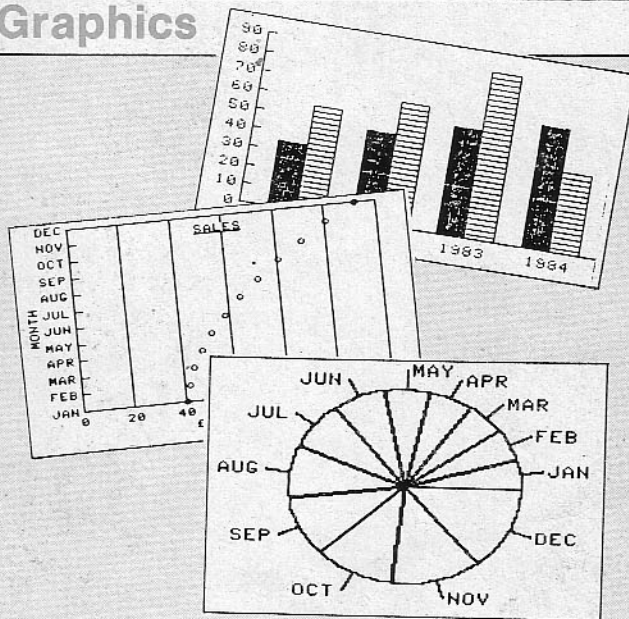
● A free disc is yours when you subscribe to Apple User.

26



Volume 5
Number 11
November 1985

Graphics



● It's the end of the road for the Apple User Graphics Library. Full details of how you can now get the complete series on disc.

10

Macintosh

● Microsoft's Excel gets the Bill Hill seal of approval.

28

● Read how Apple User's Mac got the upgrade treatment.

29

Fun & Games

● Infocom's Suspect and Broderbund's Karateka and Dazzle Draw provide this month's entertainment.

31

Review

● The Cirtech CP/M Plus system gives the Apple IIc access to an even larger software base.

34

Comparison

● John MacGibbon compares some Apple word processors with their MS-DOS competitors.

37

Pascal

● Stuart Bell continues the Pascal Tutorial series with a close look at parameter passing.

38

&DOSFile

● In Part II of the series Peter Harris provides two more important routines.

42

New Products

● Two pages of money-spending ideas to enhance your hardware and software.

47

Utility

● If you've got a new half-height disc drive, you might not be using it to capacity. Chee Wei-Li will show you how.

54

Classifieds

● Bargain offers from your fellow Apple users.

56

Feedback

● Sideways Appleworks spreadsheets, Apple III software plea, problems with E-Z Draw and more.

58

Order form

● Order everything on this one handy form.

60

Apple User Managing Editor **Derek Meakin** Features Editor **Cliff McKnight** Technical Editor **Max Parrott**
 Production Editor **Peter Glover** Advertisement Manager **John Riding** Sales **Chris Riding**
 Design **Heather Sheldrick** News Editor **Mike Cowley** Editor-in-Chief Database Publications **Peter Brameld**

Telephone: 061-456 8383 (Editorial) 061-456 8500 (Advertising) 061-480 0171 (Subscriptions)

Telex: 265871 MONREFG Quoting Ref. 79:MAG001 Telecom Gold Mailbox: 79:MAG001 Prestel Mailbox: 614568383

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

News trade distribution: Europress Sales and Distribution Limited, 11 Brighton Road, Crawley, West Sussex RH10 6AF. Circulation 0293 27053.

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

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

© 1985 Database Publications Ltd. No material may be reproduced in whole or in part without written permission. While every care is taken, the publishers cannot be held legally responsible for any errors in articles or listings.

ABC 12,850 - JULY - DECEMBER 1984

Subscription rates for 12 issues, post free:

£12 UK
 £13 Eire (IR £16)
 £20 Europe
 £15 Rest of world (surface)
 £30 Rest of world (airmail)

New products boost for Apple II

GREATER flexibility and performance for the Apple II range are promised by new hardware and software to be released by Apple UK between now and January.

The new peripherals include UniDisk 3.5, a 3½in floppy disc drive that can store up to 800k of information – more than five times the capacity of the Apple's current 5¼in drives.

And an Apple II memory expansion card now being developed expands the machine's maximum internal memory to more than 1mbyte – more than an eightfold increase over the current 128k standard.

Also due out shortly are a

mouse-based program selector and desk top organiser called MouseDesk, and ColourMonitor IIc – a high resolution colour monitor that can double as a monochrome monitor for 80 column text.

The new products, compatible with many existing programs, are also encouraging developers to create a wide variety of new enhanced applications for Apple II computers,

the company claims.

"They deliver the increased storage and memory capacities that our users and software developers have been asking for", said David Hancock, Apple UK managing director.

"The new Apple II drive and memory card demonstrate the commitment to our strategy of continually adding value to the range and, as a consequence, we expect the Apple II family to remain a viable product line into 1986 and beyond".

Designed for Apple II, II+ and IIe users, the 800k disc drive bridges the storage and speed gap between Apple's low end 143k floppy disc drives and 10mbyte Profile hard disc.

The mass storage capability of the UniDisk 3.5 also makes it a cost effective backup solution for hard disc users.

The memory expansion card is designed automatically to accept data or programs as soon as the Apple II, II+ or IIe is turned on.

The extra memory can be used as an internal disc drive, allowing programs to be directly

loaded and executed in the computer's internal memory, dramatically increasing the speed at which programs run.

With revised software the extra memory can also be used for creating larger documents or more small files.

MouseDesk, developed by Version Soft of Paris, is a powerful utility program and operating system that eliminates the need to remember cryptic computer commands and allows the user to easily move, copy, rename and delete files on the disc visually using the mouse.

As a program selector MouseDesk allows the user to load various programs into the

Turn to Page 6



This IIc system plus panel display monitor now costs £1,395

Aussie export drive

THE Australian Trade Commission in London is running a major export drive for their country's computer software publishers directed toward Britain and Ireland.

In the forefront of the campaign is Spinifex Software, a company which designs educational programs for the Apple II series.

The first series of Spinifex packages focuses on basic mathematical problems, and the Discovery program allows pupils to experiment with elemental number and maths concepts.

Associated tutorial programs have also been developed, aimed at encouraging specific mathematical skills such as long multiplication and long division.

MAJOR PRICE CUTS

APPLE has launched an aggressive new pricing strategy in a bid to retain its pre-eminent position in the forefront of the personal computer market.

Moves include dramatic cuts in the cost of Apple IIc systems and both the 512k and 128k Macintosh, as well as the Laser Writer printer.

The £1,000 price barrier of a complete Apple IIc system is broken for the first time with a special offer that brings the cost of the computer and accessories down from £1,700 to £995.

The full system offer com-

prises the Apple IIc with 128k RAM and a built-in disc drive, matching monitor and stand, external disc drive, Mouse IIc and MousePaint graphics program, and the best-selling AppleWorks integrated business package.

Included in the offer are two important extras – a colour modulator and a carry case.

Executives on the move who want a fully portable system are being offered all the above items plus the Apple IIc liquid crystal flat panel display for £1,395.

The Macintosh 512k drops

from £2,595 to £1,995 and the Macintosh 128k from £1,795 to £1,695.

The price of the Laser Writer printer is reduced by £1,000 to £5,995.

Apple UK managing director David Hancock says: "These new prices must be welcomed by a whole variety of businesses, but especially small companies seeking a low entry price for a powerful business computer."

"The Macintosh 512k at under £2,000, for instance, represents an unbeatable computer power-for-price ratio".

Anger as Jobs goes

AS forecast in last month's *Apple User*, co-founder and biggest shareholder Steve Jobs has severed his connection with the management of Apple Computer.

The split was acrimonious, with Jobs reportedly accusing his fellow directors of "adopting a hostile posture . . . contrary to Apple's best interests" and leaking statements that were "misleading to the public and unfair to me".

With Jobs have gone five senior Apple employees, recruited to help him in his new venture – a company called Next Inc that will produce hardware and software for the educational market.

Although Jobs has said: "We just want to build our new company and invent something new", his former colleagues on the board have a different view of Next Inc and are taking action "to assure protection of Apple's interests".

Reports from the US say Apple has filed a \$5 million lawsuit against Jobs alleging that his new company will be using technology on whose development Apple has spent "thousands of man years and millions of dollars".

It would mean Jobs being sued by the company in which he remains the largest single shareholder.

Apple wins Mac copyright battle

APPLE has won a major battle over copyright without a single shot having been fired in court.

It has accepted an undisclosed sum from Digital Research and the assurance that the company will make immediate changes to its Gem programs.

The dispute arose after Digital Research started to ship its Gem operating system, which recreates a Macintosh type environment, to other computer manufacturers.

Although Digital has refused to concede that it has violated Apple's copyright in any way,

the corporation has now signed a document stating:

- It will now produce new versions of Gem designed to be "substantially different to Apple's Macintosh personal computer in both screen appearance and operation".

- All changes are to be reviewed by Apple to ensure they satisfy the requirement that Gem programs no longer appear to be substantially similar to those of Apple.

- The company has agreed to disclaim any Gem program compatibility with Apple products and to point out that

the two companies' programs do not perform in the same manner.

- All Digital Research's advertising will be amended so as not to engage in comparisons with Apple products.

The immediate action to be undertaken as the result of the agreement is that Digital Research must modify three specific programs – Gem Desktop, Gem Paint and Gem Draw.

It has also been decided that as the new versions become available all the old program stock will be withdrawn from the market.



Stephen and Susan Alsop

Business booms

A FORMER contributor to *Apple User* in the days when the magazine was called *Windfall*, Stephen Alsop, has come a long way since he and his wife Susan used to design computer systems on their kitchen table.

The husband and wife team are directors of DMS Electronics – designer, manufacturer and supplier of computer software and electronic systems – which recently moved to new £250,000 headquarters in Sheffield.

The offices, workshops,

showroom and warehousing on a 1.3 acre site in Wales Village could create 24 jobs over the next three years.

It is a far cry from 1973 when the Alsops set up the company and had to use their kitchen, living room, bedroom and attic as workshops to assemble their product range for commercial, industrial and educational markets.

Thanks to a £44,000 grant from the Department of Trade and Industry and a £100,000 EEC loan, the move to the new custom-built building in Wales Village was made possible.

Projects that DMS has completed in the past include an ultrasonic flaw detection unit for the steel industry, a spark erosion machine for the tooling industry, oilrig stress analysis for firms working in the North Sea, beer profile testing and a betting system for a greyhound stadium.

Other developments include a weather data logging system, a computer telemetry linkup for oil companies and a spectrophotometry system for a glass manufacturer.

Boost for the Apple II

From Page 5

memory expansion card, the UniDisk 3.5 or another mass storage device and, with simple mouse movements, switch back and forth between programs.

ProDOS software can also use the expansion card as a RAM disc without modification, but requires changes to use the additional memory for storing larger files.

DOS 3.3 programs can use the expansion card as a RAM disc, but need modification for expanded file storage to be achieved, and they will not work with UniDisk 3.5.

Programs written in Pascal must be revised to recognise either UniDisk 3.5 for data or program storage, or the expansion card as a RAM disc or file storage device.

Apple says developers who

are using UniDisk 3.5 to expand the size of their programs or to group several programs together, intend to release their new products in the 3½ in format.

But a spokesman told *Apple User* that the company expects programs in the 5¼ in format to be available indefinitely either as separate products or with both disc formats packaged together.



Cardiff Micro managing director Phil Williams (left) with his firm's marathon runners and backroom boys in the control centre after the race.

Church moves into micros

THE recently-formed Christian Micro Users Association has begun its work of linking together computer users of various denominations and promoting the use of micros in church activities.

Association secretary Philip Clark told *Apple User*: "There is a great need to discover and make contact with the individuals and companies producing 'Christian software'.

"We also need to share the expertise and ideas of the many who have sought to use micros in their church-related activities".

The aim of CMUA is to concentrate on the popular

computers like the Apple II and it has already unearthed a significant number of programs based on Bible studies.

The Association intends to produce a quarterly magazine called Christian Micro and, from next January, launch a series of tape magazines for specific micros.

Its other activities will include the formation of local groups, arranging Christian micro exhibitions and conventions throughout the country, acting as a forum for the sharing of information about micros in the church, and promoting the highest quality Christian software.

MACPLOT UPGRADE

SOFTWARE developer Microspot has launched a professional version of its MacPlot program on the American market.

MacPlot Professional, which allows a Macintosh to be interfaced with a wide range of plotters to produce charts, drawings and documents in high definition colour up to A0 size, was revealed at the MacWorld exhibition in Boston.

The product is an enhancement of the Microspot MacPlot program which supports plotters up to A3.

Micro firm is up and running

MANAGEMENT and staff of major South Wales Apple dealership Cardiff Micro Computers are more used to running programs than races.

But they had important roles to play on the road and behind the scenes at the recent Cardiff marathon.

Using a Macintosh and a Tecmar storage device the firm provided a completely computerised management system for the event.

And several employees, led by managing director Phil Williams, still found enough energy in reserve to actually take part in the race – and complete it.

In advance of the event Cardiff Micro compiled a list of entries using a database program developed to hold details of every runner.

This was used to produce address labels for the information packs sent to entrants, and sorted lists of the runners for press and radio commentators.

Runners' times were recorded as they finished the event and the final printout was of the runners in finishing order for publication in the Western Mail newspaper.

"There was only one slight hiccup", Phil Williams told *Apple User*. "Some of the runners who had entered for the full marathon decided without telling anyone that they'd run the half marathon instead.

"And because they were registered in our database as running the full 26 miles their finishing times showed up on the printout as world records!

"Fortunately we spotted what was happening fairly quickly and were able to adapt our listings accordingly".

MELODY FOR 16 MACS

MACINTOSH computers called the tune when Apple UK managing director David Hancock opened the new computer services unit at the Dorset Institute of Higher Education.

Sixteen machines – the largest ensemble of computers ever used for a public music performance – played a 16th century composition by Thomas Tallis as a highlight of the inauguration ceremony.

But business computing is the serious aim of DIHE's new venture. The institute is expected to become a Polytechnic next year and will offer a comprehensive prospectus of business applications and programming tuition.

A total of 65 Macintosh computers has been purchased from Deverill Computers of Poole and these were used on opening day to demonstrate various software packages.

A mini graphics program to bring it all together

PETER GORRY concludes his epic series on Apple graphics

THIS month sees the end of the *Apple User Graphics Library*. At the beginning of the series I said that I wasn't going to produce a graphics program, but rather a set of routines for adding to your own programs. So here it is – a mini graphics program!

It's not that I've changed my mind about what the package is for, but it seems a nice way of illustrating how it can be used in a realistic context.

The program given here is not meant to rival the commercial packages, it's really just the bare bones of what a good graphics program should be. It will, however, produce bar charts, pie charts and point plotting diagrams with automatic ranges, scaling and annotation. It also provides simple data input and file-handling facilities in a menu-driven format.

The routines must be added to the existing Graphics Library since they make extensive use of almost all the other routines. You will almost certainly have to put DOS on the language card in order to make the program fit, since it is now very large. The program is also designed to start above hi-res page 1, so you must make sure it is placed there.

Figure I is a schematic layout of the program. The top line shows the first menu options. Selecting one of these either produces more menus or plots a graph. For instance, selecting the data option gives you a choice of inputting new data from the keyboard, listing the data, saving the data to disc, reading from disc or returning to the main program.

The simplest way of produc-

ing a graph is to go straight to the draw (plot) option. All decisions on layout, ranges, shading etc will be made for you. Having seen this graph it is then a simple matter to override the default options to produce the final result you want.

Control of the graph layout is accessed from the format menu. Here it is possible to change the ranges on both axes, the "tick mark" step size, label lengths and decimal places and titles. The graph can also be given a

border, horizontal or vertical grids, zero lines and a choice of annotation options.

It is here that one accesses the pie chart options as well. Once a pie chart has been chosen all graphs will be of that type until the default option is selected. This is done so that all data types – apart from multiple grouped histograms – can be displayed as pies if desired.

The various options are too numerous to go into detail here and I recommend a look at the detailed descriptions of the appropriate routines in the previous *Apple User* articles. In fact the program here does not offer all of the options that the routines support simply because the program would have become far too long.

Some of the graphics library routines are not even used at all, such as the function library, the line type and the cursor routine. Others are not used to their full

extent – such as the histogram routine, the string plotting routine and text type options.

It is a relatively simple matter to extend the menus to include whatever options you require, using the present routines as a guideline. These could include choosing the physical size of the graph, multiple graphs on the one page, "floating labels", plotting functions and printer dump option. This latter is especially simple if you have a card that only requires a couple of control characters to dump the hi-res screen.

For those of you who may want to produce suitable data files from within a program I have provided a schematic layout of the four data file types that the library routines support. These are shown in Figure II. The first line is common to all and is the number of data points (ZN) and the data type, signified by the letter; N (numbers only), S (symbols + numbers), H histogram.

The N data files are then followed by the X and Y values in pairs. In the program these will be placed into ZX() and ZY(). The S data files are followed by the labels and values of each point and are stored in ZPS() in the program.

The histogram, H, files are followed by the number of groups and the number of bars per group. These are read into ZH(1) and ZH(2) in the program. The label for each group and their values are then listed. The special case of ZH(1)=0 is shown in the last column. You should study the read and save routines for a more detailed description.

I hope the Graphics Library is as useful to you as it's been to me and that you can now produce flexible and professional graphics easily from within your own programs.

There are many more routines I could add but the free memory is rapidly disappearing and if I carried on there wouldn't be room for your own code in among the rest – so I'll stop here and promise that I'll never do such an epic series again.

ZN,ZD\$	ZN,ZD\$	ZN,ZD\$	ZN,ZD\$
X1,Y1	L1,Y1	L1	L1,Y1
X2,Y2	L2,Y2	Y1	L2,Y2
		Y2	L3,Y3
		L2	
		Y3	
		Y4	
		L3	
		Y5	
		Y6	
N	S	H	

Figure II

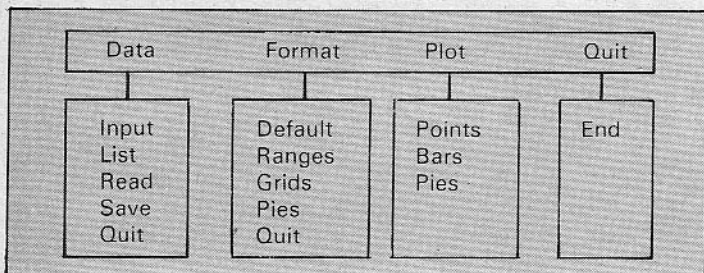


Figure I

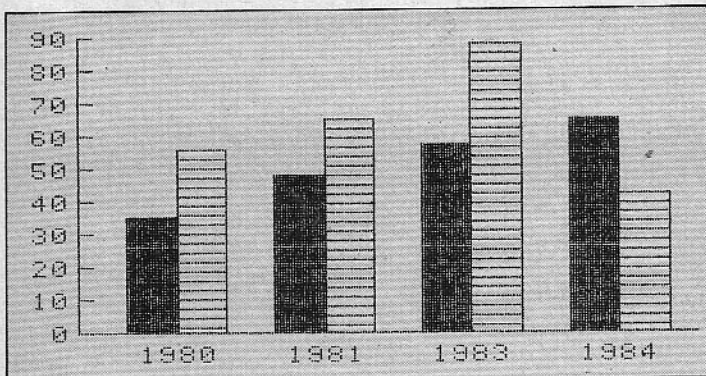


Figure III: All default options

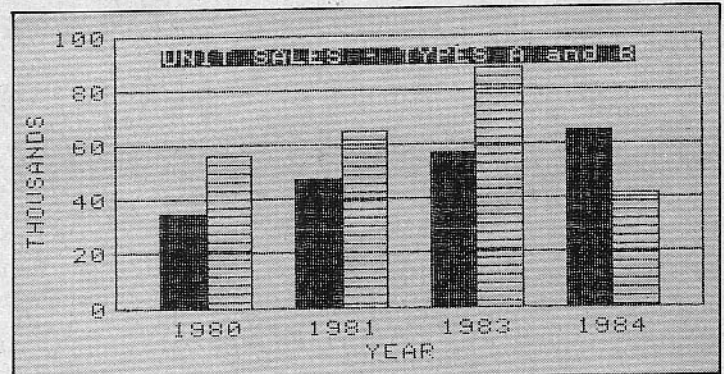


Figure IV: Overriding the defaults

```

80 DIM ZX(50),ZY(50),ZP$(50),ZD(
50)
90 REM

EXAMPLE PROGRAM

100 GOSUB 42400: REM SHAPE TABL
E LOADER
110 REM GRAPH PROGRAM
120 ZD$ = "": REM NO DATA
130 ZB(0) = 2: REM DEFAULT RANGE
S
140 TEXT : HOME : INVERSE : PRINT
"GRAPH PROGRAM MENU": NORMAL

150 PRINT : PRINT "1 DATA INPUT"
: PRINT "2 CHANGE FORMAT": PRINT
"3 DRAW GRAPH": PRINT "4 QUI
T"
160 PRINT : INPUT "WHICH ? ";MN:
IF MN < 1 OR MN > 4 THEN CALL
-198: GOTO 160
170 ON MN GOTO 180,200,220,290
180 REM DATA INPUT
190 GOSUB 55000: GOTO 140
200 REM CHANGE GRAPH LAYOUT
210 GOSUB 57600: GOTO 140
220 ZB = 0: IF ZD$ = "H" THEN ZB =
1
230 IF ZP$ = "Y" THEN GOSUB 400
: GOTO 270
240 IF ZD$ = "N" THEN GOSUB 550

250 IF ZD$ = "S" THEN GOSUB 500

260 IF ZD$ = "H" THEN GOSUB 450

270 GOSUB 57200: REM WAIT FOR S
PACE BAR
280 GOTO 140
290 END
400 REM PIE DIAGRAM

410 ZF = 1: GOSUB 49400: GOSUB 40
200: GOSUB 46100: GOTO 270
420 RETURN
450 REM CLEAR PAGE, SET RANGE,
DRAW AND LABEL - HISTOGRAM D
ATA
460 GOSUB 49400: GOSUB 49500: GOSUB
48600: GOSUB 41400
470 RETURN
500 REM CLEAR PAGE,SET RANGE,DR
AW AND LABEL - SYMBOLIC DATA

510 GOSUB 49400: GOSUB 49500: GOSUB
48600:ZT = 2: GOSUB 42600

520 RETURN :

550 REM CLEAR PAGE, SET RANGE,
DRAW AND LABEL - NUMERIC DAT
A
560 GOSUB 49400: GOSUB 47600: GOSUB
48600:ZT = 1: GOSUB 42600
570 RETURN :

55000 REM

DATA INPUT ROUTINE

55010 REM SETS UP DATA FOR PLOT
TING
55020 TEXT : HOME : INVERSE : PRINT
"DATA INPUT": NORMAL : PRINT
: PRINT
55030 PRINT "1 INPUT DATA": PRINT
"2 LIST DATA": PRINT "3 READ
DATA FILE": PRINT "4 SAVE D
ATA FILE": PRINT "5 QUIT"
55040 PRINT : INPUT "WHICH ? ";Z
A: IF ZA < 1 OR ZA > 5 THEN
CALL -198: GOTO 55040
55050 ON ZA GOTO 55060,55080,551
00,55120,55140
55060 REM DATA INPUT

55070 GOSUB 56000: GOTO 55000
55080 REM LIST DATA
55090 GOSUB 57300: GOTO 55000
55100 REM READ DATA FILE
55110 GOSUB 55300: GOSUB 57000: GOTO
55000
55120 REM SAVE DATA
55130 GOSUB 55300: GOSUB 56700: GOTO
55000
55140 RETURN
55300 REM

DATA FILE NAME

55310 PRINT : PRINT : INPUT "DAT
A FILE NAME = ";ZF$
55315 PRINT : INPUT "DRIVE NUMBE
R = ";ZE
55320 IF ZE < 1 OR ZE > 2 THEN CALL
-198: GOTO 55320
55330 ZE$ = STR$(ZE)
55340 RETURN
56700 REM

SAVE DATA ROUTINE

56710 REM REQUIRES THAT ZN, ZD$,
ZH(1), ZH(2), ZI(1), ZI(1),
ZI(2) AND ZF$ BE SET AS NECE
SSARY
56720 REM ZD$ SELECTS DATA STRU
CTURE
56730 PRINT CHR$(13); CHR$(4)
;"OPEN";ZF$;"D";ZE$
56740 PRINT CHR$(4)"DELETE ";Z
F$
56750 PRINT CHR$(4)"OPEN ";ZF$

56760 PRINT CHR$(4)"WRITE";ZF$
56770 PRINT ZN;" ";ZD$
56780 IF ZD$ < > "N" THEN GOTO
56810

56790 FOR ZI = 1 TO ZN: PRINT ZX
(ZI);";";ZY(ZI): NEXT
56800 GOTO 56910
56810 IF ZD$ < > "S" THEN GOTO
56840
56820 FOR ZI = 1 TO ZN: PRINT ZP
$(ZI);";";ZY(ZI): NEXT
56830 GOTO 56910
56840 IF ZD$ < > "H" THEN GOTO
56910
56850 PRINT ZH(1);";";ZH(2)
56860 IF ZH(1) = 0 THEN GOTO 56
900
56870 FOR ZI = 1 TO ZH(1): PRINT
ZP$(ZI)
56880 FOR ZJ = 1 TO ZH(2): PRINT
ZY((ZI - 1) * ZH(2) + ZJ): NEXT

56890 NEXT : GOTO 56910
56900 FOR ZI = 1 TO ZH(2): PRINT
ZP$(ZI);";";ZY(ZI): NEXT
56910 PRINT CHR$(4)"CLOSE ";ZF
$
56920 RETURN
57000 REM

READ DATA FILE

57010 REM SEE COMMENTS FOR SAVE
ROUTINE
57020 PRINT CHR$(13); CHR$(4)
;"OPEN";ZF$;"D";ZE$
57030 PRINT CHR$(4);"READ ";ZF
$
57040 INPUT ZN,ZD$
57050 IF ZD$ < > "N" THEN GOTO
57070
57060 FOR ZI = 1 TO ZN: INPUT ZX
(ZI),ZY(ZI): NEXT : GOTO 571
60
57070
57080 FOR ZI = 1 TO ZN: INPUT ZP
$(ZI),ZY(ZI): NEXT : GOTO 571
60
57090
57090 FOR ZI = 1 TO ZN: INPUT ZP
$(ZI),ZY(ZI): NEXT : GOTO 57
160
57090 IF ZD$ < > "H" THEN GOTO
57170
57100 INPUT ZH(1),ZH(2)
57110 IF ZH(1) = 0 THEN GOTO 57
150
57120 FOR ZI = 1 TO ZH(1): INPUT
ZP$(ZI)
57130 FOR ZJ = 1 TO ZH(2): INPUT
ZY((ZI - 1) * ZH(2) + ZJ): NEXT

57140 NEXT : GOTO 57160
57150 FOR ZI = 1 TO ZH(2): INPUT
ZP$(ZI),ZY(ZI): NEXT
57160 PRINT CHR$(4);"CLOSE ";Z
F$
57170 RETURN
57200 REM

SPACE BAR ROUTINE

57210 HTAB 8: VTAB 24: PRINT "PR
ESS SPACE TO CONTINUE": GET
ZA$: IF ASC (ZA$) < > 32 THEN
GOTO 57210
57220 RETURN :

57300 REM

DATA LIST ROUTINE

57310 TEXT : HOME : REM CLEAN S
CREEN
57320 IF ZN < = 1 THEN PRINT "
NO DATA ": GOSUB 57200: RETURN

57330 PRINT : PRINT ZN;" DATA PD
INTS"
57340 IF ZD$ < > "N" THEN GOTO
57370
57350 PRINT : PRINT "POINT": TAB(
8);"X "; TAB(20);"Y": PRINT

57360 FOR ZI = 1 TO ZN: PRINT ZI
: TAB( 8);ZX(ZI); TAB( 20);Z
Y(ZI): NEXT : GOTO 57480
57370 IF ZD$ < > "S" THEN GOTO
57400
57380 PRINT : PRINT TAB( 8)"LAB
EL": TAB( 20);"VALUE": PRINT

57390 FOR ZI = 1 TO ZN: PRINT ZI
: TAB( 8);ZP$(ZI); TAB( 20);

```

```

ZY(ZI): NEXT : GOTO 57480
57400 IF ZD$ < > "H" THEN GOTO
57480
57410 IF ZH(1) = 0 THEN GOTO 57
460
57420 PRINT : PRINT ZH(1); " GROU
PS WITH ";ZH(2); " BARS PER G
ROUP"
57430 FOR ZI = 1 TO ZH(1): PRINT
: PRINT ZP$(ZI)
57440 FOR ZJ = 1 TO ZH(2): PRINT
TAB( 5);ZY((ZI - 1) * ZH(2)
+ ZJ): NEXT
57450 NEXT : GOTO 57480
57460 PRINT : PRINT TAB( 8)*LAB
EL"; TAB( 20);"VALUE": PRINT
57470 FOR ZI = 1 TO ZH(2): PRINT
ZI; TAB( 8);ZP$(ZI); TAB( 20
);ZY(ZI): NEXT
57480 PRINT : GOSUB 57200
57490 RETURN :
57600 REM
GRAPH FORMAT ROUTINE
57610 TEXT : HOME : INVERSE : PRINT
"GRAPH FORMAT CONTROL": NORMAL
: PRINT
57620 PRINT "1 DEFAULT SETTINGS"
: PRINT "2 RANGE SETTINGS": PRINT
"3 TITLES": PRINT "4 ANNOTAT
ION/GRIDS":
57630 PRINT "5 PIE CHARTS": PRINT
"6 QUIT"
57640 PRINT : INPUT "WHICH ? ";Z
B: IF ZB < 1 OR ZB > 6 THEN
CALL - 198: GOTO 57630
57650 ON ZB GOTO 57660,57700,578
20,57870,57970,58270
57660 REM DEFAULT SETTINGS
57670 FOR ZI = 1 TO 10:ZP(ZI) =
0:ZB(ZI) = 0:ZO(ZI) = 0: NEXT
57680 ZP$ = "N":ZV(10) = 0:ZU(10)
= 0:ZB(0) = 2:ZB$(1) = "":Z
B$(2) = "":ZB$(3) = "":ZF =
0
57690 GOTO 57600
57700 REM RANGE SETTINGS
57710 IF ZD$ < > "N" THEN GOTO
57730
57720 PRINT : INPUT "XMIN,XMAX =
";ZM(1),ZM(2)
57730 PRINT : INPUT "YMIN,YMAX =
";ZM(3),ZM(4)
57740 ZB(0) = 1: PRINT : INPUT "D
O YOU WANT AUTO INCREMENTS (
Y/N) ? ";ZC$
57750 IF ZC$ = "Y" THEN GOTO 57
600
57760 IF ZD$ < > "N" THEN PRINT
: INPUT "MAX LABEL LENGTH =
";ZU(10): GOTO 57790
57770 PRINT : INPUT "NUMBER OF X
TICK MARKS = ";ZB(1)
57780 INPUT "NUMBER OF DECIMAL P
LACES = ";ZU(10)
57790 PRINT : INPUT "NUMBER OF Y
TICK MARKS = ";ZB(2)
57800 INPUT "NUMBER OF DECIMAL P
ACES = ";ZV(10)
57810 ZB(0) = 0: GOTO 57600
57820 REM TITLES
57830 GOSUB 56500
57840 PRINT : PRINT "TITLE STYLE
": PRINT "1 DEFAULT": PRINT
"2 UNDERLINE": PRINT "3 INVE
RSE ": PRINT "4 INVERSE UNDE
RLINE"
57850 PRINT : INPUT "WHICH ? ";Z
B(8): IF ZB(8) < 1 OR ZB(8) >
4 THEN CALL - 198: GOTO 57
850
57860 GOTO 57600
57870 REM GRIDS/ZERO LINES
57880 PRINT : INPUT "BORDER Y/N
? ";ZC$
57890 ZB(3) = 0: IF ZC$ = "Y" THEN
ZB(3) = 1
57900 PRINT : PRINT "GRIDS": PRINT
"0 NONE": PRINT "1 VERTICAL"
: PRINT "2 HORIZONTAL": PRINT
"3 BOTH"
57910 PRINT : INPUT "WHICH ?";ZB
(4): IF ZB(4) < 0 OR ZB(4) >
3 THEN CALL - 198: GOTO 57
910
57920 PRINT : PRINT "SHOW ZERO L
INE": PRINT "0 NONE": PRINT
"1 VERTICAL": PRINT "2 HORIZ
ONTAL": PRINT "3 BOTH"
57930 PRINT : INPUT "WHICH ? ";Z
B(5): IF ZB(5) < 0 OR ZB(5) >
3 THEN CALL - 198: GOTO 57
930
57940 PRINT : PRINT "ANNOTATE AL
ONG ZERO LINES: ": PRINT "0
NONE": PRINT "1 HORIZONTAL":
PRINT "2 VERTICAL": PRINT "
3 BOTH"
57950 PRINT : INPUT "WHICH ? ";Z
B(6): IF ZB(6) < 0 OR ZB(6) >
3 THEN CALL - 198: GOTO 57
950
57960 GOTO 57600
57970 REM PIE CHARTS
57980 IF ZD$ < > "H" THEN GOTO
58030
57990 IF ZH(1) = 0 THEN GOTO 58
030
58000 IF ZH(2) = 1 THEN GOTO 58
030
58010 PRINT "MULTIPLE DATA SETS
- CAN'T PLOT": PRINT : GOSUB
57200
58020 GOTO 57600
58030 ZM(1) = 0:ZM(2) = 1:ZM(3) =
0:ZM(4) = 1:ZP(1) = .5:ZP(2)
= .5:ZP(3) = .2:ZP$ = "Y": REM
X,Y,R AND FLAG
58040 PRINT : INPUT "DEFAULT VAL
UES (Y/N) ";ZAS
58050 IF ZAS = "Y" THEN GOTO 57
600
58060 PRINT : INPUT "SLICES IN %
ALREADY (Y/N) ? ";ZAS
58070 ZP(0) = 0: IF ZAS = "Y" THEN
ZP(0) = 1
58080 PRINT : INPUT "GENERAL OFF
SET (0-50%) = ";ZP(5): IF ZP
(5) < 0 OR ZP(5) > 50 THEN GOTO
58080
58090 PRINT : INPUT "OFFSET ANY
SLICES (Y/N) ? ";ZAS$
58100 ZP(6) = 0: IF ZAS$ = "N" THEN
GOTO 58170
58110 ZP(6) = 1: REM NOW GET SLI
CE INFO
58120 PRINT : INPUT "INPUT 0,0 T
O FINISH"
58130 PRINT : INPUT "SLICE NUMBE
R, OFFSET(X) = ";ZA,ZB
58140 PRINT : IF ZA = 0 THEN GOTO
58170
58150 IF ZA < 1 OR ZA > ZN THEN
PRINT "NOT VALID": GOTO 581
30
58160 ZO(ZA) = ZB: GOTO 58130
58170 PRINT : INPUT "CYCLE COLOU
RS (Y/N) ";ZAS$
58180 ZP(7) = 0: IF ZAS$ = "Y" THEN
ZP(7) = 1
58190 PRINT : INPUT "CYCLE SHADI
NG (Y/N) ? ";ZAS$
58200 ZP(8) = 0: IF ZAS$ = "Y" THEN
ZP(8) = 1
58210 PRINT : PRINT "SHADING TYP
E": PRINT "0 NONE": PRINT "1
PERPENDICULAR": PRINT "2 PA
RALLEL": PRINT "3 CROSS HATC
H": PRINT "4 SOLID":
58220 PRINT : INPUT "WHICH ? ";Z
P(9): IF ZP(9) < 0 OR ZP(9) >
4 THEN GOTO 58220
58230 IF ZD$ = "N" THEN ZP(10) =
0: GOTO 58260
58240 PRINT : INPUT "LABEL SLICE
S (Y/N) ? ";ZAS
58250 ZP(10) = 0: IF ZAS = "Y" THEN
ZP(10) = 1
58260 GOTO 57600
58270 RETURN

```



apple user Graphics Library Offer

This article is the last in the series. For the benefit of those readers who missed some of the articles the complete list of issues that featured the Apple User Graphics Library are given in the panel alongside. Back numbers are still available, and these are listed on Page 60.

A disc has now been prepared containing all the routines presented in the series and this is available for £5.95.

A complete set of photocopies of all the articles can be obtained, also for £5.95.

The cost of the disc plus the photocopied articles is £9.95.

Part 1	February 1984
Part 2	March 1984
Part 3	April 1984
Part 4	May 1984
Part 5	June 1984
Part 6	August 1984
Part 7	September 1984
Part 8	October 1984
Part 9	November 1984
Part 10	December 1984
Part 11	February 1985
Part 12	March 1985
Part 13	May 1985
Part 14	July 1985
Part 15	August 1985
Part 16	October 1985
Part 17	November 1985

Please use the order form on Page 61



particular application in some detail later, but to illustrate this particular point I will digress briefly.

Some of the work that I do is very low value and I may leave an invoice to accumulate over a period of say two months, adding to it as each small job is done.

The most frequent use of this disc is to make these entries, so I have arranged that the disc on booting takes me straight to the entry point in the particular record I need, switch on the Apple and I arrive at x in Figure I in 17 seconds.

It is at this point that I can go to the more infrequently used menu if I wish (I have a keyboard buffer, and enter the four letter key to the record during the boot process. With my last database the four letter words were kept for quite another purpose).

This whole screen (Figure I) is one record. Each of the seven fields is an array of 19 elements allowing quite complex calculations to take place within the record. (If that means nothing to you don't worry, it meant nothing to me a short while ago.)

Let me return to the main purpose of this article and show you what my 156 lines will do. If the particular application has no relevance to your needs, don't be put off. It is the timings and the manipulations that are of interest.

To demonstrate the power of Memdos on minimum equipment I will use an Apple II+, two floppy drives, Digitek 80 column card and Videx keyboard enhancer. The last two are not

essential, but I need upper and lower case, and want to squeeze as much on to the screen as possible. Plus, of course, the Memdos card, which incidentally does not, in my experience, interfere with programs running under DOS 3.3.

The application I shall describe is to process a renewal order received from a book agent, acting on behalf of a subscriber or subscribers to journals, up to the point where the invoice or acknowledgement of payment is ready to be

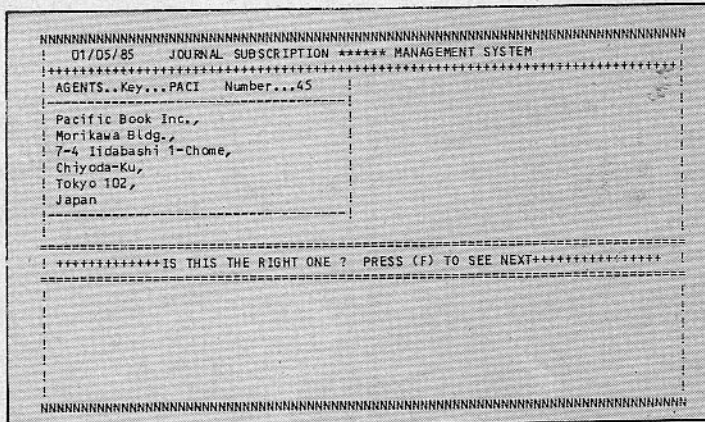


Figure III

printed.

For the purposes of demonstration I have limited the journal titles to two, the two files of subscriber details to approximately 130 each, and the file of book agents to 100.

These three files plus the screen masks – easily prepared with a Memdos utility – are on one disc.

The menu and program are on the other disc, together with three more files. These are an invoice file, a single record file containing the last invoice

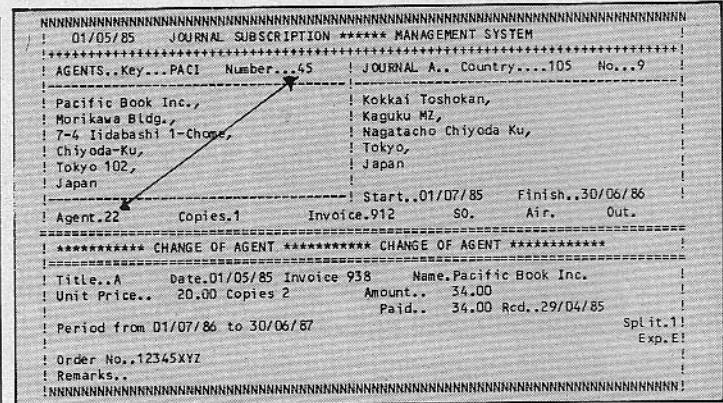


Figure IV

Escape or Return, so on entering the last letter the search is begun and the address appears (Figure III) in 2.2 seconds.

When this address is accepted the file containing the invoice number is read and a third window for invoice details appears on the screen with the next invoice number, from the invoice number file, and date, from window one, automatically entered within 2.3 seconds.

At this point I am asked for a single letter entry to determine which title is being processed. This letter will define parameters sent to the subroutine which deals with searching and reading the subscriber files. It is also instantaneously transferred to the invoice window.

I am now in a position to enter the invoice details. When all the details including the subscription period are accepted, calculations are carried out – agents discount etc – and in less than a second the updated window redisplayed.

A further question here asks if payment accompanied the order, if so the window is again updated with the cursor at the date received entry point. This will also automatically determine whether, at the printing stage, an invoice or acknowledgement of order is sent.

A fourth window is now displayed for subscriber details and is dealt with as was the agent window, except that a key number is used to identify the subscriber. Again the address is found and displayed in less than three seconds.

In addition the agent's key number stored in the subscriber record is compared with the number of the agent displayed

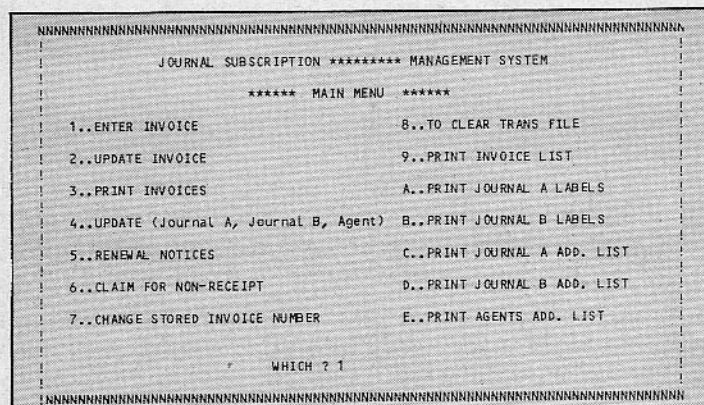


Figure II

and if there is a change of agent a message is displayed (Figure IV) and the cursor placed in the relevant entry point so that the subscriber record can be amended.

Memdos allows the cursor to be placed in any of the data entry points of a window. This is the position in Figure IV. I cannot leave this window until the two numbers are the same.

In a package this could be an irritation, but I have decided it is what I want, and can alter it at any time.

A single keystroke now transfers the new subscription period etc from the invoice window to the subscriber window.

I give myself the opportunity to alter or update the subscriber and invoice window, say a change of address. If satisfactory, a single keystroke saves the invoice record, updates the subscriber file with the amended record, increases the

invoice number file by one, and saves the pointers in the transaction file for later printing, all in six seconds.

I am asked two further questions: "Another subscriber on this invoice?" and "Another invoice?" Answering "No" to both returns me to the menu in five seconds. I am now ready to print the invoices or wait until I'm having lunch, or process another invoice later.

To summarise, from switching on the machine to returning to the menu the total machine time is 48 seconds, or 35.5 seconds from the menu back to menu.

During this time six files have been opened and have remained active throughout, five windows have been opened and remained on the screen (one of them invisible to receive responses to questions).

Two files of approximately 100 records have been search-

ed, and one file with one record read. Two records have been written to separate files and the records of two other files updated.

During this time various calculations have been made, and data in records from different files compared.

There may be other programs or systems for the Apple II+ which will allow me to do all this, but I have not come across them. I have only scratched the surface. The system can be transferred to hard disc, and as most of the times mentioned earlier are disc access times the increase in file size and decrease in times would be dramatic.

There is a utility which allows Memdos to cohabit with DOS 3.3 so existing records can be transferred.

So far in the context of database operation, whatever I have wanted to do I have been able to do, and there are still

options within the system which I don't fully understand.

The program described makes use of called subroutines to which parameters can be passed. The longest subroutine is 38 lines, making program alteration easy.

The report printing facilities included in Memdos are, although very versatile, a bit slow. How much this is a function of my Spinwriter printer I am not sure, but ordinary Basic print statements can be used for simple listings and label printing etc.

The manual is not very satisfactory, but the results outweigh all problems. I shall not be getting rid of my old reliable Apple II+ yet.

But Memdos does run on the IBM and other machines, and with a few alterations I would be off again. I would certainly not buy a machine which did not support Memdos.

BASIC PASCAL

P-tral: BASIC to Pascal Translation Software

P-tral is the user-friendly translation software that converts Applesoft BASIC programs to Apple Pascal.

- Translates any business, scientific, graphics and game software from source.
- Translation results approach 100%.
- Translates any sized program—Large or Small.
- Essential compile directives generated.
- PIONEER version only £125 + VAT.

REQUIREMENTS: 64/128K RAM, Dos 3.3, 80-column card, Apple II, IIe, IIc, Apple Pascal 1.1 or 1.2

To order or request more information contact:

DMS ELECTRONICS LTD
Bretton Court, Manor Road, Wales Village,
Sheffield S31 8PB, England
Tel: Worksop, (0909) 773399

P-tral is published by: **WOODCHUCK INDUSTRIES®**
340 West 17th Street, New York, NY USA 10011
(212) 924-0576 / (212) 206-6490

Memdos Software

We've turned business/accounting software on its head!

Frustrated by inflexible computer packages and poor support?

We'll give you a powerful Open Item Purchase/Sales/Stock System* and in 2 days, train you to customise it to suit your own specific business requirements (Order Processing, Costing, etc)

Suitable for most popular micros—Apple, Macintosh, IBM, Olivetti, Pinnacle, and others

Single or multi-user versions available

EASY to learn, EASY to change, EASY to use

*Requires the Omnis 3 database

TO DO IT PROPERLY—DO IT YOURSELF

BUSINESS CONTROL SYSTEMS

FREEPOST, Caterham, Surrey CR3 9UH Telephone: (0883) 40861/43553

ALL the talk nowadays about fancy new programming languages makes one wonder whether to admit to liking and actually using Apple's Basic, Applesoft.

There seems to be a growing aversion to Basic in any form in schools, which is a shame as it is probably the easiest high level language to learn.

Unfortunately schools tend to favour Logo, et al. Have you ever seen a sales ledger program written in Logo? It's a pity that so much emphasis seems to be put into the drawing of pretty pictures, or making a turtle crawl about the floor.

These are not really the stuff of computer programming in the Eighties. Unfortunately schools seem to consider that it is. I don't think for one minute that I'm alone in these thoughts.

However, back to ancient Applesoft.

When the Apple II was first put on the market it had Applesoft, written jointly by Apple and the well-known purveyor of languages, Microsoft.

It was, and still is, one of the best implementations of Basic on a micro. Little did Apple suspect at the time that the Apple II series would be around for so long.

As time went by, and more and more people used Applesoft, it was clear that there were several shortcomings in the Applesoft/DOS system, rather than just the language.

Various reasonable solutions were found for some of them, such as making room for a high resolution page in the middle of a big program in memory, but some of the "features" we just learned to live with.

ProDOS has changed all that.

A disc operating system (DOS) is just software that looks after all the reading from and writing to discs, both floppy and hard. Apple's DOS 3.3 is the DOS more usually associated with systems that have programs in Applesoft Basic or machine code. (Pascal has its own built-in DOS, which we'll ignore here.)

DOS 3.3 is quite long in the tooth now, although various enhancements are available to make it read and write much faster - Fast-DOS and Diversi-

A new lease of life for Applesoft

CHRIS PAYNE studies the dramatic effect that ProDOS has on Apple's Basic

DOS, to name but two.

Apple have now produced ProDOS which has been featured in these pages before. ProDOS is now included instead of DOS 3.3 when you buy one or two Apple disc drives.

What isn't obvious unless you have a close look at ProDOS is that most (if not all) of the "problem features" of Applesoft have been solved, or there is now some other way of doing things.

For people who use Applesoft, they may not know what they are missing.

When you boot ProDOS, if there is a program called BASIC.SYSTEM on the disc then this is executed. This program, along with ProDOS, automatically gives Applesoft its new lease of life. In this article I'll use ProDOS to mean ProDOS and/or BASIC.SYSTEM.

Apple seem to have emptied their complaints box on the floor and worked through Applesoft's

funnies one by one.

Take CHAINING for instance. It now works! Well, better than it did - see Peter Meyer's article in the July issue.

There were one or two problems with the old method of CHAINing, and you had to fiddle about with a small machine language program. With ProDOS, all you do is PRINT DS"CHAIN OTHERPROG", and all your variables are preserved, the other program runs in memory, and all is well.

Any files that were opened by the program that called CHAIN are left open. ProDOS is so fast in loading programs that you don't ever need to have the whole of your program in memory at one time.

It is very easy to split a program into various functions and CHAIN to them when needed.

Not content with making CHAIN work, if you do PRINT DS"CHAIN OTHERPROG,

@500", the CHAINed program starts running at line 500. This means that you can now go to different entry points in the CHAINed program dependent upon the place that you left the previous program.

Talking of variables, how would you like to leave your MEGASTARINTERGALACTICNEBULARWARS with your score intact, then resume it after you've had tea. Now you can.

ProDOS allows a filetype of VAR (more on filetypes later). This is a file of all the variables in memory at the time that the file is written to disc.

If you do PRINT DS"STORE ALLMYVARS" then the file ALLMYVARS will have preserved all the variables for posterity. To get the variables back, reload file ALLMYVARS with PRINT DS"RESTORE ALLMYVARS".

You could also use this feature to start a program in a certain condition, dependent upon which variable file is loaded. You could also change the condition of a program while it is running, to enable or disable features.

Applesoft contains a house-keeping routine, sometimes called garbage collection, that clears out memory that has been used for variables. This doesn't cause the loss of data that has been carefully put into the machine.

When you execute INPUT AS repeatedly, old AS data is still in memory, albeit not easily accessible to the programmer. Consequently programs that have a large number of frequently-changing variables, both numeric and string, use up



Have you ever seen a sales ledger written in Logo?

/EXAMPLE								
NAME	TYPE	BLOCKS	MODIFIED	CREATED	ENDFILE	SUBTYPE		
CUST.DATA	TXT	17	12-DEC-84 21:45	28-JAN-85 12:43	7744	R=	64	
UTILITIES	DIR	1	26-MAY-85 14:53	26-MAY-85 14:53	512			
AUX.DATA	TXT	1	3-DEC-84 2:23	28-JAN-85 12:44	146	R=	15	
*SMACH.0	BIN	1	6-NOV-84 11:55	28-JAN-85 12:44	420	A=\$9400		
*SMACH.1	BIN	1	6-NOV-84 11:56	28-JAN-85 12:44	125	A=\$9461		
*MAIN.A	BAS	41	11-JAN-85 15:00	28-JAN-85 12:44	20430			
*BACKUP	BAS	7	13-DEC-84 0:33	28-JAN-85 12:45	2989			
*PRODOS	SYS	30	14-DEC-84 11:27	28-JAN-85 12:45	14048			
*STARTUP	BAS	3	2-JUL-84 17:24	28-JAN-85 12:46	537			
*BASIC.SYSTEM	SYS	21	14-DEC-84 11:27	28-JAN-85 12:46	10240			
BLOCKS FREE:		150	BLOCKS USED: 130		TOTAL BLOCKS: 280			

Figure 1: Example of a Volume Directory

lots of memory.

Without ProDOS, if AppleSoft saw that it was running low on variable space it sometimes put your program to sleep for several minutes to do a garbage collection.

This is clearly inconvenient, and would need more careful use of arrays and the like to lessen the likelihood of subjecting the user to a long pause. ProDOS comes to the rescue with FRE, a bit like the old FRE(O), and is used to force a garbage collection.

ProDOS automatically does a very fast garbage collection when it needs to, but if you want to do it in a program all that is needed is PRINT DS"FRE", and it's all over very quickly.

Gone are the days of "stuck" programs when you don't know whether they have crashed or not.

IN# and PR# are commands that enable you to take input from a specific peripheral card slot in the Apple or send output to a slot. The slot in question may contain a clock, a printer interface or maybe a serial card connected to a modem.

With Applesoft you are restricted to getting input or sending output just to slot numbers. Using ProDOS you may now say PRINT DS"IN#\$300" and input will now come from your machine code routine at \$300.

Output may be diverted in the same manner with something like PRINT DS"PR#\$320".

If you have an empty slot,

such as slot 5, then by doing PRINT DS"PR#5 A\$300" near the beginning of your program, or anywhere else that you want to use your new output routine, just do PRINT DS"PR#5". In this way it is not necessary to keep quoting the address of the routine.

You will also be pleased to know that TRACE and NOTRACE now work while accessing discs. There are enhancements to most of the disc access commands, but I'll just describe a few of the more interesting ones.

Before I do that though, here

the size of a file by a three figure number showing how many sectors on the disc the file takes up.

ProDOS uses pathnames to describe where on a disc to find files. A disc now has a name, such as /MASTER. The volume directory is the basic directory of the disc. In this case CATALOG /MASTER would list the contents of the volume directory. The / preceding the name is very much a part of ProDOS.

ProDOS allows the use of more than one directory file on a disc. For instance, assuming UTILITIES is another directory

Gone are the days of "stuck" programs when you don't know whether they have crashed or not

is a brief description of some of the philosophy behind ProDOS.

A disc has a catalogue, or catalog as we've been forced to call it with DOS 3.3. This holds details about each file on the disc. DOS 3.3 allows various types of file, machine code data or programs, Applesoft programs, integer programs and textfiles. These are designated B, A, I and T respectively.

There is also an indication of

file, then /MASTER/UTILITIES/LINE.EDITOR would be the pathname to get to the LINE.EDITOR program.

If you were to CATALOG /MASTER/UTILITIES you would get a list of files in the UTILITIES directory file.

It is clearly very easy to put data or programs into separate directories to make for a more logical structure to the disc.

Instead of having to keep

track of where in an enormous file certain sections of data are stored, something like /RECIPES/EGGS/OMELETTES/SPANISH would lead you straight to the correct recipe.

Part or all of a pathname can be put in the PREFIX. This is the default pathname that is automatically prefixed to any other part of a pathname that is entered.

If you don't want to type a long pathname while continuously working on a file, by entering PREFIX /RECIPES/EGGS, this would be attached to the front of any other filename used. There is much encouragement to keep filenames short with ProDOS.

Figure 1 shows an example of a volume directory. Under the heading NAME, is a list of the filenames in the directory. Taking the files in turn and considering the other headings, CUST.DATA is a textfile (TXT in the TYPE column) of 17 blocks in length. A block is 512 bytes. It was last modified on 12-DEC-84 at 21:45. The file was first created on this disc on 28-JAN-85 at 12:45.

It may seem an anomaly that it was modified before it was created, but there is a simple explanation. The file was copied from another disc on 28-JAN-85, but the modification date correctly reflects the last alteration to the contents of the file.

ProDOS allows the interfacing of a clock card, so that the time and date are available to it whenever any updating on disc is needed.

If you haven't got a clock card there is a short program supplied that allows the system date to be set. It is possible to also set the time, which isn't so useful as it obviously doesn't change if there is no clock.

ENDFILE reveals how long the file is in bytes, in this case 7744, and is a decimal number. SUBTYPE shows the length of a record in a random access file, in this case 64, and is also decimal.

This file would normally be opened with the usual PRINT DS"OPEN CUST.DATA,L64". Had the figure in the SUBTYPE column been 0, we could probably, but not conclusively,

assume that it was a sequential file.

UTILITIES is a directory file that has one block allocated to it. The other columns reflect similar data about the file as before. AUX.DATA is another textfile with a record length of 15, that was last modified at some unearthly hour of the morning.

Turning to the machine code programs SMACH.0 and SMACH.1, these are both locked, shown by the asterisk before the file name, as in DOS3.3. TYPE shows that these are binary files, BIN, and the SUBTYPE column now shows the starting address in hexadecimal.

The length of the file in ENDFILE is still in decimal.

File MAIN.A is of type BAS, and as you might expect is an Applesoft Basic program. It takes up 41 blocks, and is 20430 bytes in length.

The two files PRODOS and

BASIC.SYSTEM are of type SYS, a special type of machine code program following a convention that is beyond the scope of this article. They are, however, the two programs that form the basis of the ProDOS system.

Instead of a HELLO program that could be called anything you liked under DOS 3.3, STARTUP is always the HELLO program with ProDOS. It may be of any filetype, including text. This would make it an EXEC file in the same way as with DOS 3.3.

As can be seen, a ProDOS directory contains a large amount of information for the programmer.

All this data can be read from within an Applesoft program. This would make showing a list of files that are available to a user considerably easier than before. See page 226 of the manual Basic Programming with ProDOS for more details.

The last line of the CATALOG shows the number of blocks free, blocks used and the total number of blocks available on the device. For a normal floppy disc this is 280.

For a Profile 5mbyte hard disc which ProDOS supports without any modification, the total number of blocks available is in the region of 9700.

The sharp-eyed will not have failed to notice that the individual number of blocks that each program uses when added together does not equal 130. Seven blocks are missing. These are used for the volume directory and other housekeeping purposes.

There is to all intents and purposes, no reasonable limit to the number of filenames on a disc, although the limit in a volume directory is 51. Other directories' capacity is only limited by disc space.

Up to 32 mbytes may be allocated to a volume, assuming

a large enough hard disc, and up to 16 mbytes may be in a file. As many as eight files may be open at one time.

If you had a random access file that needed to be altered in the middle of a record with several strings in it, with DOS 3.3 you had to use the ,B parameter to get to the correct place in the record. This meant counting how many characters into the record you needed to go before starting to write, and hopefully not forgetting to subtract 1, as the ,B parameter starts from 0.

ProDOS has an F (field) parameter that helps in this situation. This means that if you say PRINT D\$'WRITE MYFILE,R100,F3' and PRINT A\$, then A\$ will be written as the *fourth* string in the record, without having to know the length of the previous strings. The field parameter starts from 0, as would be expected.

The group of disc commands that have seen the biggest face-lift, are probably BLOAD, BRUN and BSAVE. These are all to do with loading, running or saving binary files. They may be programs, hires pages or just chunks of data.

It is now possible to have four possible parameters that define how the file will be treated. I won't go into too much detail here, as it would take too much space. You can say PRINT D\$'BLOAD MYBINFILE,A\$ 2000,E\$3FFF' to load part of a file.

PRINT D\$'BLOAD MYBINFILE,A\$2000,B\$100,L\$FF' loads from byte \$100 of the file to the \$FFth byte. There are clearly other combinations of defining the portion of file to be loaded.

You may also BRUN a program part way through the file. Any type of file may be BLOADed, so a textfile may be loaded anywhere in memory for examination or for some other purpose.

A portion of memory may also be BSAVED as any type of file. A textfile may be loaded into memory, modified, and put back on a disc as a textfile, without anyone being the wiser.

If you buy ProDOS after reading this article you will

Apple Ref. No.	Title	Price
A2D2010	ProDOS User's Kit	£35
A2D2037	Basic Programming with ProDOS	£23
A2W0010	ProDOS Tech. Ref. Manual	£18
A2W0013	ProDOS Assembly Tools	£35
A7P0001	Apple Workshop Binder	£9
A2L0082	Applesoft Tutorial Manual	£23
A2L4031	Applesoft Reference Manual	£20

The User's Kit contains a manual that describes the functions of various programs provided with ProDOS, such as the FILER that may be used to transfer programs and files between discs, and for formatting discs, among other things.

CONVERT allows the transfer of programs and files between DOS 3.3 and ProDOS in either direction.

Basic Programming with ProDOS details the changes and additions made to Applesoft, and includes a disc of sample and useful programs.

Remember that if you buy a disc drive with a controller card, you will get the ProDOS User's Kit. You won't get the Basic Programming with ProDOS though.

The ProDOS Technical Reference Manual describes ProDOS in detail, and is a necessity if you are going to interface

hardware or machine code programs to ProDOS. It's not strictly necessary if you confine yourself to Applesoft, but good to have around.

The ProDOS Assembly Tools contains the ProDOS assembler, which is excellent, and a superb program called Bugbyter. This allows you to step through a machine code program and observe where it is going wrong. That's its fundamental purpose, but it contains many other tricks.

The binder will hold the Technical Reference and the Assembly Tools manuals, as these are printed in loose-leaf A5 size.

The Applesoft Manuals are just that, giving full details of Applesoft. These apply even if you are using DOS 3.3.

All prices were taken from the July 1985 Apple II range price list.

List of manuals for ProDOS and Applesoft users

notice that the Basic Programming with ProDOS manual mentions that old chestnut, using hi-res graphics pages with large programs.

The manual says that you may issue an HGR or HGR2 command, and the program will be magically wrapped round the graphics page. Unfortunately this has not been implemented in ProDOS yet. Who knows whether it will in the future? It would be good if it was.

ProDOS will work on an Apple II+ or IIe, but you do need the 16k language card installed in a II+. This extra 16k of memory is already part of an Apple IIe.

ProDOS allows the use of most 80 column cards available for a II+. If you have a IIe, however, there are several 80 column cards available that have an extra 64k of RAM fitted.

Using DOS 3.3 it is possible to access this extra memory, but it really is tortuous. Several commercial programs use the extra 64k, such as Quickfile, Applewriter IIe, various Visicalc options, etc.

It would be nice to be able to use this extra memory easily and usefully, wouldn't it? Well, you've guessed it, ProDOS comes to the rescue.

How often have you been fascinated by the thought of a RAMdisc, but decided that they were too expensive? At the small difference in price that an extended 80 column card costs compared to an unextended one, it's a wonder that the latter don't sit on dealers' shelves forever. Perhaps they do.

A RAMdisc is a portion of RAM that looks like a disc to the rest of the system. There are various versions on the market, usually sold as a card to go in one of an Apple's peripheral slots.

The advantage of a RAMdisc is that instead of having to wait for a disc to whirr round and get to the correct part and the read/write head to move across it, a RAMdisc holds the data in normal RAM chips. Therefore the access to data or loading of programs is very fast. For example, a 25000 byte program loads in something like 0.7 seconds. The same program is

```
HOME
?"* A very moving experience for EDASM! *"
BLOAD EDASM.SYSTEM,TSYS,A#2000
CREATE/RAM/EDASM.SYSTEM,TSYS
BSAVE/RAM/EDASM.SYSTEM,TSYS,A#2000,L#095
BLOAD EDASM.ED
BSAVE/RAM/EDASM.ED,A#3000,L#191
BLOAD EDASM.ASM
BSAVE/RAM/EDASM.ASM,A#3000,L#14079
PREFIX,D1
-/RAM/EDASM.SYSTEM
```

Figure II

written to RAM in about 1.7 seconds.

When ProDOS is booted, the RAM on the extended 80 column card is initialised as a volume called delightfully, /RAM. This volume gives 128 blocks of "disc". As a normal floppy gives 280 blocks, this is just under half a floppy's worth. It may not seem much at first, but it's surprising how much you wonder what you did without it.

One of the delights of machine code programming is that it is so slow. When you have modified the source code of the program you are working on, it has to be assembled, then back to Applesoft, try it, and if it hasn't hung the system, or corrupted a disc, then it's back to the editor to tweak it a bit more.

It was with this performance in mind that I thought about getting the assembler and editor into /RAM and running it from there each time I needed to alter

start of a program, then just search and read that copy of the file? If the file needs to be modified during the course of a program you may find that it wouldn't take much time to keep the version in /RAM updated as well as the file on disc.

To get EDASM, the assembler/editor into /RAM, you will need an EXEC file. As with DOS 3.3 and many other systems, an EXEC file is a text file, but consists of a series of commands as if you had just typed them in.

EXEC files may also contain any Applesoft commands that are valid when entered from the keyboard, such as HOME.

/ASM.TOOLS is the disc containing all the parts of the assembler/editor system for ProDOS, so this needs to be available, or a disc containing EDASM.SYSTEM, EDASM.ED and EDASM.ASM.

The next step is to create the

```
REM * RUN THIS TO RESTART EDITOR *
REM * JUST TYPE "-ED" *
PREFIX,D1
-/RAM/EDASM.SYSTEM
```

Figure III

the program that I was working on.

Running the assembler in /RAM considerably speeds up the whole proceedings. The principle is useful for other purposes too, so run your frequently used programs from /RAM.

How about using BLOAD to move a textfile to /RAM at the

EXEC file. This can be done using EDASM.ED. The text to be entered is in Figure II. Care must be taken to type it in correctly, with particular attention paid to the hex addresses. The - in the last line is ProDOS's shorthand for RUN.

The previous line sets PREFIX to whatever disc is in drive 1. Doing this just before

running EDASM.SYSTEM saves having to set PREFIX by hand once you are in the editor. I call this textfile MOVED, but you can call it what you like. Just type -MOVED from Applesoft to get EDASM into /RAM, and it runs automatically.

Make another textfile called ED using the text of Figure III. I called it ED so that it only took four keystrokes to get back to the editor. This file should be available after you have come back to Applesoft from EDASM. Typing -ED sends you very quickly back to the editor.

Once you are used to having /RAM available, it is a simple matter to save the source file that you are working on to /RAM as well.

This makes it quicker to see if it will assemble, before bothering to let the assembler generate an object file.

Since EDASM.SYSTEM was started up from /RAM, then the assembler will automatically be run from there also. This obviously means that you don't need the assembler available to the system on disc. If you do any amount of machine code programming the ProDOS assembler is worth considering. It is very much like the DOS 3.3 Toolkit assembler but much better.

Most disc operating systems go through some revision before they are perfect, and ProDOS is no exception. There are one or two minor problems in ProDOS 1.0 that are fixed in ProDOS 1.1.1. Most software, however, will work fine, so don't feel that ProDOS is full of bugs.

If you have a problem that you suspect could be due to ProDOS not functioning correctly, your local Apple dealer will be only too pleased to give you the update. He will also have a list of problems that the update fixes, so that you will be able to tell if yours is included.

Applesoft was always to my mind a perfectly respectable vehicle for real application software. ProDOS allows this to continue, with the added advantage of some better ways of handling machine code programs in conjunction with Applesoft.

SPREADING THE LOAD

BILL HILL reports on an unusual use of a spreadsheet – making out an office's duty roster

WE are all accustomed to thinking of spreadsheets as number-crunchers to be used when we have masses of related figures to handle. But there's another side to the spreadsheet. You can, if you put your mind to it, use one to perform quite complex logic which has really very little to do with number-crunching at all, and to carry out tasks for which you might never have considered a spreadsheet to be the obvious choice.

Over the past few months I've been developing a "logical spreadsheet", now at an advanced and usable stage, which in 2.5 minutes can complete a job that used to take me hours of mental juggling and leave me with a very sore brain indeed.

It was only after mentioning it to a few people – some of whom gave me very strange looks – that it became clear that this was a pretty unusual spreadsheet application.

Although it was developed on the Macintosh, the principles it uses will work on any spreadsheet, running on any computer, provided that it supports the IF ... THEN ... ELSE formula which you can paste into cells in Multiplan (or Excel, when that is released) or enter in cells in Lotus Jazz.

First, there was the problem. I'm second in charge of a busy newspaper office with nine staff, and one of my more onerous responsibilities is making up the duty rosters.

It's a complex task. We work seven days a week, with three different shifts Sundays to Fridays, and one man on day shift on Saturdays. All staff work

a four-day week, except the boss, who works a nine-day fortnight.

Annual holiday entitlement is such that there is someone on holiday almost every week of the year. The boss and myself, and one specialist writer, work constant day shift and there are six reporters on shift working. The specialist writer and one of those reporters are available for newsdesk stand-in when the boss and I are both off at the same time.

Ideally, it should be possible, if you do a little mental arithmetic, to have each reporter working one Saturday in six, six late shifts every six weeks, six back shifts every six weeks, and to work every second Sunday.

But when it comes down to the practicalities, there are always some special factors to

consider. Holidays never fit in to any pattern you can use. A reporter working four days of late shift, Sunday to Wednesday, gets five days off immediately afterwards.

The man working on Saturday must also work on the following day. A man on holiday should stop work by Friday at the latest in the preceding week, and should not be asked to return to work on a Sunday. And so on ...

All these, and a host of other special factors, combine to make the task of working out the permutations one which can make strong men weep.

There are so many conditions which you have to bear in mind at each stage, each of which rules out certain combinations for one reason or another, that in trying to remember them all is mind-bending. If I had a pound

for every time I thought I'd got it right, only to have someone point out a problem which meant going right back to the beginning again, my bank manager would be a happy man.

The computer seemed the obvious answer, but how?

My first thought was a program in Basic, and I did some initial work in that on a string of different machines. It was something I'd pull out every now and again and have a bash at. But by the time I'd got to 300 lines of Basic and realised I'd solved only one section, I was beginning to despair.

You'll have gathered by now that I'm not really a programming enthusiast. I prefer to use off-the-shelf business software.

Then a friend suggested Multiplan. (*Sounds like one of those advertisements. You know the kind: "I was always the wallflower at the party – until a friend told me about Multiplan"!*)

I started work on that but my efforts were eventually defeated because Multiplan would not handle a spreadsheet large enough. However, as you will see elsewhere in this issue, I then received a very early copy of Excel from Microsoft and that succeeded admirably.

Using a spreadsheet seemed a very unlikely way of doing things – but it works. Here's how.

Let's take the case of the newsdesk slot first. If he's on duty, that slot should be filled by my boss, the news editor. If not, then I'm the second choice. If I'm not available either, then one of the two stand-in men can be used. So it's really a case of considering, in order of priority, which of the four of us is available.

You can do this in a spreadsheet with only two cell formulae – IF ... THEN ... ELSE, and IF(SUM(...), THEN ... ELSE, which is a combination of IF() and SUM() formulae pasted together.

The first task is to assign everyone a code number – boss, 1, me, 2, and then on down the list of staff in alphabetical order – 3 to 9 inclusive.

The spreadsheet works in three stages, like the computer itself – input, processing and output. The input side is small, and merely records the code

SHIFTS				
	A	B	C	D
1	STAFF	CODE NO	HOLIDAY LAST WEEK	7
2	GILL	1	HOLIDAY WEEK 1	2
3	HILL	2	HOLIDAY WEEK 2	2
4	CAMERON	3	HOLIDAY WEEK 3	0
5	CRAINEY	4	HOLIDAY WEEK 4	6
6	DRON	5	HOLIDAY WEEK 5	9
7	FORBES	6	HOLIDAY WEEK 6	9
8	MACGEE	7	HOLIDAY WEEK 7	8
9	MCKENZIE	8	LATE LAST WEEK	5
10	STEVART	9	DESK LAST SUN	2
11			GILL 4(1=Y,2=N)	1
12			ON LAST SAT	3
13			LATE 2 LAST WK	4
14			BACK SU LAST WK	9

Input stage. Input numbers are in Column D

numbers of everyone on holiday from the week before the six-week cycle through until the week after. Who worked late shift the previous week, the previous Saturday, and so on.

You have only to type in the 14 new numbers for the next six week cycle, tell the spreadsheet to calculate again, and go away and make a cup of tea for the four minutes or so that it takes the computer to work everything out (I told you it was a complex problem.)

The information you enter in input covers the whole six weeks. The processing stage consists of six separate groups of three columns, and the starting information for each week is pasted in at the top of each set of three columns, either from input or from the results of the processing columns for the week preceding.

So the starting figures for Week 1 come entirely from input, those for Week 2 from a combination of input and the results of Week 1 calculations,

and so on – bearing in mind that with each successive week there are more factors to consider. For example, Week 4 starting figures use input and results from Weeks 0, 1, 2, and 3.

The method of calculation is best demonstrated by example. Let's take the newsdesk slot for Sunday and Monday of Week 1. If the boss is on holiday, he's obviously ruled out, or if he was on holiday the previous week, or if he worked the previous weekend. (In normal circumstances, with either of us on holiday, we work weekend about.)

This takes four cells. The first three:

$IF(G2=1,2,1)$

$IF(G3=1,2,1)$ and

$IF(G9=1,2,1)$ all return values of 1 if all the conditions under which the boss would be working are met.

The fourth cell, in the next column (H), is $IF(SUM(G2+G3+G9)=3,1,0)$. That cell returns a value of 1 if all

conditions have been met, and a value of 0 if any of the conditions are not met. At first, I used the binary Yes=1, No=0 for the first three cells, but changed that for clarity and used zero only in the fourth cell for reasons which become apparent later.

The next group is of five cells. The first, $IF(SUM(G2+G3+G9)=3,2,1)$, returns a value of 2 if the previous sort has decided the boss is working, and a value of 1 if it has not. The next three cells go through the same sorting process as in the first group of three, examining the same three cells, G2, G3 and G9, but this time looking for the code number 2 instead of 1.

The fifth cell in column H adds up the previous four, and returns a value of 2 if all conditions have been met and 0 if any have not been met. It can be seen that the first of these five cells will return a value of 2 if the boss is filling the slot – thus ruling out two people working the same shift.

The remainder of this section goes on to consider the two stand-in men in the same way. Each successive sort is one cell larger, since it also has to consider whether any of the previous sorts have met the conditions required.

The result of these four sorts is an output of four cells in column H. Three of them will return values of 0, while the fourth will return a value which is the code number of the person who will fill the slot – in the case of my spreadsheet, that is 1, 2, 6 or 7.

So the sum of these four cells takes the value of the code number of the person who will fill the slot (one number plus three zeros), and this value is pasted in to the final output stage for Week 1 in tabular form.

And that's it. The whole spreadsheet from then on consists of building up a complex matrix of small sorting sequences like that, calculated on the basis of different start figures



Elephants love Apple users...

And to prove it, here's a very special offer
exclusively for subscribers to  apple user

**FREE
DISC!**

Subscribe or renew your subscription now and you will receive an Elephant Memory Systems 5¼in single sided single density disc absolutely free!

**FREE
VOUCHER!**

You will also receive a voucher to enable you to obtain three boxes of 10 5¼in discs for the price of two, redeemable at your nearest Elephant Memory stockist.

Existing subscribers may renew early in order to take advantage of this offer.

Please use the order form on Page 61

which are determined by results for the previous weeks, or the initial input.

In order, these determine: Desk Sunday and Monday, desk Tuesday and Wednesday, desk Thursday and Friday, reporter working four late shifts, reporter working two late shifts, Saturday reporter, reporter on back shift Sunday, Monday and Tuesday.

The man who's working back shift on Sunday automatically works day shift on Friday. The man who works Saturday precedes it with back shifts on Wednesday, Thursday and Friday. So those sorts in total fill in all blanks but day shifts.

It's not quite that simple, of course. The sorts become progressively more complex when you consider the shift working of each reporter, until certain parts of the matrix take as many as 12 or more cells for each sort.

For example, if he's already been selected for four nights of late shift, Sunday to Wednesday, he's ruled out for back shifts on Sunday, Monday, Tuesday, Wednesday, Thursday and Friday as well as day shift on Saturday of that week. And with each successive week they become even more complex.

You don't have to enter the formulae for all cells manually, though. Large sections of the sheet are almost identical, and you can copy and paste whole sections of rows and columns from one week to the next, inserting additional cells or changing cell references if necessary.

The output stage consists of six tables, one for each week, which are filled in with values (which correspond to staff code numbers) pasted in from the six processing stages.

As I said at the beginning, the spreadsheet is not finished yet. For instance, it does not calculate who's working on day shift on many of the days.

However, that's not such a pressing problem as the rest of the shifts. Nobody minds working more day shifts than his share - while everybody objects to more nights. And once all the nights, back shifts and Saturday are filled in, it's easy enough to manually complete the rosters, since the hardest part is licked.

I've learned a lot in the process of putting it together, and can think of more elegant

ways of doing certain things, which I plan to incorporate in later versions. One example of this is coming to realise that if each sort is carried out in alphabetical order the process tends to be biased towards picking those whose names appear first. I've taken some steps to correct that, but can think of a more sophisticated approach which I intend to try out.

But it does work - I'm now using it routinely - and if your problem is not as complex, and does not have to span a six-week period like mine, then this might be an answer.

I can think of a few possible applications for the technique. It could be used to build up faultfinding trees for equipment diagnosis, where the fault-finding steps can be phrased in a manner which generates Yes or No answers.

A word of warning though, even though it may not apply to you. With every sort being repeated, in a more complex sequence, six times, my particular spreadsheet gobbles up disc space like there's no tomorrow. So far it takes 185k, and each successive sort sequence you add - since it has to take account of everything that's gone before - takes up progressively more disc space than the one before.

SHIFTS		
	F	H
1	WEEK 1	WEEK 1
2	ON HOLIDAY	2
3	HOLIDAY LAST WK	7
4	HOLIDAY NEXT WK	2
5	LATE LAST WK	5
6	LATE2 LAST WK	4
7	ON LAST SAT	3
8	BACK SU LAST WK	9
9	DESK LAST SUN	2
10	GILL 4?	1
11		
12	DESK SUN/MON	DESK SUN/MON
13	GILLSORT	1
14		1
15		1
16	HILLSORT	2
17		2
18		1
19		2
20	MACSORT	2
21		1
22		1
23		2
24		1
25	FORBSORT	2
26		1
27		1
28		1
29		1

First Sort. Week 1 starting figures are pasted in by making cells assume the value of relevant input cells. Sum of four cells in Column H gives code number of man working newdesk Sunday/Monday.

SHIFTS						
	X	Y	Z	AA	AB	AC
1	WEEK 1		HOLIDAY	2		
2		DESK	DAY	BACK	LATE	OFF
3	SUNDAY		1	8	3	ALL OTHERS
4						
5	MONDAY		1	8	3	5
6						
7						
8	TUESDAY		1	8	3	
9						
10						
11	WEDNESDAY		1	4	3	
12						8
13						
14	THURSDAY		1	4	5	3
15						8
16						
17	FRIDAY		7	8	4	5
18						
19						
20	SATURDAY	NONE		4 NONE	NONE	ALL OTHERS

Output stage for Week 1. In the latest version numbers are translated back into names

Excel... the most sophisticated spreadsheet/database/graphics package I've ever seen on a personal computer

—enthuses
BILL HILL

LOTUS launched its integrated Jazz package for the Macintosh a couple of months ago with the slogan: "Now the Macintosh really means business".

As a piece of advertising hype, I suppose it was pretty well conceived — and it certainly seems to have made an impact, judging by reports of the sales figures for the product.

Readers of *Apple User* will recall that I reviewed Jazz, and was perhaps less than enthusiastic about it than might have been expected. Well there was a good reason for that — though I was sworn to secrecy not to reveal it at the time.

Jazz would have been a lot more impressive had I not been sitting on the only copy in Europe outside the company itself of Microsoft's Excel, a business application with what to my mind is a sensible level of integration which avoids the "Jazz-of-all-trades-master-of-none" syndrome.

What the Mac needs to achieve success in the corporate market is software that puts it ahead of its competitors — software like Aldus PageMaker, which I also reviewed recently, which allows you to do things on the Mac you could do on no other system costing less than ten times its price.

Excel is another such software package. If any application has the right to say, "Now the Mac really means business", then it is this one.

This is the most sophis-

ticated spreadsheet/database/graphics package I've ever seen on a personal computer. The fact that it runs at the moment anyway exclusively on the 512k Macintosh or a Lisa running MacWorks2, is likely to prove as much of a boost for Apple as for Microsoft. Lotus 1-2-3 was by itself a major reason for many sales of the IBM PC. If there is any justice in the world at all, Excel will do exactly the same for the Mac.

Not to mince words, it's a piece of genius which is so far ahead of anything else around that it's unfair to compare it — Jazz included. It doesn't try to do as many different things, but what it does do, it does much better.

I've had a copy of Excel since May this year, and I've lived with it and run it hard since then. I started out with a PreAlpha test version of the package which still had 200k-odd of uncompiled sourcecode on it, was clearly still at the testing stage, and had the habit, which you'd expect in any package at that stage of development, of falling over now and again. But it was still fabulous, and completely useable.

Since then I've received a newer, compiled version which runs faster, saves faster and which I haven't yet been able to crash, which indicates that Excel is now just about ready for launching.

In fact, I've lived with it for so long now and from such an early stage that it seems like my baby as much as Microsoft's, so it's

hard to know where to begin describing it.

This is REAL number-crunching power. Excel is massive, and has a matrix of no less than 4.194 MILLION cells (16,384 rows by 256 columns). It has terrific features like a macro facility, which lets you record spreadsheets as you create them, so they can be run by someone with no experience at all.

Anyone who's used Multiplan will know that it is a wonderfully simple yet powerful package. Now picture Multiplan, with all of the facilities and functions you ever wished it had, with a powerful and sophisticated graphics package like Chart integrated to it, and a spreadsheet-oriented database, and that's Excel.

As one example of the power of the beast, the Paste Function command, which will be familiar to Multiplan users, now allows you to build formulae containing combinations of no less than 73 different functions, from the simple SUM() and IF() to such arcane ones as TREND(), TIME() and SEARCH(). And yes, Excel worksheets will handle time in the same way as any other kind of number.

As a package it has so many sophisticated features that, despite its user-friendliness, the manual is no less than 529 pages long — almost 200 of them devoted to macros, which on Excel have their own new and powerful computer language.

Like Jazz, Excel is big (400k).

It needs two discs, one for the system and data, the other for the program itself. However with a little good housekeeping you can leave yourself with almost 200k headroom on your data disc.

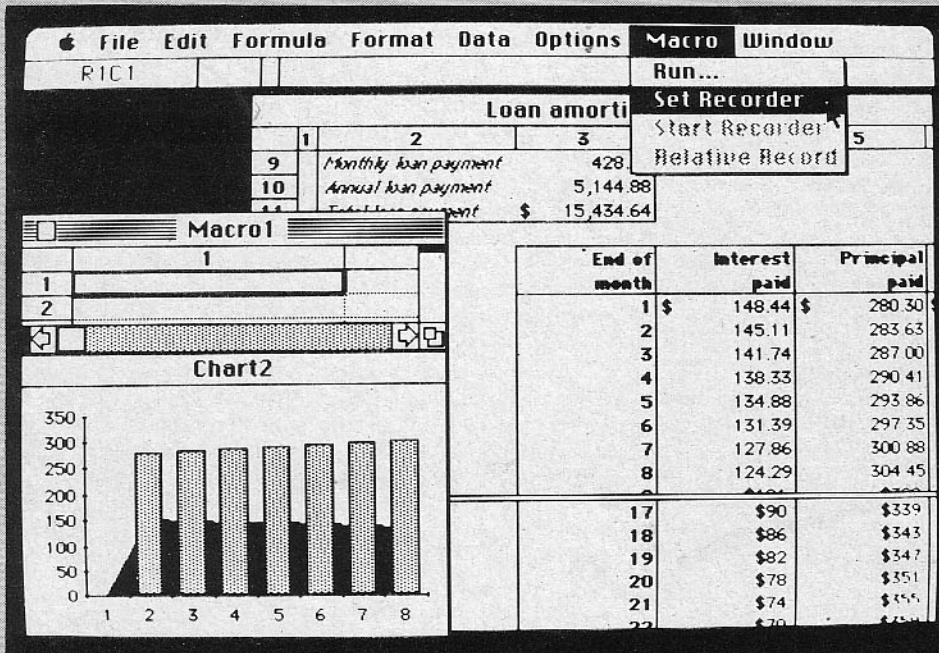
When you first enter the application you automatically get a new spreadsheet. The program loads quickly when booted from cold like that. However as you might expect, once you actually start building up a large spreadsheet it takes quite a while to load.

There's a good reason for that. Excel will not run on the 128k Mac, since it sits mostly in RAM with some overlaying. Disc accessing is very limited and that makes it incredibly fast. You can, for example, scroll from top to bottom through all 16,384 rows in a fraction of a second.

One very interesting feature is that Excel will also run on the Lisa (MacXL), which has 1mbyte of RAM — and this gives you about 800k of space for data.

Microsoft tells me that the package has been designed to support Macs with memory expanded up to 4mbytes of RAM — about the addressing capability of the 68000 chip. Do you see the future beckoning, perhaps?

I'd been wracking my brains for a long time over a system of calculating office duty rosters, which I have outlined elsewhere in this issue. Multiplan wouldn't do it — it kept running out of memory and falling over. Jazz



Excel display, showing linked worksheets

with the Switcher. So if you want to integrate word processing or communications software (or both) with it, you can quite happily do that.

And here is the really interesting bit. Excel will go on sale in this country at a price, including Switcher, of £395 – which may seem like a lot, but take it from me, you are getting your money's worth. However if you already own Multiplan the price will be £200. If you own both Multiplan and Chart then you will pay only £100.

Excel is totally windows-dependent, so those of you who have been following the long saga of Microsoft Windows for the IBM PC will be aware that it might still be some time yet before Incredibly Boring Machine users will be able to run a version of Excel. If you know any I suggest you're careful about showing it to them – unless you can stand to see grown men cry.

might just have looked at it (but possibly at the expense of my sanity).

Excel allowed me to build up a spreadsheet which came to 185k – and took about 2.5 minutes to carry out a job which used to take me hours.

Since then I've used the application for many other tasks, more normal spreadsheet jobs involving number-crunching, preparation of charts, and so on. It handles the most complex formulae and relationships with ease. So much so that I haven't been able to test it to its limits, because I haven't found them yet.

Once you have built up your spreadsheet, which you do in pretty much the normal way, it's the work of a second or two to turn it into graphics. Excel's graphics capability, as you would expect, looks very much like Chart.

All you have to do is highlight the cells in the spreadsheet which you want to graph. You then pull down the file menu and select NEW. Up pops a window which gives the choice of creating either a new spreadsheet, a new macro, or a new chart. Click Chart and a Chart window opens up with your spreadsheet figures already loaded.

Now it only remains for you

to select the chart configuration you require – and that's it.

Another new feature of Excel is its database capability. You can use an Excel worksheet as a database, and then sort, find or extract information from it which matches certain criteria and put it on another part of the worksheet, for example, personnel records, sales to clients, accounts payable, or whatever. Sort commands will support wildcards, and you can, say, use a formula as a criterion.

I don't propose to delve too deeply into the mysteries of Excel here. This is a package with such possibilities that I think people will still be finding new things they can do with it 10 years from now.

But it's worth taking a look at the general thinking behind it.

The fact is Multiplan had to be changed for the 512k Mac, anyway. I've heard differing views of the cause of the difficulty from Apple (which says it was a software problem) and Microsoft (which says it was a hardware problem) and I'm not qualified to judge. But Multiplan has an upper limit of spreadsheet size on the 512k Mac, which means you would be just as well running it on a 128k machine.

As the saying goes, it's an ill wind that blows no one any

good. And what has blown our way is a package which the average business computer user would give his or her eye teeth for. Financial modelling, sales forecasts, accounting – Excel will make them a doddle.

Microsoft is packaging Excel

Fattening up a 128k Mac

WHEN the Mac was first released many people thought that they'd never need more than the 128k version. In the land of the 64k machine, the 128k was king.

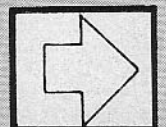
Of course what few people realised at the time was that the Mac's operating system gobbled up a fair bit of memory and didn't leave anywhere near 128k available to the user.

This meant that, for example, Microsoft Basic running on the Mac had less room for user programs than Applesoft on a trusty Apple IIe – a silly situation in anyone's book. Also programs like Excel simply wouldn't

fit in the 128k machine and can only be run on 512k.

If you're using your Mac for work then, the chances are that you need a 512k version, but you bought a 128k version because that's all that was available at the time. Congratulations – you're on the upgrade path.

That's how we were here at *Apple User* until recently. Worse, we had an early 110 volt



MacReview

US keyboard version that needed all the software localising back to the US version.

When we finally decided we could stand it no more, a quick call to Apple (UK) confirmed that they could convert us to a UK 240 volt system. However at the time they couldn't manage to upgrade us to 512k.

What could we do? Well, we could attempt the upgrade ourselves. After all, hadn't another magazine recently published a lengthy blow-by-blow account of fattening a Mac?

If you saw that article and fancied undertaking the task, you'd better make sure you read the letters that followed. As expected, Apple (UK) eschewed the DIY process, but several other people also suggested that the article was not a sufficient guide.

We'd also heard rumours of

upgrades being offered for little more than the cost of the chips. However we decided not to bother trying to trace one of these operators.

Instead we gave P & P Micro Distributors a call. They undertake an upgrade and give their own guarantee on the work for 12 months. Not only that, but they only charge £399.

Off went the Mac to P&P then. It was back in a couple of days. Apparently the upgrade doesn't take very long, but they like to keep the machine on test for some time.

Of course it *looked* the same when it came back. We knew it would, but there was a feeling that we should have been able to see some difference, maybe like the stickers which used to proclaim: "This appliance has been converted to natural gas".

The difference started to

become clear when we booted the Switcher. Instead of being told to "go get a 512k Mac", the Switcher loaded in and allowed us to flip back and forth between applications. If you're not easily impressed, try the Switcher - it will blow your mind.

The next difference showed itself when we copied a disc. With only the built-in drive, copying was a slow process. However, with all that spare RAM now available, copying speeds up appreciably.

Of course we can run Excel now, and all those other programs which will come along requiring a 512k Mac.

Even if you're only using MacWrite and MacPaint to write very short reports with pretty pictures in them, a 512k upgrade will allow you to have both applications in memory, so

you can save quite a lot of time.

It's clear that the Mac ought to have been a 512k machine from the start, and it's reasonably certain that the 128k machine will be phased out eventually.

If you've got a 128k machine you should seriously consider the upgrade. Apple offer the "official" upgrade path, but in keeping with much of their pricing policies of late it is grossly over-priced.

You can attempt the upgrade yourself, but this isn't something we'd recommend. In fact, P&P's upgrade seems to be the ideal solution. It is competitively priced, quickly and efficiently carried out, and supported with a guarantee.

It's given our Mac a new lease of life, and it could do the same for yours.

Cliff McKnight

RAMWORKS for APPLEWORKS

RAMWORKS is the sensational new memory card for the Apple IIe that gives the Appleworks user previously unheard of memory capacity. And more.

- * 736K AppleWorks Desk Top
- * 80 Column Display + Double Hi-Res
- * Simultaneous Ram-disk for AppleWorks.
- * 5,100 records per database file.
- * 5,100 lines per word processor file.
- * Segments files for disk storage.
- * OPTIONS: 437K Visicalc; 1 Meg Ram-Disk; RGB Output.

The RAMWORKS card plugs into the Apple IIe auxiliary slot and completely replaces an 80 (or extended 80) column card. In use it functions and behaves EXACTLY like Apple's extended 80 column card, but with much more memory. It is TOTALLY compatible with ALL Apple 80 column software.

APPLEWORKS EXPANSION

Ramworks Size	Desktop Size (*)
128K	101K
256K	183K
512K	367K
1 MEG	736K

- (*) Ramworks is supplied with an expander disk which modifies your copy of Appleworks to give increased desktop and file sizes.
- (*) The Appleworks program itself can also use Ramworks to simultaneously operate as a Ram-Disk, while keeping the same desktop size! (256K Ramworks or larger).
The speed-up is dramatic - particularly in spreadsheet work!
- (*) For files larger than disk capacity, Ramworks automatically segments the file to spread the file over more than one disk.

ORDERING INFORMATION

128K Ramworks	£249.00
256K Ramworks	£299.00
512K Ramworks	£399.00
1 Meg Ramworks	£649.00

OPTIONS

Ram Drive for Ramworks - Software	£29.00
(Turns Ramworks into a solid state disk drive)	
CP/M version of Ram Drive for Ramworks	£29.00
Visicalc IIe Expander (pre-boot disk)	£29.00
RGB Option (can be added later)	£129.00

NEW ... NEW ... NEW ... NEW

ALSO AVAILABLE FOR APPLE IIc

Z-RAM

Z-RAM is available with either 256K or 512K of additional memory PLUS a powerful Z-80 micro-processor for running CP/M software.

Z-RAM is installed inside your IIc, and expands Appleworks, acts as a Ram-Disk and runs CP/M!

256K Z-RAM	£499.00
512K Z-RAM	£599.00

Add £1.00 P&P per order. Add VAT at 15%

ALL GOODS CARRY A TEN-DAY NO-QUIBBLE "MONEY BACK IF NOT DELIGHTED" OFFER. PLUS ONE YEAR GUARANTEE.

BIDMUTHIN TECHNOLOGIES

42 New Broad Street,
London EC2M 1QY.
Tel: 01-628 0898

Guilty as charged... Infocom really are brilliant



IT was a wet night when I arrived at the mansion. As I entered, my reporter's instincts made me wonder whether I might be a little too conspicuous. I needn't have worried.

Over by the fireplace Titania, queen of the fairies, was chatting up a sheik. She was going on about one of her horses, name of Lurking Grue.

A vampire seemed to be enjoying himself hugely. A six-foot rabbit cavorted with a man-sized peanut butter sandwich. Three mice in dark glasses brushed past Pacman. An astronaut strolled up to the bar.

Near the entrance a gorilla was doing a splendid job of looking after the guest's coats. Other assorted weirdos were discussing everything from politics to local scandals while the more athletic took to the dance floor.

By comparison, the cowboy looked the very model of sobriety and good taste. The man in the outlandish cowboy outfit was me.

I wasn't having a bad dream — the nightmare would come later. No, I was attending one of Veronica Ashworth's famous halloween parties.

Actually I was mixing business with pleasure. Pleasure because I had been invited as a friend of the hostess,

and business because my editor thought there might be a good story in it.

The fairly queen, none other than mine hostess Veronica, had clearly been knocking back the Buck's Fizz from quite early on. Her words were slurred and she was none too steady on her fairy pins.

Emphasising a point to the sheik, she waved her glass on high and succeeded in slopping alcohol and ice cubes all down her tinsel dress. Using a word that would make an elf's hair curl, she staggered off to clean up.

A little later, I saw Veronica again. She was slumped on the floor of her office elsewhere in the mansion. She was less than dead drunk — she was dead, period.

And if that wasn't enough to dampen my party spirits, my cowboy's lariat, which I could have sworn I hung up in the closet with my damp coat, was wrapped tightly round her pretty little neck.

Just to put the cherry on the cake, a bullet lay by the body. Guess whose gunbelt had one empty cartridge loop? I could almost hear the click of the jail cell door.

So there I was, plunged deep into *Suspect*, a superb new text adventure from Infocom. *Suspect* is one of their best and follows in the footsteps of

Deadline and *Witness*, their earlier detective adventures.

Deadline called you in to investigate a suspicious suicide, while *Witness* placed you as an eye witness to a dastardly crime.

In *Suspect* you find yourself far more embroiled in murder most foul than ever before — all the evidence points to you as the prime suspect. Unless you can find out who really dunnit and pronto, your reporting days are over.

The case is stacked against you from the outset. There's the rope for starters. Then there's the bullet and gunbelt. You're also an outsider, a social nobody.

Yes, it's a set-up all right but since you know you're not a killer, it follows that the real murderer must be among the motley collection of upper-crust characters present at the party.

The police are soon on the scene, among them Sgt. Duffy, of *Deadline* fame. What you must do is to build up a case by exploring the mansion and grounds, watching and talking to the various characters, and analysing and deducing.

Only by gathering enough irrefutable proof and presenting it to the police can you establish your innocence and another's guilt.

You can't make a citizen's arrest — you have to convince

the detective to do that. You can accuse people though, but much good will it do you if you go around making wild allegations.

You have but a few hours (game time) to solve the mystery. Failure to do so will result in your being arrested and found guilty of murder.

Suspect is graded as an advanced adventure, so it's likely you'll be arrested and convicted many times over many weeks. Never mind, the preceding stimulation and entertainment more than compensates for being falsely imprisoned.

All the Infocom hallmarks are here, massive vocabulary, sophisticated input parser, dazzling and detailed prose, twists and turns, good humour, plenty of original puzzles and multiple solutions.

No doubt about it, this is yet another excellent adventure from Infocom — how do they keep it up? I accuse Infocom of being maddeningly brilliant. Prove their guilt beyond a shadow of a doubt yourself by buying *Suspect*. I rest my case.

Bob Chappell

Title: Suspect
Author: Dave Lebling
Publisher: Infocom
Requirements: Apple II or Macintosh formats available

WHEN Apple decided to upgrade its Apple II with the IIe in January 1983, an 80 column display card of its own was offered. This then put the machine in the mainstream of computers already offering this facility.

During the design of the card 64k of RAM was added as an option and as an afterthought, circuitry allowing a increased graphics resolution of 560 by 192 pixels.

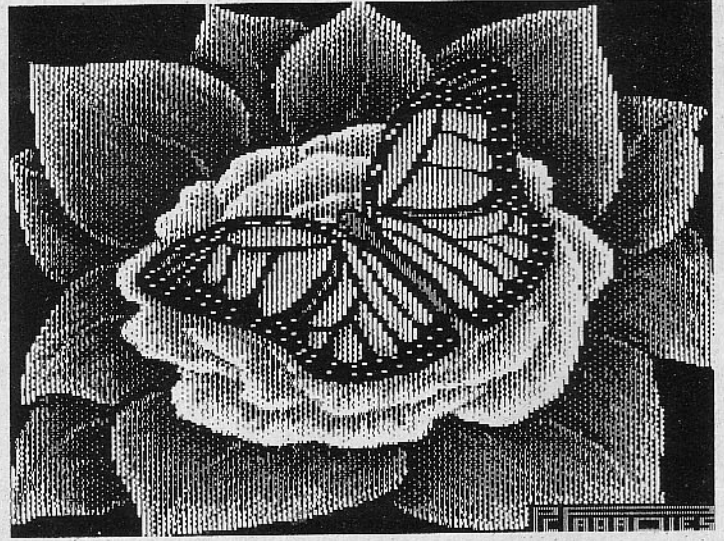
This is the double hi-res graphics mode which is now built into the IIc. As the 80 column card has now become a part of most Apple systems, software companies are starting to produce programs to take advantage of the new graphics capabilities.

The most recent of these is a freehand drawing program from Broderbund Software called Dazzle Draw, similar in principle to MacPaint.

The thing that makes this program a cut above anything else at present available is the way in which the author has incorporated speed, colour and features. The colour for those of you with a good monitor is amazing - 16 colours can be used at any time.

The program makes use of Macintosh-style icons, windows and pull down menus. The working screen is divided into

Dazzle Draw makes the most of double hi-res



three sections.

The top shows the title blocks for the menus and in the top right corner there is an undo box. This feature allows you to erase your last action.

The centre and largest part of the screen is the drawing area. To the bottom of this is the options screen which shows the colours or tools available for use on the option you have chosen from the menus.

Also on this part of the screen is a scroll bar which, when

moved by the cursor, allows you to work with the covered areas of the screen. The drawing tools for use with the program are extensive. These include several different paint brushes, spray paint, lines, shapes and so on.

Text can also be used. There are only two fonts, but there are two sizes with each and both have three different styles.

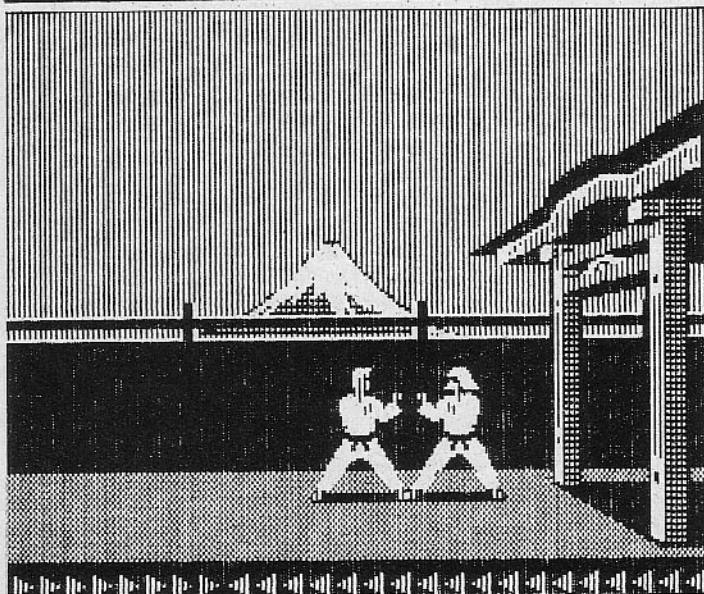
My favourite choice from the tools menu is the flood fill which lets you fill a desired area with any pattern or colour. If this still

doesn't look right the pattern can be exchanged for any other.

The only thing I would have liked to have seen is a lasso as in MacPaint, but that's a different story.

When you've made a mistake or wish to manipulate your picture in some way the edit menu comes to the rescue. The choices of the menu allow you to capture any area then invert it, flip it horizontally or vertically.

After you have captured your section you can cut it to memory



Karate caper makes for 3D thrills

A KARATEKA, according to the game, is one skilled in the art of karate. Makes sense, right? Well, once again you are the hero of this game, and what a hero you have to be.

The story, as it goes, is that the beautiful princess Mariko has been captured by the evil warlord Akuma, and imprisoned in the deepest dungeon of his fortress, high up in the mountains.

Your task is obvious - enter Akuma's fortress and risk your life to save her.

The first thing you notice is the superb use of graphics. Not a byte has been left un-programmed in this game. The graphics are beautifully drawn and the animation is so smooth

and paste it anywhere else on the screen or even change its colours.

You're probably thinking that all the previous menus are enough for any program, but Dazzle Draw still offers more from the goodies menu.

From here you can choose mirrors which let you divide the screen in any plane to give a mirror image of either side.

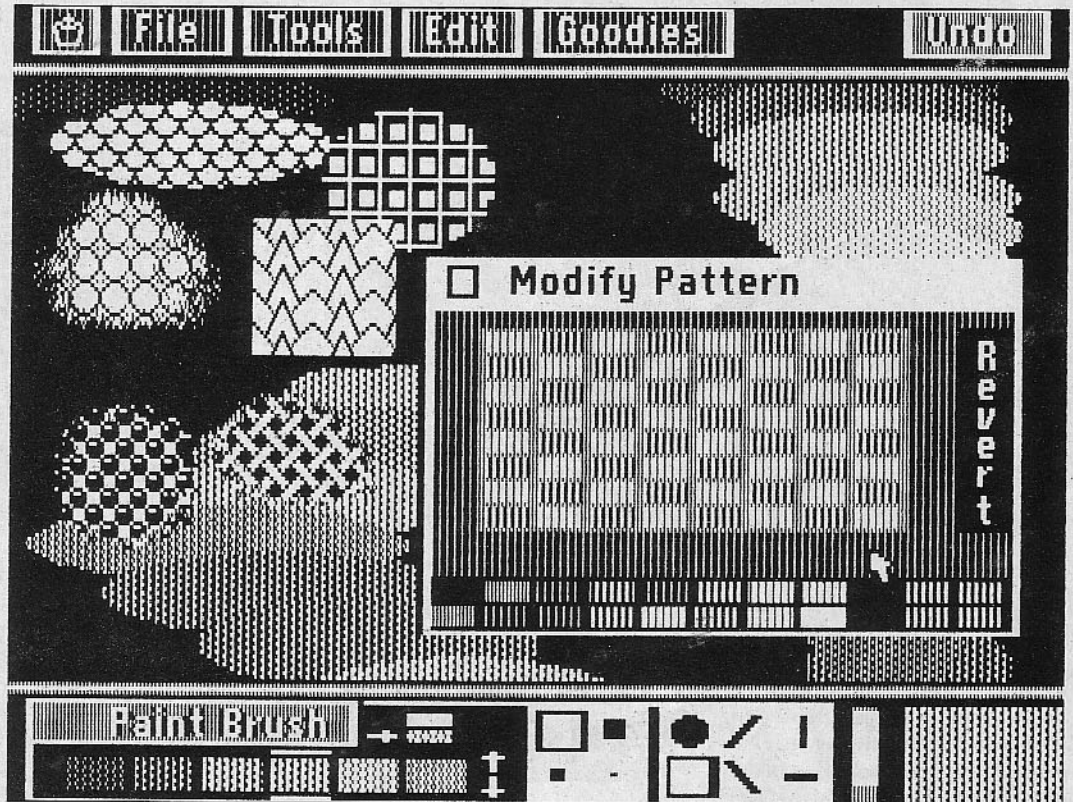
The next feature is the best on this menu – the modify pattern which allows you to change an existing pattern or create a new collection, which can be saved to disc.

There is also a view picture option which lets you look at your whole drawing, removing the menu titles and function bars. And if you don't like what you see you can erase the screen with the clear picture command.

If the picture was up to expectation you can print it to paper – it's all as simple as that... well maybe, if you have a printer and interface that the program accepts.

There are 11 printers to choose from, all the Epson range including the JX-80 colour printer or the Apple Scribe.

I printed out the pictures shown here with an Image-writer and SuperSerial card, in black and white of course. If you



do have a non-standard printer you can use a graphics dump program that accepts double hi-res pictures, something like Triple-Dump by Beagle Bros Software.

Dazzle Draw runs under the ProDOS operating system, and during the booting of the program you are asked for a choice of filing systems, Easy file, which is similar in action to DOS 3.3, or Professional file which can create multi-levelled

files the same as in ProDOS.

My only criticism is the use of disc space. You can only save eight pictures on a normal disc, and when using the slide show utility, six pictures.

However the program is flexible in the amount of different input devices you can use. I used an Apple mouse but a joystick, Koala pad or Apple graphics tablet can be used.

I think David Snider has produced a stunning program,

and just as I was starting to lose faith in new software. Ten out of ten to Mr Snider, a piece of software worth every penny.

Edward Alejos

*Title: Dazzle Draw.
Author: David Snider.
Publisher: Broderbund Software.
Requirements: Apple IIc or IIe with extended 80 column card.*

you'd think you were watching a film. Akuma's fortress really appears as a dark and mysterious place, which is a direct effect of the atmospheric graphics, and very much to the author's credit.

The whole game has a 3D aspect, which really makes you feel you are there.

In the demo you see the princess being sent to her cell, and when the door shuts behind her she turns around accompanied by a quick bit of dramatic music, collapses on the floor and starts crying.

The accompanying sad music really makes you feel sorry for her.

Then along comes the hero, who climbs over the top of a cliff

(appropriate "hero" music). As you walk the background scrolls along as well very smoothly.

You then meet your first opponent, and must defeat him using your karate skills. Before each fight the speaker lets out a wonderful digitised yell which sounds quite realistic. Control is by joystick or keyboard, so nobody gets left out.

You have three kicks available to you – middle, high and low. Likewise, you also have three punches. One thing I wished the game had was a range of defensive moves to block your opponent's punches and kicks. Still, it probably would have made it that little bit more complicated to play.

When you defeat the first

guard you start running towards the fortress entrance, and after a few steps the scene changes to show Akuma in his lair.

Accompanied by more dramatic music, he calls a guard and orders him to go after you. The guard bows and then starts after you.

The scene then changes between the pair of you in a movie fashion, until he reaches you. Then you have to fight him to the death.

There are different types of fighters for you to defeat, each wearing a different battle mask and each with his own style of fighting.

You also have to contend with various traps and the savage falcons sent to kill you.

Ultimately, you'll have to fight Akuma himself.

All in all, this is quite a game, and it will take time, stamina and skill to win.

The end result is well worth it (but not without more unseen surprises for the unwary). Highly recommended especially if you like adventure games, but with a lot more graphic and arcade-like involvement.

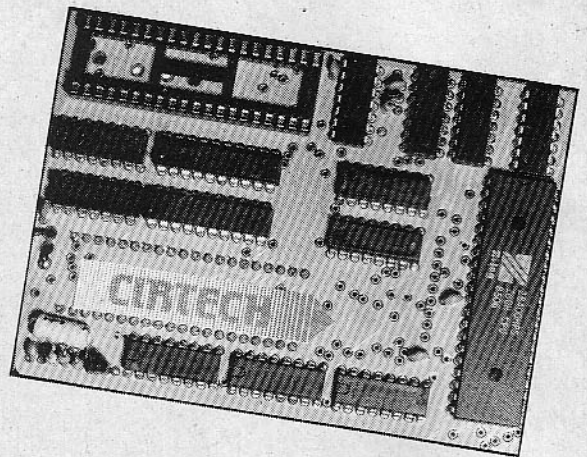
It's probably one of this year's most exciting games.

Leon Seltsikas

*Title: Karateka.
Author: Jordan Mechner.
Publisher: Broderbund Software.
Requirements: Any 48k Apple II.*



CP/M for the IIc gives the best of both worlds



IN 1984 Steven Frankel wrote that "more Apples run CP/M than any other brand of computer in the world".

It was a surprising statement to say the least, but when you consider that Microsoft claims to have sold more than 100,000 SoftCards since 1979 and you add the sales of other Z80 cards to that, the statement starts to seem reasonable.

The advent of the Apple IIc may have been welcomed by CP/M competitors. As far as Apple was concerned the IIc was a closed box. The corporate attitude was "you won't need to open it because it's got everything built in".

In the case of the disc controller and the serial card which many Apple users had added to their II+ or IIe, this was certainly the case. However, Apple didn't include a Z80 processor - after all, Apples are 6502 machines, aren't they?

Cirtech's IIc CP/M system

CLIFF McKNIGHT reviews the Cirtech IIc CP/M Plus system

might fly in the face of Apple policy in that it involves opening the box (and voiding the warranty, of course, if you do the job yourself). On the other hand it gives to Apple IIc users what many thousands of other II family users have found useful - the ability to run the ubiquitous CP/M-based software.

The basis of the system is a small pcb containing a Z80 processor and a few other chips. In fact, the processor is a Z80H which runs at 8MHz and speeds things up considerably.

Fitting the board involves removing the IIc's 65C02, plugging it into a socket on the Z80 board and then plugging

the board into the 65C02's socket on the motherboard.

The manual strongly recommends that you take your IIc and Z80 module to your official Apple dealer and have him fit them together. This way your warranty on the machine is protected. However, the manual also gives complete instructions for those who wish to do the job themselves.

In fact for an old Apple II fan like me who's used to delving about inside, the Cirtech manual is worth its weight in gold because it actually tells you how to open a IIc. Without explicit instructions it would be very easy to damage the many little

plastic clips which hold the thing together.

Opening the case is the hard part. Fitting the module is a piece of cake. Apart from the obvious care with which you treat electronic bits and pieces, the only other major problem is the static sensitivity of components like the 65C02.

The manual recommends that you carry out the procedure on an anti-static mat and that you discharge yourself regularly by touching the power supply box. Not having the former, I relied on the latter without any apparent problems.

Once the module is in place you must lift the disc drive up so that a clip can be attached to one of the chips on the motherboard. Then it's just a case of putting it all back together and testing it.

The whole operation took me about 45 minutes including reading the appropriate section of the manual beforehand.

The manual was perfectly clear, although at one point it tells you that the *previous* two operations could prove fatal to your IIc. It might have been better for this warning to precede the description of the dangerous bits.

The other half of the system is the supplied software, two discs which contain the CP/M Plus operating system and various utilities. Although CP/M Plus is licensed from Digital Research, the supplied version is written specifically for the IIc.

If you've got a IIc and don't know what to do with the mouse since you bored with Mousepaint, CP/M might be an

HELP UTILITY V1.1

At "HELP" enter topic (,subtopic)...

EXAMPLE: HELP> DIR EXAMPLES

Topics available:

APPLEIIc	COMMANDS	CNTRLCHARS	DATE	DEVICE	DIR
DISKS	DUPL	ED	ERASE	FILESPEC	GET
HELP	INITDIR	MOUSE	MOUSEKEY	PATCH	PIP (COPY)
PUT	RENAME	SDT	SET	SETDEF	SHOW
SUBMIT	TOOLKEY	TYPE	USER		

HELP>

Help is available on a wide range of topics

answer, albeit an unlikely one. The supplied version does more for the mouse than Apple has, allowing you to enter characters by pointing at them.

This support for the mouse is important because it occurs at the system level, not the software level. It means that you can use the mouse in Wordstar, even though Wordstar doesn't know a mouse from a piece of cheese.

You can even enter Control characters using the mouse. If an inverse upper-case character is pointed at it will be entered as a Control character.

Since this is how programs like Wordstar display the menu options it means that such programs could be used by someone whose movement was too restricted for normal keyboard use.

If you're a CP/M user of old, the error messages will surprise you. They actually tell you what's wrong in an intelligible way - most unlike CP/M! Wot,

no more "BDOS Error on A"? Certainly not.

There are other features too, like a disc-based help file system, the ability to print the current screen and so forth.

It's also possible to copy and format discs from within a program. How many times have you needed to do that? Lots, if you're like me!

So the module is easy to fit and the system is powerful and friendly... but will it run the software?

The answer is certainly "yes". With one small exception, everything I tried worked fine, including Wordstar, dBase II and MBasic. There is a patch supplied for MBasic, which would otherwise have given problems, but the patch worked fine.

It felt a bit strange using Wordstar on a IIc at first, but it was nothing I couldn't get used to.

The one small exception was an earlier version of CP/M. Just

to be perverse, I dug out a 2.20 version and tried that. It didn't work, but it was the only thing that didn't.

If I've given the impression that my IIc is now a dedicated CP/M machine, perish the thought. The beauty of the Cirtech system is that it adds to the IIc without detracting anything.

From a cold boot everything still works as it did before, including DOS 3.3, ProDOS, Pascal, Fortran, Logo and all the games I've played since fitting the module. It even ran BBC Basic, but that's another story which I'll leave Max Parrott to tell. However, boot a CP/M disc from cold and the Z80 springs into life. It's the best of both worlds.

I was supplied with just the module and operating system for review, but Cirtech are now offering a bundle which includes Pocket Wordstar/Mailmerge for only £280. When you think of what Wordstar alone would

normally cost, that's got to be a bargain.

Also available is a Programmer's Pack containing a full set of programming utilities - macro assembler, symbolic debugger and so forth - at a cost of £89. Alternatively, you can buy just the hardware module for £86.

I know quite a few people who showed interest in the IIc when it was launched, but stuck to their IIe because they didn't want to give up CP/M. Now those people have a real choice, and Apples can continue to be the world's most popular CP/M machine.

*Product: CP/M Plus System (hardware and software)
Price: £195
Distributor: Cirtech (UK), Currie Road Industrial Estate, Galashiels, Selkirkshire, Scotland TD1 2BP.
Tel: 0896 57790.*

Highest Quality Lowest Prices
Free Delivery

Unbranded 5 1/4" Disks from 72p

A range of unbranded floppy disks supplied in boxes of 10, with envelopes, labels and write-protect tabs. Made by well-known manufacturers and supplied with full replacement guarantee.

Prices per box of 10 (excl VAT)	1-4	5-9	10-24
SS/DD 48 tpi	11.90	11.20	10.80
DS/DD 48 tpi	14.90	13.90	13.40
DS/DD 96 tpi	19.90	18.90	17.90

C.D.C. 'StorageMaster' 5 1/4" disks

Prices per box of 10 (excl VAT)	1-4	5-9	10-24
SS/DD 48 tpi	11.90	11.20	10.80
DS/DD 48 tpi	14.90	13.90	13.40
DS/DD 96 tpi	19.90	18.90	17.90

C.D.C. 'StorageMaster' 3 1/2" disks

SS 135 tpi	25.50	23.90	22.60
DS 135 tpi	34.90	33.50	32.70

Wabash 'DataTech' 5 1/4" Disks
The new range from Wabash, one of the best known names in floppy disks

Wabash 'DataTech' 5 1/4" disks

Prices per box of 10 (excl VAT)	1-4	5-9	10-24
SS/DD 48 tpi	14.90	13.90	13.40
SS/DD 48 tpi	16.90	15.90	15.40
DS/DD 48 tpi	18.90	17.90	17.30
SS/DD 96 tpi	22.90	21.90	21.20
DS/DD 96 tpi	25.90	24.90	24.20

Wabash 'DataTech' 3 1/2" disks

SS 135 tpi	26.50	24.90	23.10
DS 135 tpi	35.90	34.50	33.70

Fuji 3 1/2" disks

Prices per box of 10 (excl VAT)	1-4	5-9	10-24
MF1DD Single Sided 135 tpi	27.90	25.90	23.90
MF2DD Double Sided 135 tpi	36.90	34.90	33.90

See 10 3 1/2" Library case each 2.30 2.05 1.95
See 10 5 1/4" Library case each 2.30 2.05 1.95
5 1/4" Economy 50 Stg Case 7.95 6.95 5.95

1. All prices include 2nd Class delivery (U.K. mainland) but exclude VAT. For urgent deliveries or outside U.K. mainland, please telephone for delivery charges.
 2. Telephone orders can be accepted from Government bodies, schools, etc, or with a VISA card.
 3. Send cheque made payable to "IDS Computer Supplies", with order to the address below. Don't forget the V.A.T.

IDS Computer Supplies (0908) 569655

15 Darin Court, Crown Hill, Milton Keynes MK8 0AD

II+ IIe
IIc

fulltext

55/80

WORD PROCESSOR MAIL MERGE/SORT SPELLING CHECKER (optional)

- Now . . . Fulltext 55/80 version 4 . . . even easier to use, even more powerful.
- New! The fulltext integrated Spelling Checker with 45,000 word expandable English dictionary.
- The complete word-processing system!

- Gives 55 characters per line on screen and upper and lower case even on an Apple II+
- Uses 80-column card if present to give 80 characters per line.
- Fully compatible with IIe, IIc keyboards and shiftkey mod. Compensates for missing features of II+
- Type-ahead buffer. Preview mode. Help file.
- Built-in Mailmerge/Mailsort/Word-sort.
- Powerful Calculator!
- All menu/prompt-line driven using key-letters (L for Load, E for Edit, F for Find, etc).
- Enjoy the full word-processing Editor.
- And printer control sequences are easily inserted into your text without upsetting the formatting.
- Advanced formatting options: page numbering, page headings, indentation, justification, definitions, auto-idents, auto-numbering of (sub-) sections and (sub-) paragraphs (1, 2, 3, etc or a) b) c) etc).
- Requires 64K memory. DOS based.
- Professional 160-page manual with full descriptions. Disk includes SELF-TEACH files for fast learning, and other utilities.
- Owners are offered upgrades to new releases.
- Foreign language characters can be viewed on screen from the 55-column version.
- Simple to learn; powerful to use. The best Apple software deal around!

Fulltext 55/80 £50, with Spelling Checker £75

SPACIFIC SOFTWARE

P.O. Box 58, Morpeth, Northumberland, NE61 1EQ

HAVE you ever wondered how Apple word processors stack up against those used by the MS-DOS and CP/M-80 crowd?

The other day I was reading a magazine review of NewWord, a new MS-DOS/CP/M-80 word processor. NewWord was written by some of the original Wordstar programmers, and is apparently a considerable improvement over that cumbersome beast.

I was intrigued by benchmark comparisons, based on a 4,000 word text file, applied to the program. How would an Apple II program, running on the much slower 8 bit 6502 processor, compare?

I checked it out with AppleWorks and Apple Writer II, both running under ProDOS on a 128k machine. Testing DOS 3.3 word processors was trickier.

Few if any DOS 3.3 programs can handle files as big as 4,000 words, as they are generally designed for 48k or 64k Apples. Of those that I own, the Australian program Sandy IIe managed the most — 3,600 words.

So I used that and adjusted the timings upwards on a pro rata basis. Probably not entirely accurate, but surely not far wrong.

The test compared times for loading, saving, searching for the last word in the article, and manually scrolling from top to bottom. The magazine tests were on the IBM PC, PCjr, Compaq and Morrow MD-11, and presumably the results were averaged. All those machines should theoretically be much faster than the Apple.

Here are the results, timed in seconds:

	Doc. load	Doc. save	Search	Scroll
AppleWorks	21.8	31.8	2.8	33.7
Apple Writer II (ProDOS)	18.1	32.4	7.0	32.2
Sandy IIe	7.2	20.3	2.2	33.8
NewWord	19.8	23.2	10.8	1.21.5
Wordstar	9.9	24.9	10.5	41.2
Volkswriter Deluxe	22.0	21.9	7.0	30.1

WIP v WIP

Apples hold their own against the big boys

All three Apple word processors did remarkably well in comparison with their big brothers. One, Sandy IIe, beat all comers by a wide margin. It may be no coincidence that its writer, Sandy Donald, was inventor of FastDOS, the first speeded up DOS for the Apple II.

Considering it is not a stand-alone word processor, AppleWorks did remarkably well overall, and ran rings round the 16 bit machine word processors in the search test.

Actually it walks over the others in the scroll test too, if you do the sensible thing and speed up the down arrow by simultaneous use of the open Apple key. That way I zapped manually from top to bottom in 5.1 seconds.

Apple Writer II scored very close to AppleWorks, except that it was a little faster at loading files, and noticeably slower in the search test.

Incidentally, when reading

through the NewWord review I was struck by the number of "improvements" over Wordstar that I already take for granted in my Apple II word processors.

Sixteen bits and CP/M-80 ain't necessarily best for every-

thing. The 6502 chip may be slow by today's standards, but any computer is only as good as its weakest link. Or perhaps, in the case of the venerable Apple II, its strongest link — software.

John MacGibbon

appletip

t In Turbo Pascal the Boolean function KeyPressed will not always function quite as intended. This program illustrates the problem:

```
Program test;
begin
  repeat
    writeln('Hello');
  until KeyPressed;
  if KeyPressed then writeln('KeyPressed still active');
end.
```

This is easily cured by inserting a writeln; before the if statement.

Max Parrott

Immediate mode DOS commands

t To use DOS commands in immediate mode which would usually only be accessible from deferred mode, go into the monitor (CALL -151) and enter:

```
A021: EA EA EA
```

You can restore things to normal by entering:

```
A021: 20 5E A6
```

Brian Hennessy

LAST month we looked at examples of the use of the assembler supplied with the Apple Pascal system, and examined just one example of the way in which data can be passed between the Pascal calling program and the assembly language routine.

Now we will consider this question a little more closely, for I have found that parameter passing is the main source of confusion to those using the assembler for the first time, particularly if they have previously used another assembler.

Perhaps we had first better define what we mean by a parameter. It is a variable, for example an integer, a real number or an array, which is passed between a calling program and a sub-program.

In Pascal they may be passed either by value, in which case the sub-program cannot change them, or as variables, in which case they can be changed. Obviously any values calculated by the sub-program must be variable parameters.

In Pascal a procedure with two integer parameters passed by value might be declared thus:

```
procedure demo1(xcoord,ycoord:integer);
```

If the procedure might change the values, then they would be declared like this:

```
procedure demo1(var xcoord,yccord:integer);
```

As we shall see later, whether a parameter is passed by value or as a variable affects the manner in which it is passed, and the way in which the assembly language routine must handle it.

The important concept to grasp is that of the stack. This is a data structure in which the last item to be added is the first one to be removed. Hence it is often called a LIFO, or Last-In-First-Out structure. Visualise it as being like a pile of plates – only the last plate to be added can be removed safely – and you won't go far wrong.

In common with many systems, Apple Pascal passes parameters using a stack. The Pascal calling program pushes all the parameters onto the stack, just like piling plates, and then pushes the return address. This is the location in the program to which the assembly language routine must go when it has completed its task.

Any assembly language routine must first unstack the return address (take the top plate) and save it, and then pull off the parameters. When it has finished its last act will be to push the return address back on to the stack, so that the RTS (return from subroutine) instruction will be able to pull the address and hence know how to get back to Pascal.

We shall now look at each standard data type in turn, and see how Pascal puts them on to the stack, and how our assembly

Passing the parameters can confuse

language programs should handle them. For each piece of assembly language code, the Pascal statement which calls the routine is shown as a comment.

Before calling the assembly language routine, the system puts integer variables onto the stack. For example, a routine with three integer parameters would look like Figure 1.

In this example the assembly language routine could not change the values of variables passed to it. If we want to do so, they are declared as VAR parameters and the system puts onto the stack the addresses of the variables, rather than the variables themselves (see Figure II).

Although a whole word is used to store a single boolean value, only the least significant bit contains valid data. A 1 is used to store true, a 0 for false. A typical routine which handles a boolean parameter is given on pages 147-8 of the Pascal Operating System manual.

In Apple Pascal long integers are stored in the following format:

word:	0	1		2,3,4,5.. ..10	
byte:		lb	hb	lb	hb
content:	length	sign	not used	2 most sig digits	2 least sig digits

The length (byte 0) is the number of digits in the integer. For example, an integer declared as integer[12], and holding the

TEMP1 .EQU 0 ;temporary storage locations	POP RETURN ;save return address
TEMP2 .EQU 2 ;in the first page of memory	POP TEMP3 ;get parameters:
TEMP3 .EQU 4	POP TEMP2 ;NB pulled of stack in opposite order to the
RETURN .EQU 6 ;temporary store for return address.	POP TEMP1 ;Pascal declaration.
.MACRO POP ;a macro is a shorthand way of repeating	
PLA ;the same sequence of instructions in a	
STA %1 ;number of different places.	
PLA ;Each time POP appears in the main program,	
STA %1+1 ;it will be replaced by this sequence of	
.ENDM ;instructions - see Pascal O/S Manual p 143	
;The routine below is called by the following Pascal statements,	
; (the variable names need not be the same in the Pascal &	
; assembly language programs.)	
;procedure demo1(temp1,temp2,temp3:integer);	
;external;	
.PROC DEMO1,3 ;the '3' indicates the number of parameters.	
	;the main body of the routine would be located here
	;at this point, the routine may make free use of TEMP1, TEMP2 & TEMP
	;nb if any of them are changed, the variables in the Pascal program
	;will not be affected.
	;now, restore the return address
	LDA RETURN+1
	PHA ;put it on the stack
	LDA RETURN
	PHA
	;and return back to Pascal.
	RTS
	.END ;end of the assembly

Figure 1

```

RETURN .EQU 0
TEMP1 .EQU 2

;the procedure would be called thus:
;procedure demo2(var param:integer);

.PROC DEMO2,1 ;one word of parameters
POP RETURN ;(assume macro defined as in DEMO1)
POP TEMP1 ;location 2 contains ADDRESS of variable

;to access the variable, use something like:
LDY #0
LDA @TEMP1,Y ;acc now holds data

;before returning, make sure the variable is updated:
STA @TEMP1,Y

LDA RETURN+1
PHA
LDA RETURN
PHA
RTS
.END
    
```

Figure II

value 123456789012, would be stored thus:

word:	0	1	2	3	4
byte:			lb hb	lb hb	lb hb
content:	12	0	12 34	56 78	90 12

The digits are stored in binary coded decimal, that is each decimal digit is stored using four bits, hence four digits are stored in each word. To access long integers using assembly language, we could use something like Figure III.

Whereas it is perfectly feasible to put small parameters on to the stack itself, we can't do this with large amounts of data such as arrays. The problem is that the 6502 processor only allows the stack to be 256 bytes long. (Z80 fans will point out that their processor allows a 64k stack.) Hence for arrays the parameter passed is a pointer to the array itself, that is it holds the address of the array. This happens with all arrays, not just those declared as VAR parameters.

```

RETURN .EQU 0
LENGTH .EQU 2
SIGN .EQU 4
DIGITS .EQU 6

;This procedure called by:
;type longint=integer[36];
;procedure demo3(parlong:longint);

.PROC DEMO3,11 ;11 words of parameters
; (a long integer)
PULL RETURN ;(assuming macro defined as above)
PULL LENGTH
PULL SIGN
LDA LENGTH
ASL A ;multiply by 2 to give length in bytes,
TAX ;copy length into X reg
NEXTDIG PLA ;get 2 digits
STA DIGITS,X
DEX ;decrement byte count
BNE NEXTDIG ;if not zero, get next digit

;all digits now in locations 6..14
;rest of routine follows.....
    
```

Figure III

```

RETURN .EQU 0
ARRPTR .EQU 2

;this procedure called by
;type pactyp = packed array[0..255] of 0..255;
;procedure demo4(pac:pactyp);

.PROC DEMO4,1
PULL RETURN ;assume macro PULL defined as above
PULL ARRPTR ;get pointer to the array.

;to access all 256 elements in turn,
;we could use something like:
LDX #0
NEXTEL LDA @ARRPTR,X ;acc now contains Xth element

;now do required operations.

STA @ARRPTR,X
INX ;move on to next element
BNE NEXTEL ;if not finished

;push return address & return to Pascal, as above.
    
```

Figure IV

If we wanted to write an assembly language routine to work with a packed array of 256 bytes, then it might look like Figure IV.

Strings are stored as packed arrays of chars. The "zereth" element contains the length and the remaining bytes contain the characters of the string. The required code will look rather like that required for long integers, but remember that the second byte of the first word contains the first character. The interested reader might like to try writing an assembly routine to convert all lower case characters in a string to upper case.

Because of the complexity of the way in which real numbers are stored, any operation on reals using assembly language is a most difficult task. An indication of this is that about 1200 bytes of the Apple Pascal p-code interpreter are taken up solely by the floating point operations. Real numbers are stored in this format:

	word 1		word 2
BIT NUMBER:	31	30..23	22..16 15..0
CONTENT:	sign	exponent	mantissa mantissa

The mantissa and exponent are the two halves of any real number, the first being the 'numbers', the second the 'size', for example, in the number 6.32×10^8 , the mantissa is 6.32, the exponent is 8. In Apple Pascal, of course, the numbers are stored in binary, making operations even more complex. In short, leave well alone.

While we have seen that the handling of real numbers with assembly language can be quite tricky, other types of data need cause us no problems. The key to understanding parameter passing is the idea of a stack. Once we can visualise that assembly language and Pascal programs can share data with ease.

There's no specific demonstration program this month. I hope that the pieces of code given above will inspire you to play around with the Apple Pascal Assembler, and get used to passing data between Pascal and 6502 assembly language routines. That way we can use the power of Pascal most of the time, but drop into machine code to handle hardware and when speed is really needed.

● Next month we look at some of the options available with the Apple Pascal Compiler, not least the ones that help us to write crash-proof programs.



THIS month's article in the &DOSFILE series of & routines for filing systems covers two routines. One serves to initialise data discs from within a program.

The other passes the names of files on a disc to a previously declared string array, thus easing the tasks of sorting file names, listing them perhaps in alphabetical order by file type, and generally processing disc catalogs within a program.

The initialisation routine needs the syntax:

&I(N)

where N is an integer between 1 and 254, and uses the DOS file manager routines to format a disc with a volume number N. Because the disc VTOC sector is then marked to show tracks 1 and 2 as free, more space is available for storing programs and data—a total of 8192 bytes.

The disc cannot be used for booting, however, since a complete copy of DOS is not stored.

Initialising data discs... and easing the task of file names sorting

PETER HARRIS continues his series on file maintenance commands

Track 0 is not marked as free, as it is not accessible to the file manager. Data can be stored on track 0 only by using RWTS routines, which make more demands on the programmer.

Before using the &I com-

mand it is necessary to establish in which slot and drive the disc to be initialised has been placed. This is done by POKING the appropriate value for the required slot into memory location 945, and for the

required drive into memory location 944:

**1000 POKE 945,6:
POKE 944,1**

for slot 6, drive 1. These memory locations are part of the &DOSFILE interface, not part of DOS itself.

The syntax for the cataloguing command is:

&F,N\$

where N\$ is a previously dimensioned string array — any legal string name may be used instead of N\$ — &F is used instead of &C, because the latter has already been used for the file close command. However F stands for Filename.

The routine reads the filenames found on a disc in the appropriate slot and drive, values previously POKED as above, into the string array.

Clearly there must be enough elements in the array to hold all the file names. The maximum number of files which can be

```

SOURCE FILE: DOSFILE
9252: 429 LST ON
9252: 430 *****
9252: 431 *
9252: 432 * DOSFILE PART 2. TO BE TYPED IN AFTER
9252: 433 * LINE 426 OF PART 1
9252: 434 *
9252: 435 *****
9252: 437 *****
9252: 438 *
9252: 439 * &I(N) INITIALISE NEW DATA DISC
9252: 440 * TO VOLUME N
9252: 441 *
9252: 442 * &F,A$ LOAD DISC CATALOGUE INTO ARRAY A$( )
9252: 443 *
9252: 444 *****
9252:20 16 92 446 INIT JSR DECHX
9255:A5 50 447 LDA LINNUM
9257:80 AF 03 448 STA VOL
925A:20 5F 90 449 JSR FPL
925D:A9 9D 450 LDA #9D
925F:A0 01 451 LDY #1
9261:91 00 452 STA (PTR),Y
9263:A9 08 453 LDA #40B
9265:20 E0 90 454 JSR V1
9268:D0 15 455 BNE ENDINIT
926A:A9 FF 456 LDA #FF
926C:80 F7 B3 457 STA #B3F7 MARK
926F:80 FB B3 458 STA #B3FB TRACKS 1-2
9272:80 FB B3 459 STA #B3FB AS
9275:80 FC B3 460 STA #B3FC UNUSED
9278:20 FB AF 461 JSR WRVTOC USES RWTS
927B:A9 00 462 LDA #0
927D:85 48 463 STA #48 NEEDED AFTER USING RWTS
927F:60 464 ENDINIT RTS
9280:20 BE DE 466 CATALOG JSR CHKCOM
9283: 467 * FIND ADDRESS OF ARRAY DESCRIPTOR AND OFFSET TC
9283: 468 * FIRST ELEMENT, AND SAVE RESULT IN PTR
9283:20 D9 F7 469 JSR GETARYPT
9284:18 470 CLC
9287:A0 04 471 LDY #4
9289:81 98 472 LDA (LOWTR),Y
928B:0A 473 ASL A
928C:65 98 474 ADC LOWTR
928E:67 05 475 ADC #5
9290:85 00 476 STA PTR
9292:A5 9C 477 LDA LOWTR+1
9294:A9 00 478 ADC #0
9296:B5 01 479 STA PTR+1
9298: 481 * PREPARE TO READ VTOC
9298:A9 11 482 LDA #11
929A:80 FA B5 483 STA #B5FA
929D:A9 FF 484 LDA #FF
929F:80 F9 B5 485 STA #B5F9
92A2:A0 B1 03 486 LDA SLOT
92A5:0A 487 ASL A
92A6:0A 488 ASL A
92A7:0A 489 ASL A
92A8:0A 490 ASL A
92A9:8D F7 B5 491 STA #B5F7
92AC:A0 B0 03 492 LDA DRIVE
92AF:8D FB B5 493 STA #B5FB
92B2:A9 60 494 LDA #60
92B4:8D 9E B0 495 STA #B09E SUPPRESS ERROR ROUTINE
92B7:20 F7 AF 497 JSR RDVTOC USES RWTS
92BA:10 B0 498 LDA #B0
92BC:8D 9E B0 499 STA #B09E
92BF:A0 00 500 LDY #0
92C1:84 48 501 STY #48 NEEDED AFTER USING RWTS
92C3:84 04 502 STY TEMP
92C5:A0 F5 B7 503 LDA #B7F5 RWTS RETURN CODE
92C8:8D B5 03 504 STA RTCODE
92CB:F0 03 505 BEQ GOODCAT
92CD:4C DD FB 506 JMP BELL
92D0:20 5D 93 508 GOODCAT JSR INTADR ESTABLISHES TEMPORARY
92D3: 509 * BUFFER TO RECEIVE FILENAMES
92D3:18 510 CLC
92D4:20 11 B0 511 CAT8 JSR RDDIR
92D7:80 71 512 BC5 EXIT
92D9:A2 00 513 LDX #0
92DB:8E 9C B3 514 CAT7 STX #B39C
92DE:8D C6 B4 515 LDA #B4C6,X
92E1:F0 67 516 BEQ EXIT
92E3:30 5E 517 BHI NXFFILE FILE NOT LOCKED
92E5:A0 A0 518 LDY #A0
92E7:8D CB B4 519 LDA #B4CB,X
92EA:10 02 520 BPL CAT2
92EC:A0 AA 521 LDY #AA FILE LOCKED
92EE:98 522 CAT2 TYA
92EF:20 4F 93 523 JSR STORBUF
92F2:8D CB B4 524 LDA #B4CB,X
92F5:29 7F 525 AND #7F
92F7:A0 07 526 LDY #7
92F9:0A 527 ASL A
92FA:0A 528 CAT3 ASL A
92FB:8D 03 529 BCS CAT4
92FD:88 530 DEY
92FE:00 FA 531 ENE CAT3
9300:B9 A7 B3 532 CAT4 LDA #B3A7,Y FILE TYPE
9303:20 4F 93 533 JSR STORBUF
9306:A9 A0 534 LDA #A0
9308:20 4F 93 535 JSR STORBUF
930B:EB 536 INX
930C:EB 537 INX
930D:EB 538 INX
930E:A0 1D 539 LDY #1D
9310:8D C6 B4 540 CAT5 LDA #B4C6,X

```

Listing 1

held on a normal DOS disc is 105, and the routine reads the first file name into element 0 of the array, so the statement:

10 DIM N\$(104)

should cover all eventualities. Normally, of course, much smaller arrays should suffice, particularly if one already has some idea of the catalog size.

Each array element contains more than just the file name. The first character of the string may be a space, in which case the file is not locked, or an asterisk, in which case it is locked.

The second character indicates the type of file, just like a normal CATALOG command. The third character is always a space, and the file name proper begins only on the fourth character.

Listing I gives the assembly language and machine code necessary for these two routines. It is intended that these be typed in to follow the

previous &DOSFILE routines listed in last month's article. (Some of the dummy routines at the end of that listing should be removed.)

Listing II is for a disc formatting program, which may be used as a module within readers' own programs. It is obviously necessary that the disc you are about to initialise does not contain valuable data, so the program attempts to read a catalog with the &F command before proceeding with the &I command.

If no error is detected, then

obviously the disc has already been used to store DOS data.

N.B. This precaution will not protect you from erasing a CP/M or Pascal disc.

If the listing is to be incorporated into another program, then lines 160-240 must go with the program's own initialisation routines. The disc formatting module is then reached by GOSUB 260 (the module can obviously be relocated if required).

The listing seems rather long, but it comes with two hidden bonuses. The first is contained in lines 170-230, which serve to read the DOS error messages into the array ERMESS\$.

The values assigned to the variables OFFSET and TXT in line 170 are for 48k systems. These strings can then be used in error-handling routines.

In connection with ONERR GOTO and ONERRGOSUB commands, an index to the type of DOS error can be retrieved by:

X=PEEK(222)

The value in X can then be used to retrieve an error message from ERMESS\$(X). Similarly, an error when using &DOSFILE routines can be detected by PEEKing RTCODE (decimal 949), one of the memory locations in the &DOS-FILE interface, and using a non-zero value in the same way.

In this way DOS errors can be brought to the attention of a program user without the program crashing.

The second bonus is the "idiot-proof input" featured in lines 670-830. This was inspired by the "bullet-proof input" which comes in the program DISKIO supplied with Apple Pascal. It is not as

versatile as the Pascal procedure, as Basic does not allow sets as data structures, but it is very much better than many input routines.

It needs setting up as in line 680. A1 is the lowest Ascii value, and A2 is the highest Ascii value of the range of characters which will be accepted by the routine for incorporation in the output string F\$. The maximum length of F\$ is given by the value of MAX in line 680.

By suitably defining A1 and A2, it is possible to accept only digits, or only capital letters, or all characters apart from control characters, just as one pleases.

One other peculiarity of the listing deserves comment. Extensive use has been made of FOR <variable>=0 TO 1 ... NEXT loops in order to avoid using GOTO.

Listing III is another Basic program to demonstrate the utility of the &F command, together with the &O, &R and &C commands described in the previous article. It shows that these commands can be used to inspect not only text files, but also other file types.

The output of the program is a list of files available on a particular disc separated into types of file. The Applesoft and Integer titles are marked with the length of the program, and the binary files are marked with the length of the file, and its starting address when BLOADED.

The program makes use of the fact that these details are stored on the disc at the very beginning of each file, and can be read by the &DOSFILE routines into the RDBUF area. ● Next month I'll look at some space-saving ideas.

```

9313:20 4F 93 541 JSR STORBUF
9314:FB 542 INX
9317:BB 543 DEF
9318:10 F6 544 RPL CAT5
931A: 545 * REMOVE SURPLUS SPACES ON END OF STRING
931A:A0 1E 546 LDY #*1E
931C:BB 547 SHORT DEY
931D:B1 02 548 LDA (BUFF),Y
931F:C9 20 549 CMP #*20
9321:F0 F9 550 BEQ SHORT
9323:CB 551 INY
9324:9B 552 TYA ACC., HOLDS LENGTH

9325:A0 00 554 LDY #0
9327:84 04 555 STY TEMP
9329:91 00 556 STA (PTR),Y
932E:CB 557 INY
932C:A5 02 558 LDA BUFF
932E:91 00 559 STA (PTR),Y
9330:CB 560 INY
9331:A5 03 561 LDA BUFP+1
9333:91 00 562 STA (PTR),Y
9335:20 DE E2 563 JSR FRE
9338:18 564 CLC
9339:A5 00 565 LDA PTR
933B:A9 03 566 ADC #3
933D:B5 00 567 STA PTR
933F:90 02 568 BCC NXTFILE
9341:E6 01 569 INC PTR+1
9343:20 30 B2 570 NXTFILE JSR NXTDIR
9346:90 93 571 BCC CAT7
9348:E0 8A 572 RCL CAT8
934A:A9 00 573 EXIT LDA #0
934C:85 48 574 STA #48 NEEDED AFTER USING RWTS
934E:A0 575 RTS

934F:04 05 577 STORBUF STY TEMP+1
9351:A4 04 578 LDY TEMP
9353:29 7F 579 AND #*7F
9355:91 02 580 STA (BUFF),Y
9357:C8 581 INY
9358:84 04 582 STY TEMP
935A:A4 05 583 LDY TEMP+1
935C:A0 584 RTS

935D: 586 * ESTABLISH TEMPORARY BUFFER AT LOWER
935D: 587 * END OF FREE RAM
935D:A5 60 588 INTADR LDA STREND
935F:85 02 589 STA BUFP
9361:A5 6E 590 LDA STREND+1
9363:85 03 591 STA BUFP+1
9365:A0 592 RTS
    
```

*** SUCCESSFUL ASSEMBLY: NO ERRORS

Listing II

```

100 TEXT : HOME : HTAB 5:
    VTAB 5: INVERSE
110 PRINT "DISK
    INITIALISATION PROGRAM":
    NORMAL
115 D$ = CHR$(4): PRINT
    D$"BRUN DOSFILE.OBJ0"
118 PRINT D$"BLOAD
    DOSFILE.OBJ1"
120 GOSUB 160: REM
    INITIALISATION
130 GOSUB 260: REM INIT
    DATA DISK MODULE
140 END
150 REM INITIALISATION
160 DIM
    N(15),ERMESS$(15),CT$(104)
170 OFFSET = 43583:TXT =
    43377
180 FOR A = 1 TO 15:N(A) =
    PEEK (OFFSET + A): NEXT
190 FOR A = 1 TO 15:T = 0
200 FOR N = 0 TO 1
210 L = PEEK (TXT + N(A) +
    T):N = (L > 127)
220 ERMESS$(A) = ERMESS$(A)
    + CHR$(L - N * 128)
230 T = T + 1: NEXT : NEXT
240 SLOT = 945:DRIVE =
    944:RTCODE = 949: REM
    &DOSFILE INTERFACE
245 RETURN
250 REM START OF MAIN INIT
    MODULE LOOP
260 FOR Q = 0 TO 1
270 REM GET SLOT AND DRIVE
280 PRINT : PRINT TAB(
    20)"SLOT ":
290 FOR N = 0 TO 1: GET
    S$:S = VAL (S$)
300 N = (S > 0) AND (S < 7):
    NEXT
310 PRINT S: POKE SLOT,S
320 PRINT : PRINT TAB(
    19)"DRIVE ":
330 FOR N = 0 TO 1: GET
    D$:D = VAL (D$)
340 N = (D = 1) OR (D = 2):
    NEXT
350 PRINT D: POKE DRIVE,D:
    PRINT
360 REM GET VOLUME NUMBER
370 PRINT :VT = PEEK (37)
380 FOR N = 0 TO 1: HTAB
    11: VTAB VT: PRINT
    "VOLUME NUMBER ":; CALL
    - 868
390 GOSUB 680:V = VAL (F$)
400 N = (V > 0) AND (V <
    255): NEXT
410 REM CHECK DISK IS BLANK
420 FOR M = 0 TO 1:Y$ = "Y"
430 PRINT : PRINT : PRINT
    "PUT IN BLANK DISK AND
    PRESS (RETURN) ":
440 FOR N = 0 TO 1: GET
    R$:N = (R$ = CHR$(13)):
    NEXT : PRINT : REM WAIT
    FOR RETURN KEY
450 FOR N = 0 TO 104:CT$ =
    "": NEXT : REM ZERO
    ARRAY
460 & F,CT$
470 IF PEEK (RTCODE) = 0
    THEN GOSUB 540: REM
    DISK ALREADY IN USE
480 M = (Y$ = "Y"): NEXT :
    REM NEXT M - CHECK
    ANOTHER DISK
485 REM PROCEED IF OK
490 & I(V)
500 X = PEEK (RTCODE): IF X
    < > 0 THEN PRINT :
    INVERSE : PRINT TAB(
    5)ERMESS$(X): NORMAL :
    PRINT
510 PRINT : PRINT
    "INITIALISE ANOTHER DISK?
    (Y/N) ":; GOSUB 610
520 Q = (Y$ = "N"): NEXT :
    RETURN
530 REM DISK ALREADY IN USE
540 PRINT : HTAB 6: FLASH :
    PRINT "THIS DISK IS
    ALREADY IN USE": NORMAL
550 PRINT : PRINT "SHOW
    CATALOG? (Y/N) ":; GOSUB
    610
560 IF Y$ = "Y" THEN GOSUB
    640
570 PRINT : PRINT "GIVE UP?
    (Y/N) ":; GOSUB 610
580 IF Y$ = "Y" THEN POP :
    RETURN
590 PRINT : PRINT "ERASE
    THIS DISK? (Y/N) ":
600 REM YES/NO SUBROUTINE
610 FOR N = 0 TO 1: GET Y$
620 N = (Y$ = "Y") OR (Y$ =
    "N"): NEXT : PRINT Y$:
    RETURN
630 REM PRINT CATALOG
    ARRAY
640 FOR N = 0 TO 104
650 IF CT$(N) < > "" THEN
    PRINT CT$(N)
660 NEXT : RETURN
670 REM IDIOT-PROOF INPUT
    ROUTINE
680 A1 = 48:A2 = 57:MAX = 3
690 GOSUB 770: REM FIRST
    CHARACTER
700 FOR N = 0 TO 1
710 GET A$
720 IF (ASC (A$) = < A2
    AND ASC (A$) = > A1 AND
    LEN (F$) < MAX) THEN F$ =
    F$ + A$: PRINT A$:
730 IF A$ = CHR$(8) THEN
    GOSUB 750
740 N = (A$ = CHR$(13)):
    NEXT : RETURN
750 PRINT A$: " ";A$:
760 IF LEN (F$) < > 1
    THEN F$ = LEFT$ (F$, LEN
    (F$) - 1): RETURN
770 FOR M = 0 TO 1
780 GET A$
790 M1 = (ASC (A$) < = A2
    AND ASC (A$) > = A1)
800 M2 = (A$ = CHR$(13))
810 M = M1 OR M2: NEXT
820 IF A$ < > CHR$(13)
    THEN F$ = A$: PRINT A$:
    RETURN
830 F$ = "": POP : RETURN
128 PRINT D$"BLOAD
    DOSFILE.OBJ1"
130 DIM
    CT$(104),OUT$(104,4),TYPE(
    4)
140 DEF FN DE(A) = PEEK
    (A) + PEEK (A + 1) * 256
150 SLOT = 945:DRIVE =
    944:RECLN = 942:RDBUF =
    FN DE(954): REM &DOSFILE
    INTERFACE
160 VTAB 12: PRINT "CATALOG
    DISK IN:"
170 REM GET SLOT AND DRIVE
180 PRINT : PRINT TAB(
    20)"SLOT ":
190 FOR N = 0 TO 1: GET
    S$:S = VAL (S$)
200 N = (S > 0) AND (S < 7):
    NEXT
210 PRINT S: POKE SLOT,S
220 PRINT : PRINT TAB(
    19)"DRIVE ":
230 FOR N = 0 TO 1: GET
    D$:D = VAL (D$)
240 N = (D = 1) OR (D = 2):
    NEXT
250 PRINT D: POKE DRIVE,D:
    PRINT
260 REM
270 FOR N = 0 TO 4:TYPE(N)
    = 0: NEXT : REM
    INITIALISE FILE TYPE
    COUNTERS
280 & F,CT$
290 REM SORT FILES INTO
    CATEGORIES
300 FOR N = 0 TO 104
310 IF LEN (CT$(N)) < > 0
    THEN GOSUB 750
320 NEXT
330 POKE RECLN,4: & N(0):
    REM READ 1ST 4
    CHARACTERS ONLY
340 IF TYPE(1) THEN GOSUB
    400: REM APPLESOFT FILES
350 IF TYPE(2) THEN GOSUB
    410: REM INTEGER FILES
360 IF TYPE(3) THEN GOSUB
    550: REM BINARY FILES
370 IF TYPE(4) THEN GOSUB
    420: REM TEXT FILES
380 IF TYPE(0) THEN GOSUB

```

Listing III

```

100 TEXT : HOME : HTAB 6:
    VTAB 5: INVERSE
110 PRINT "EXTENDED CATALOG
    PROGRAM"
120 PRINT : PRINT "WITH
    FILE LENGTHS AND START
    ADDRESSES": NORMAL
125 D$ = CHR$(4): PRINT
    D$"BRUN DOSFILE.OBJ0"

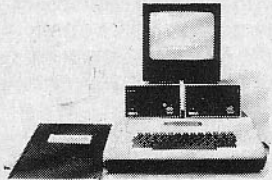
```



```

430: REM MISCELLANEOUS          PRINT T$;: HTAB 35: PRINT
FILES                          "LENGTH": NORMAL
390 TEXT : HOME : PRINT        490 POKE 34,1
"END OF CATALOG": END          500 FOR N = 0 TO TYPE(T) -
400 T$ = "APPLESOFT FILES":T   1
= 1: GOSUB 480: RETURN          510 GOSUB 700: REM LENGTH
410 T$ = "INTEGER FILES":T =   520 PRINT OUT$(N,T);: HTAB
2: GOSUB 480: RETURN           35: PRINT LE$
420 T$ = "TEXT FILES":T = 4:   530 NEXT : GOSUB 440:
GOSUB 660: RETURN              RETURN
430 T$ = "MISCELLANEOUS":T =   540 REM BINARY FILES
0: GOSUB 660: RETURN           550 TEXT : HOME : INVERSE :
440 PRINT : PRINT "PRESS       PRINT "BINARY FILES"
<RETURN> TO CONTINUE ";       560 PRINT "START":; HTAB
450 FOR N = 0 TO 1: GET A$     35: PRINT "LENGTH":
460 N = (A$ = CHR$(13)):       NORMAL
NEXT : RETURN                  570 POKE 34,2
470 REM APPLESOFT/INTEGER     580 FOR N = 0 TO TYPE(3) -
FILES                          1
480 TEXT : HOME : INVERSE :   590 T = 3: GOSUB 700: REM
                                LE$ GIVES START, NOT
                                LENGTH, OF BIN FILE
600 PRINT OUT$(N,3): PRINT    610 L = FN DE(RDBUF + 2)
LE$:                             620 L$ = RIGHT$ (" " +
610 L = FN DE(RDBUF + 2)      STR$(L),5)
620 L$ = RIGHT$ (" " +
STR$(L),5)
630 HTAB 35: PRINT L$
640 NEXT : GOSUB 440:
RETURN
650 REM TEXT/MISCELLANEOUS
FILES
660 TEXT : HOME : INVERSE :   670 FOR N = 0 TO TYPE(T) -
PRINT T$: POKE 34,1:         1
NORMAL
680 PRINT OUT$(N,T): NEXT :   GOSUB 440: RETURN
690 REM BASIC FILE
                                LENGTH/BINARY FILE START
700 L = LEN (OUT$(N,T)) -
3:0$ = RIGHT$
(OUT$(N,T),L)
710 & O(0$): & R: & C
720 LE = FN DE(RDBUF)
730 LE$ = RIGHT$ (" " +
STR$(LE),5): RETURN
740 REM FILE SORTING
750 TYPE$ = MID$
(CT$(N),2,1): REM TYPE
INDICATOR LETTER
760 T = (TYPE$ = "A") +
(TYPE$ = "I") * 2 +
(TYPE$ = "B" * 3 + (TYPE$
= "T") * 4: REM CONVERT
TYPES A..T INTO NOS. 1..4
770 OUT$(TYPE(T),T) = CT$(N)
780 TYPE(T) = TYPE(T) + 1:
RETURN
    
```

DATA LOGGER AND VIDEO DIGITISER FOR YOUR APPLE II



"LABPACK"
Data Acquisition, Processing and Graphics package from Stem Computing

LABPACK is a complete data logging and processing system for the Apple II. It uses a 12-bit A/D converter, with programmable gains for each of 16 channels, input parameters (channel selection, sampling rate etc) are software selectable. All logged data are stored in files on disc and can be edited, mathematically manipulated and plotted, using the exceptional graphics facilities. Also FFT, PSD, Statistics, Numerical Integration and Differentiation all incorporated into LABPACK's menu operated system. No programming knowledge is required. Comprehensive operating manual is included. Ideal for laboratory use. **Price: £590**



"COMPUTEREYES"
PAL version of the popular video digitiser by Digital Vision.

COMPUTEREYES digitises from any video source including video recorder and video disc, into the Apple's Hi-Res screen. Images may be stored on disc or screen dumped to a printer. The menu driven software also included 4 and 8 level simulated grey scale image capture routines. Additional optional enhancement software is available for image compatibility with Printshop and Newsroom. Also a Double Hi-Res option for the Apple IIe. **Price: £139**

Stem Computing

3 Blackness Avenue, Dundee DD2 1ER. Tel: 0382 65113

APPLE

Best in the North West
NEW LOW MACINTOSH PRICES

MACINTOSH

Macintosh 512k with free Macwrite/paint	£1,595.00
Macintosh 128k with free Macwrite/paint	£1,345.00
Macintosh External Drive	£275.00
Macintosh Numeric Keypad	£55.00
Macintosh Carry case Deluxe	£55.00
Macintosh Disks (10) Apple	£30.00
Ormbeta Compact Accounting	£275.00
Ormbeta Compact Accounting Plus	£550.00
Ormbeta Invoicer	£85.00
Filevision	£99.00
Microsoft Basic	£105.00
Microsoft Multiplan	£155.00
Microsoft Word	£140.00
Macdraw	£89.00
Mainstreet Filer	£79.00
MacTerminal	£55.00
PFS File & Report	£175.00
DB Master	£135.00
Lotus JAZZ	£350.00
Omnis 3	£350.00
Systematics 11 Accounting Ledgers, each	£195.00
Systematics Payroll	£195.00
Laserwriter	Ring for Price
Imagewriter 10"	£295.00

*- Access and Visa Cards accepted -
We also operate the Special 'Apple in Education Schemes'
Full Support and Training Available*

holdens

COMPUTER SERVICES

85 Fishergate, Preston PR1 2NJ. Tel: (0772) 561321

Colourful displays

DISPLAYS of up to 16 colours on a television set or colour monitor are made possible for the Apple II+ and IIe with a new Peanut Computer product.

Called the Colour Display Card it provides two outputs – a PAL compatible signal for colour television and a composite signal for a colour monitor.

Aimed at serious games enthusiasts the card costs £42.

Two joysticks have been launched by the company for use with the Apple II.

The first, price £13.95, is a simple self-centring unit designed for games use. It features two pairs of two fire buttons distinctively coloured and positioned.

Priced £19.50, the second joystick is more sophisticated with much smoother operation.

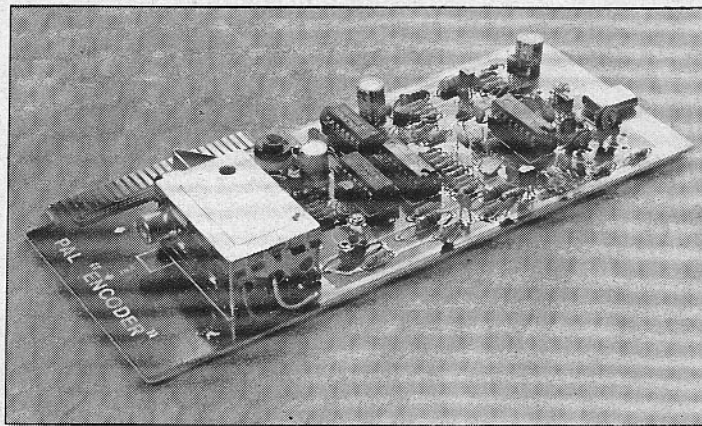
Intended for the professional working with graphics design software, or dedicated games enthusiast, it has optional self-centring and full trim controls.

An 80-column display card is also available. It is software compatible with the Videx Videoterm card and is recognised by all CP/M, Pascal and DOS 3.3 software.

A special feature is its automatic 40 or 80 column display switch. Price: £52.50.

Yet another offering from Peanut is a time-day-month-year clock cum calendar card designed for the Apple II+ and IIe.

Simply activated it provides four time-variable interrupts



Peanut colour display card

and, claims the company, has many programming possibilities.

The Clock Card, price £49, comes with a full manual and demo software.

● Peanut Computer, Low Mill, Dewsbury WF13 3LX. Tel: 0924-499366.

Reading tutor

A NEW way to teach reading is offered by Chatterbox, a voice reading ability drill from The Voice Connection.

It runs on Apple II computers and uses both voice output and voice recognition to train and drill youngsters in reading aloud.

A microphone/headset enables the student to hear the word in a natural voice through the Echo+ speech synthesiser while seeing it spelled out and illustrated on the monitor.

The voice recognition peripheral card IntroVoice then analyses the voice input and stores the voice pattern for each

student.

In the drill phase the student is presented with only the written word on the monitor and reads it aloud. The computer monitors the student's accuracy, correcting if necessary.

The individualised instruction provides the student with unlimited practice with corrections, yet does not require the amount of teacher supervision usual with individual tutoring.

Chatterbox offers more than 460 full colour graphic illustrations created by the Colorado Institute for Art. Price of Chatterbox is \$295, with Echo and speech synthesiser \$445, with IntroVoice I \$895 and with Echo and IntroVoice I \$995.

● The Voice Connection, 17835 Skypark Circle, Suite C, Irvine, California 92714. Tel: 0101 714 261 2366.

Protocol conversion

A COMMUNICATIONS product which enables the integration of text and data between Macintosh and IBM host has been released by Apple.

Called AppleLine, it is a protocol converter which enables the Macintosh user to emulate an IBM 3278 terminal.

It does this by translating protocol from a connected IBM 3274 cluster controller to an async VT100 form used by the Macintosh and MacTerminal.

Priced £1,000, the product is being aimed at large corporations who want to connect the Macintosh to an IBM host or who want IBM compatibility.

● Apple Computer, Eastman Way,

Hemel Hempstead, Herts. Tel: 0442 60244.

Tuneful tutor

DESIGNED to teach the reading of music through the use of an Apple II and a four or more octave Midi synthesiser keyboard, Note-Us is a sight reading series from Computers & ME.

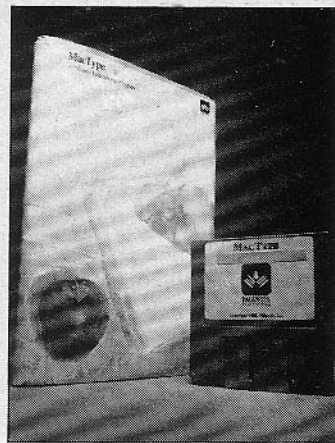
The software leads the student as rapidly as possible into creating pleasant sounds by challenging him to use both hands on the synthesiser keyboard.

The computer is used to display the music staff, notes and instructions to the student. It is also used to check the player's accuracy.

When a wrong note is hit the computer signals the student and asks for the proper one.

To reinforce the video screen display there are 25 original music examples in the software manual. The package costs \$100.

● Computers & ME, Ashbrook Road, Exeter, New Hampshire. Tel: 0101 603 772 4399.

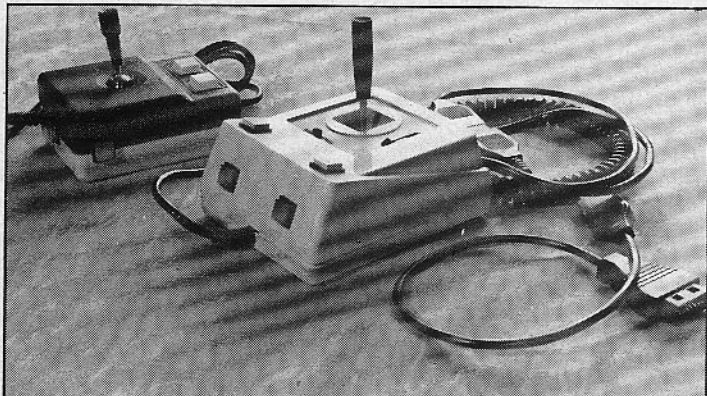


AN educational program for people who want to learn to touch-type is now available for the Macintosh from Palantis Software.

MacType is based on a study of real touch-typing courses and the experience of a former instructor.

It works in standard training cycles – review, drill and test. The drills contain practice text of words and sentence fragments that mix upper case, lower case and punctuation.

Features include: Advanced



Peanut joysticks

NEW PRODUCTS

placement – designed to allow rusty typists to brush up their skills – warmup, metronome – if you start typing erratically a metronome slows you down to an even pace – progress measurement, automatic record keeping, and the qwerty or Dvorak keyboard.

Price £49.

● *P&P Micro Distributors, Todd Hall Road, Carrs Industrial Estate, Haslingden, Rossendale, Lancs. BB4 5HU. Tel: 0706 217744.*

Tuck in a drive

HYPERDRIVE is a hard disc which fits inside the Macintosh case, with a piggyback circuit connecting to the machine's circuitry.

In addition to providing disc drive electronics, the added board can be configured with the 384k of memory necessary to bring a 128k Macintosh up to 512k.

Hyperdrive presents startup messages more than twice as fast as the normal Macintosh, says P&P Micro, and depending on how many files the user has left on the hard drive, boot time is usually half what it is on an unaltered machine.

The difference in loading application programs is said to be even more dramatic, allowing text entry only seven seconds after double-click on the MacWrite icon.

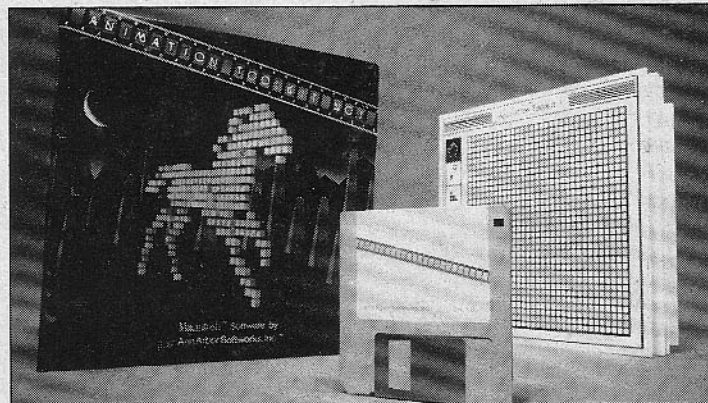
Going from MacWrite to MacPaint is said to take less than 15 seconds, rather than close to a minute as before. Price: £1,995.

● *P&P Micro Distributors, Todd Hall Road, Carrs Industrial Estate, Haslingden, Rossendale, Lancs BB4 5HU. Tel: 0706 217744.*

Apple Works for the II+

THANKS to Plus Works, a patch program from MGA MicroSystems, Apple II+ owners can join the ranks of AppleWorks users.

PlusWorks modifies the



Animation Toolkit 1

AppleWorks startup disc allowing it to operate on an Apple II, II+ or any Apple II workalike.

An extended memory feature is included which will expand the AppleWorks desktop. PlusWorks will recognise most popular peripheral slot RAM cards and use 100 per cent of their memory for desktop expansion up to 1mbyte.

For configurations of 256k and above, a dynamic RAM disc feature is included to minimise disc access.

There is a one-time configuration after which the altered startup disc, which remains copyable, will boot itself each time.

Users with full Ascii keyboard have the option to customise the keyboard driver, particularly beneficial to users with keyboard enhancers.

All AppleWorks functions are available from the keyboard, eliminating the need to use paddle buttons. Price is £49.95.

● *MGA MicroSystems, 140 High Street, Tenterden, Kent TN30 6HT. Tel: 05806 4278.*

Film editing

CREATIVE 128k Macintosh users can now create, edit and animate films with their computer.

This has been made possible by the Animation Toolkit 1.

The program's animation tools are used to bring still pictures, made using MacPaint,

to life. These tools include pencil, eraser, scroll, rotate, invert, distort, mask, matte, scale and flyby.

Individual frames are automatically combined into film clips and each sequence can then be edited. This is done by playing back clips in slow motion, cutting and adding new frames and clips or splicing.

Price: £48.20.

● *P&P Micro Distributors, Todd Hall Road, Carrs Industrial Estate, Haslingden, Rossendale, Lancs. BB4 5HU. Tel: 0706 217744.*

Run a company

PRODUCER of business simulation programs for Apple computers, Blue Chip Software has come up with a game which puts the player in charge of a robotics manufacturing company over a period of six years.

The player becomes chief executive officer with seven department heads reporting in the functions of sales, manufacturing engineering, production, research and development, materials management, quality assurance and finance.

Each department reports on its performance and the player can change a few critical business variables such as sales price. General industry trend data is also available in graphical format.

General economic as well as industry specific trends fluctuate independent of the

player's business decisions. These factors include GNP, inflation, interest rates, competitor's prices, industry demand, labour rates, raw material prices and lead times.

The challenge is to assimilate the vast amount of business data available, decide what – if any – changes are dictated, and implement them without having control over external business conditions.

● *Blue Chip Software, 6740 Eton Avenue, Canoga Park, California. Tel: 0101 818 346 0730.*

Expanded keyboard

A DETACHED keyboard from Peanut Computer offers Apple II+ users 24 function and seven user-definable keys.

Priced £99, it is also provided with principal DOS and CP/M commands, lower case Ascii code generator, cursor keys, Ctrl-Reset combination, separate numeric keypad area and optional key bleep.

● *Peanut Computer, Low Mill, Dewsbury WF13 3LX. Tel: 0924 499366.*

Files transfer

AN applications package MacLink, which allows communications between a Macintosh and an IBM PC for the transfer of files between the two, has been developed by DataViz.

For micros and applications requiring their own specialised file formats, MacLink provides a series of translators which convert files into the format required by applications on the receiving machine. All the translators work in both directions.

MacLink transfers text files or binary files and costs £119 including disc for both the PC and Mac and manual.

● *P&P Micro Distributors, Todd Hall Road, Carrs Industrial Estate, Haslingden, Rossendale, Lancs. BB4 5HU. Tel: 0706 217744.*

IN recent years many low priced Apple compatible slim drives have appeared and most are able to access up to 41 tracks, including T\$0.

However, the discs that are created by DOS on the Apple drives and on these slim drives are formatted for 35 tracks. Patches must be made to DOS before formatting the discs if the additional six tracks (24k) are to be used.

Having a collection of close to 200 discs at 24k extra storage per disc it would work out to a staggering 4.8mbyte, which is simply too much to ignore.

However, it would be an awesome task to reformat 200 discs and FID all the files across to the 41 track discs.

As such, I have written Disc Updater (Listing I) which enables me to update all the 35 track discs to 41 tracks without destroying any of the data already on them.

The program is relatively straightforward and uses RWTS to check and update the VTDC except for certain patches that are made to DOS to format the additional six tracks.

Users of this program must make sure that they do not have any data on tracks \$ST-\$ET (35-40) as all data will be erased. If your disc is one which already uses T\$23 - a very popular patch - then the variable ST in line 30 must be changed to reflect the correct start value (36).

Not all drives will be able to access up to track \$28 (40) and as such, the variable ET in line 30 must be changed to the maximum physical track that the drive can access so as to maximise your drive's ability.

If you are not sure, consult your drive manual under the specifications section or ask your dealer.

Alternatively, run the program in Listing II and it will tell you the maximum number of tracks that your drive can access reliably. Make sure the variables ST and ET in line 30 of Listing I reflect the correct values, save the program then test it on a duplicate disc to make sure it functions correctly before you use it on your precious discs.

Both Listing I and Listing II

Squeeze more tracks on to your discs

CHEE WEI-LI offers a disc updater which makes the most of your storage space

assume that they are running under a 48k DOS 3.3. If you are not, you must make some changes to the programs before they will run correctly.

One final point. If you own two or more disc drives, and especially if they are from different manufacturers, the drives may not be able to access the same number of tracks. In my case, one drive could access up to 43 tracks while the other

only 41.

It is then up to the reader to decide, by setting ST and ET of Listing I, line 30, whether they want to update to only 41 tracks so that all tracks can be read by all the drives or whether they want to maximise storage space at the expense that higher tracks can only be read by certain drives.

I would recommend maintaining compatibility to avoid

future complications, especially where the disparity of storage space is not great.

Editor's note: None of the drives we've tried would manage 40 tracks, but we've managed to get some extra tracks on all drives. Try it yourself and see, but make sure you use unimportant discs for your first attempts.

Listing I

```

10 GOSUB 630
20 CALL 760
30 ST = 35:ET = 40:VX =
  254:RWTS = 776:IF ET <
  ST THEN GOTO 660
40 A = PEEK (1) * 256 +
  PEEK (0):BUF = PEEK (A +
  9) * 256 + PEEK (A + 8)
50 HOME : PRINT "INSERT
  DISK FOR UPDATING IN SLT
  "; PEEK (A + 1) / 16;"
  DRV "; PEEK (A + 2);
  PRINT "TYPE (Q) TO QUIT
  ANY OTHER TO PROCEED";
  GET A#
60 IF A# = "Q" THEN GOTO
  650
70 HOME : PRINT "READING
  VTDC"
80 POKE A + 3,0: POKE A +
  4,17: POKE A + 5,0: POKE
  A + 12,1: POKE A + 13,0
90 CALL RWTS
100 IF PEEK (A + 13) < >
  0 THEN GOTO 500
110 IF PEEK (BUF + 52) =
  (ST) AND PEEK (BUF + 3)
  = 3 THEN GOTO 170
120 IF PEEK (BUF + 52) =
  ET + 1 AND PEEK (BUF +
  3) = 3 THEN GOTO 150
130 IF PEEK (BUF + 3) < >
  3 THEN GOTO 160
140 HOME : PRINT "NOT A
  STANDARD DOS 3.3 DISK";
  CALL - 756: RUN
150 HOME : PRINT "DISK
  ALREADY UPDATED"; CALL
  - 756: RUN
160 HOME : PRINT "NOT A DOS
  3.3 DISK"; CALL - 756:
  RUN
170 VX = PEEK (BUF + 6): IF
  VX = 0 THEN LET VX =
  PEEK (47)
180 ONERR GOTO 340
190 IF PEEK (48843) < >
  169 OR PEEK (48844) < >
  80 OR PEEK (48845) < >
  32 OR PEEK (48846) < >
  149 OR PEEK (48847) < >
  190 THEN GOTO 590
200 B = PEEK (48828):C =
  PEEK (48894)
210 POKE 48828,ST: POKE
  48894,ET + 1
220 POKE 48843,234: POKE
  48844,234: POKE
  48845,234: POKE
  48846,234: POKE 48847,234
230 HOME : PRINT
  "FORMATTING TRACKS
  ";ST;"-";ET
240 POKE (PEEK (A + 1) +
  49289),0: POKE (PEEK (A
  + 1) + 49293),0
250 IF PEEK (PEEK (A + 1)
  + 49294) > 127 THEN GOTO
  610
260 POKE A + 3,VX: POKE A +
  4,0: POKE A + 5,0: POKE A
  + 12,0: POKE A + 13,0:
  CALL RWTS
270 POKE A + 3,VX: POKE A +
  4,0: POKE A + 5,0: POKE A
  + 12,0: POKE A + 13,0:
  CALL RWTS
280 IF PEEK (A + 13) < >
  0 THEN GOTO 340

```

Here is a breakdown of the program in Listing 1:

- | | |
|---|---|
| <p>Line</p> <p>10-20 Place a short machine language routine at \$0300 and call it so that the address of the RWTS IOB is available to the Applesoft program.</p> <p>30 Sets up the variables of the first and last tracks to be initialised as well as the default volume number and the address of the short machine language routine which will call RWTS. Checks to make sure that the first and last tracks are valid.</p> <p>40 Calculates the address of the RWTS IOB and places it in variable A and also calculates the address of the buffer used by RWTS.</p> <p>80-90 Set up the IOB and call RWTS to read the VTOC.</p> <p>110-130 Check the VTOC to ensure that it is a DOS 3.3 disc and has only ST tracks on the disc before proceeding to format tracks ST-ET.</p> <p>170 Obtains the volume number of the disc for future use and places it in the variable V%.</p> <p>190 Ensures that the current DOS under which the program is running is what the program expects it to be - 48k DOS 3.3.</p> <p>200 Saves the values in DOS to the variables B and C.</p> | <p>210 Places the start (ST=35) and end (ET=40) +1 values of the tracks to format in DOS.</p> <p>220 Disables the recalibration of the disc arm to T\$00 by DOS before it performs the track initialisation. Forces DOS to start the track initialisation at the track which the drive head is currently on.</p> <p>240-250 Check the disc for the write protection tab and display an error message if there is.</p> <p>260-290 Performs a double seek to T\$00,S\$00 to simulate a recalibration followed by a seek to T\$ST where the track initialisation is to start.</p> <p>300 Calls RWTS to start the initialisation process starting from where the drive head is (T\$ST).</p> <p>390-400 Restore the patched DOS to its original values before updating the VTOC.</p> <p>410 Sets up the IOB and reads the VTOC.</p> <p>420 Checks to make sure that there are no read errors and that the VTOC indicates that the disc is formatted by DOS 3.3 and has ST tracks.</p> <p>430 Changes the VTOC in the buffer to indicate a ET+1 (41) track disc.</p> <p>440-480 Set VTOC bit map to indicate that tracks \$ST-\$ET are unused.</p> <p>500 Writes the modified VTOC back to the disc.</p> |
|---|---|

```

290 POKE A + 3,VX: POKE A +
  4,ST: POKE A + 5,0: POKE
  A + 12,0: POKE A + 13,0:
  CALL RWTS
300 POKE A + 3,VX: POKE A +
  4,ST: POKE A + 5,0: POKE
  A + 12,0: POKE A + 13,0:
  CALL RWTS
310 IF PEEK (A + 13) = 16
  THEN GOTO 610
320 IF PEEK (A + 13) < >
  0 THEN GOTO 340
330 GOTO 380
340 HOME : PRINT "ERROR IN
  FORMATTING": PRINT "<R>
  TO RETRY": PRINT "<Q> TO
  QUIT": GET A$
350 IF A$ = "R" THEN GOTO
  230
360 IF A$ = "Q" THEN GOTO
  620
370 GOTO 340
380 POKE 216,0: PRINT :
  PRINT : PRINT "UPDATING
  VTOC"
390 POKE 48828,B: POKE
  48894,C
400 POKE 48843,169: POKE
  48844,80: POKE 48845,32:
  POKE 48846,149: POKE
  48847,190
410 POKE A + 3,0: POKE A +
  4,17: POKE A + 5,0: POKE
  A + 12,1: POKE A + 13,0:
  CALL RWTS
420 IF PEEK (BUF + 52) <
  > (ST) OR PEEK (BUF + 3)
  < > 3 OR PEEK (A + 13)
  < > 0 THEN GOTO 600
430 POKE BUF + 52,ET + 1
440 J = BUF + 56: B = ST - 1
450 B = B + 1
460 IF B > ET THEN GOTO
  490
470 POKE J + B * 4,255:
  POKE J + 1 + (B * 4),255
480 GOTO 450
490 POKE A + 3,VX: POKE A +
  4,17: POKE A + 5,0: POKE
  A + 12,0: POKE A + 13,0:
  CALL RWTS
500 POKE A + 3,VX: POKE A +
  4,17: POKE A + 5,0: POKE
  A + 12,2: POKE A + 13,0:
  CALL RWTS
510 IF PEEK (A + 13) = 0
  OR PEEK (A + 13) = 2
  THEN GOTO 560
520 HOME : PRINT "ERROR IN
  WRITING VTOC": PRINT
  "T$11, S$00 MAY BE
  DAMAGED": PRINT "TYPE <R>
  TO RETRY WRITING": PRINT
  " <Q> TO QUIT": GET
  A$
530 IF A$ = "R" THEN GOTO
  490
540 IF A$ = "Q" THEN GOTO
  570
550 GOTO 520
560 PRINT : PRINT : PRINT
  "DONE": CALL - 756:
  HOME : RUN
570 HOME : PRINT "WARNING --
  VTOC (T$11,S$0) MAY BE
  DAMAGED": CALL - 756:
  RUN
580 TEXT : HOME : VTAB 12:
  HTAB 14: INVERSE : PRINT
  "READ ERROR": NORMAL :
  HTAB 9: PRINT "(NOT A DOS
  3.3 DISK?)": CALL -
  756: RUN
590 TEXT : HOME : PRINT "I
  CANNOT WORK WITH THIS DOS
  IN MEMORY": PRINT "PLEASE
  BOOT A STANDARD 48K DOS
  3.3 DISK": PRINT : PRINT
  : NEW : END
600 TEXT : HOME : INVERSE :
  PRINT "VTOC NOT FOUND":
  NORMAL : PRINT "VTOC WILL
  NOT BE UPDATED": CALL -
  756: RUN
610 POKE ( PEEK (A + 1) +
  49288),0: TEXT : HOME :
  PRINT "DISK IS
  WRITE-PROTECTED": PRINT
  "REMOVE WRITE-PROTECT
  TAB": CALL - 756
620 POKE 48828,B: POKE
  48894,C: POKE 48843,169:
  POKE 48844,80: POKE
  48845,32: POKE 48846,149:
  POKE 48847,190: RUN
630 FOR I = 768 TO 768 +

```

```

19: READ A: POKE I,A:
NEXT : RETURN
640 DATA
32,227,3,133,1,132,0,96,32
,227,3,32,217,3,169,0,133,
72,24,96
650 TEXT : HOME : NEW : END

660 TEXT : HOME : PRINT
"THE VARIABLES ST AND ET
IN LINE 30": PRINT "ARE
NOT SET UP PROPERLY":
PRINT "PLEASE SET UP
VARIABLES AND RERUN"
    
```

Listing 11

```

10 GOSUB 350
20 CALL 768
30 VX = 254:RWTS = 776
40 A = PEEK (1) * 256 +
PEEK (0):BUF = PEEK (A +
9) * 256 + PEEK (A + 8)
50 SI = PEEK (A + 1) /
16:DI = PEEK (A + 2)
60 HOME : PRINT "SLOT NO.
    
```

```

OF DISK: "; GET A$:
PRINT A$:SLT = VAL (A$)
70 IF SLT < = 0 OR SLT > 7
THEN GOTO 60
80 VTAB 3: HTAB 1: PRINT
"DRIVE NO. : "; GET A$:
PRINT A$
90 IF A$ < > "1" AND A$ <
> "2" THEN GOTO 80
100 DRV = VAL (A$): HOME
110 VTAB 10: HTAB 1: PRINT
"INSERT A BLANK DISK IN
SLOT ";SLT;" DRIVE ";DRV
120 VTAB 12: HTAB 11: PRINT
"HIT <ESC> TO ABORT":
VTAB 14: HTAB 8: PRINT
"ANY OTHER KEY TO
PROCEED": FLASH : VTAB
16: HTAB 7: PRINT "ALL
DATA WILL BE DESTROYED":;
NORMAL
130 GET A$
140 IF A$ = CHR$ (27) THEN
GOTO 370
150 IF SLT < > SI THEN
    
```

```

GOTO 170
160 GOTO 180
170 POKE A + 1,SLT * 16:
POKE A + 15,SI * 16
180 IF DRV < > DI THEN
GOTO 200
190 GOTO 210
200 POKE A + 2,DRV: POKE A
+ 16,DI
210 VTAB 18: HTAB 13: PRINT
"FORMATTING..."
220 B = PEEK (48894): POKE
48894,43
230 POKE A + 3,VX: POKE A +
4,0: POKE A + 5,0: POKE A
+ 12,4: POKE A + 13,0
240 CALL RWTS
250 IF PEEK (A + 13) < >
0 THEN GOTO 340
260 POKE 48894,B
270 VTAB 18: CALL - 950:
HTAB 11: PRINT "CHECKING
DRIVE..."
280 NT = 0: FOR J = 0 TO 42
290 POKE A + 3,VX: POKE A +
    
```

```

4,J: POKE A + 5,0: POKE A
+ 12,1: POKE A + 13,0
300 CALL RWTS
310 IF PEEK (A + 13) = 0
THEN NT = NT + 1
320 NEXT
330 VTAB 18: CALL - 950:
HTAB 2: PRINT "THIS DRIVE
IS ABLE TO READ ";NT;"
TRACKS": VTAB 19: HTAB
10: PRINT "(TRACK 0
INCLUSIVE)":; CALL -
756: RUN
340 HOME : VTAB 12: HTAB 8:
PRINT "ERROR IN
FORMATTING DISK":; CALL
- 756: RUN
350 FOR I = 768 TO 768 +
19: READ A: POKE I,A:
NEXT : RETURN
360 DATA
32,227,3,133,1,132,0,96,32
,227,3,32,217,3,169,0,133,
72,24,96
370 TEXT : HOME : NEW : END
    
```

apple classifieds

APPLE IIe duo disk drives (DOS 3.3) with controller, 9in monitor. Cards: parallel printer, super serial, 80 column. Manuals and software, £600 ono. Tel: Nick 021-565 2766, ext. 30 (day), 0905 640507 (evenings).

128k RAMCARD for Apple II, includes manual and Pseudodisk software (DOS, CP/M, Pascal), £90. Phone 051-336 4434.

EPSON MX82 PRINTER. Perfect condition, £140. Northampton (0604) 858395. Daytime ansaphone.

APPLE IIc with external drive added. Less than a year old, £650. Appleworks software package, £80. Tel: Steve (0722) 338711.

128k APPLE IIe, twin floppies, IIe monitor, Appleworks, Megaworks, Snapshot, manuals. Make offer. Tel: 07914 5498 (evens. after eight).

APPLEWRITER IIe DOS 3.3 version in original pack, £40. Northampton (0604) 858395 (daytime Ansaphone).

FOR SALE Apple IIc with monitor stand and Appleworks, £920. J.E.O. Stemmer, 14 Princes Street, Cardiff CF2 3PR.

FREELANCE OMNIS 3 programmer available to help with your project. Now Tel: 0531 820900.

MACINTOSH 128k, Imagewriter printer, 4 months old, with boxes, £2,142 new, sell £1,800 inc. Macwrite/Macpaint plus other software. Phone (0734) 696620 evenings.

APPLE II/III, Systematics invoicing, stock control, general ledger, £100 each. Applewriter, Applepost, Appleplot, £30 each. E. Evans, 59 Waltham Terrace, Blackrock, Dublin.

APPLE BITS. Daisystep daisywheel printer, £175. Juki 6100 printer, £250. Parallel printer card, £15. New Format80, £40 (enhanced). New Apple Pascal, £35. Game joystick, £5. Call 0703 787916 or 769705 evenings.

APPLE III 5mbyte hard disk and colour monitor, all hardly used. Nearest offer to £2,000 including £350 software FREE. Ring 794 7709.

CARRYING CASE IIe plus drives, customised aluminium, £80. Mountain hardware clock card, £80. Tel: 0865 56321 (Oxford).

APPLE IIc monitor, £119. Koala pad w/software, £49. Compatible Mouse II w/software, £49. Speech card w/software, £49. IBM compatible joystick, £25. All few months old. Phone 041-332 7969 evenings.

APPLE II+ 64k, Z80, 80 columns, two original drives, Grappler, FastDOS, ProDOS, structured basic, Pascal, Fortran, Wordstar, games, utilities. Fast sale, £699, will split. Tel: 794 6250.

APPLE GAMES for sale, all original, complete with documentation under half price. Send stamped SAE for descriptive list to: 110 High Street, Hurstpierpoint, Hassocks, West Sussex BN6 9PX.

ROBOCOM BITSTIK 1000 plus software and manuals. Cost £500, accept £275. Tel: 0438 60926.

APPLE II+, disc drive interface and one drive (needs attention), £200. Tel: 0438 60926.

COMPLETE SET of Windfall/Apple User, bound four volumes plus, £40. Tel: 0438 60926.

APPLE IIe, twin disc drives, Kaga green screen monitor, extended 80 column card, Epson FX80 printer and Grappler interface, PCB. Z80 card with CP/M (no manual), two serial cards, two music cards (Alf and Zapple, T.G. joystick and Glanmire Timekit plus £750 worth of software including Applewriter, Visicalc, Superbase, Ramdrive, and 38 BASUG library discs nine of which are special release issues. Also Quickfile and Apple Toolkit etc, £1,550. No offers, will not split. Tel: 0438 60926.

APPLE 48k (lookalike) D/D 9ins hi-res monitor, 16k lang. card, joystick, printer card + cable, RX-80 printer and paper, Homeword W/P Wizardry etc + manuals, £650. Phone Bim (0784) 59211 (office), 59783 (home).

apple classifieds

- Classified ads can only be accepted from private readers, not companies.
- The cost is 20p per word, with a minimum of 10 words prepaid.
- Your ad will be printed in the next available issue of Apple User.
- Your accompanying cheque should be made payable to Apple User.
- Ads can only be accepted on this form (or a photocopy of the form).
- There is no maximum to the number of words you include in your ad.
- Ads too long for the form should continue on a separate sheet of paper.
- Ensure your phone number or address is included in the ad.

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	10 words £2 00
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	15 words £3 00
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	20 words £4 00
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	25 words £5 00
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	30 words £6 00

Name _____ Address _____
 POST TO: Apple Classifieds, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

I AM the owner of an Apple compatible 6502/Z80 micro with 64k memory.

I am visually handicapped with very little eyesight left. I normally read a 12in screen with the aid of a magnifying glass and I wonder if any readers can help with the following problems.

1. I have altered the catalog routine so that the output has a gap between alternate lines enabling me to read it more easily. I would like to do this with all ordinary output on the screen when writing programs, in other words changing the screen format to 40 columns x 12 lines.

2. I would like to make use of the DOS Toolkit facilities or Apple Mechanics to make bigger letters, or when necessary, to change the text to bigger letters if I am not sure what is written there.

3. I own an SCB-Apple II speech card and would like software similar to the Echo II to use with it. Or is there an alternative card and software at a low cost or a better system?

4. As I'm busy learning braille, I am looking around for information on building a 20 cell or more braille terminal for my computer.

If anyone can help I would be very grateful and I'm sure many of my blind friends would also be very interested as there are a few of them who own Apples and cannot afford to buy the bigger and more expensive equipment with the new speech card.

5. Can anyone suggest a cheap printer? - **Renaldo Guelpa, Natal, South Africa.**

● The following short program may help with the first point when you are programming under DOS 3.3 in either Basic or in the monitor.

It intercepts the normal output routine, putting lines on the screen with a gap in between until at the bottom of the text window.

When at the bottom it waits until any key is pressed then clears the screen and puts out more lines. It will affect the catalog output as well.

Line editing programs will interfere with the program as will any printer output. After printing restore it with a CALL 768. Type the program in as

Program to assist visually handicapped

follows after entering the monitor with a CALL -151:

```
300:A9 B A0 3 B5 36 84 37 4C EA 3 C9 8D D0 1B 4B
310:A5 23 38 E9 3 C5 25 B0 D 84 35 A4 24 20 1B FD
320:20 58 FC 4C FC FD 68 20 F0 FD 4C F0 FD
```

Then type 300L and check that the listing corresponds to this:

0300-	A9 0B	LDA	##0B
0302-	A0 03	LDY	##03
0304-	85 36	STA	##36
0306-	84 37	STY	##37
0308-	4C EA 03	JMP	##03EA
030B-	C9 8D	CHP	##8D
030D-	D0 1B	BNE	##032A
030F-	4B	PHA	
0310-	A5 23	LDA	##23
0312-	38	SEC	
0313-	E9 03	SBC	##03
0315-	C5 25	CHP	##25
0317-	B0 0D	BCS	##032B
0319-	84 35	STY	##35
031B-	A4 24	LDY	##24
031D-	20 1B FD	JSR	##FD1B
0320-	20 58 FC	JSR	##FC58
0323-	4C FC FD	JMP	##FDFC
0326-	68	PLA	
0327-	20 F0 FD	JSR	##FDF0
032A-	4C F0 FD	JMP	##FDF0

When all is well save the program by typing:

```
BSAVE LINESKIP,A$300,L$2D
```

Start it off by typing 300G, or CALL 768 from Basic, or BRUNNING it from disc.

In answer to point 5, I would suggest that you buy a dot matrix printer with Epson compatible control codes - or even an Epson. You also need an intelligent interface card which uses Cntrl-I as its command code for ease of use with published software.

Max Parrott

Offer they can't refuse?

HAVE you any ideas how to get firms in the USA to reply to telexes?

I have a duff and legitimate

Visiterm, and all I want is a courteous reply from Paladin or Visicorp to say they won't help with a new (or used) copy.

All the firms here that are in the chain of supply for my copy have gone bust - or is it just me? - **Alan Finn, London.**

● In our experience US companies work well on the phone, but it's an expensive way of doing things. Do any of our readers have suggestions?

Prodos tip

MY dot matrix printer requires an 'Escape E' code to turn on 12 pitch (Elite) printing.

To save me putting this code at the beginning of all my Applewriter IIe (Prodos version) documents, I wonder if there is some way of permanently altering the program to send this code prior to printing?

I think the code would have to be sent prior to every print run because resets and power downs cause the printer to return to its 10 pitch (pica) default setting. - **J. Bebbington, Guildford, Surrey.**

● Can any of our readers help? We don't have Applewriter IIe under Prodos.

Hidden commands

I AM writing this in the hope that it will enable other Apple users to avoid the shock of watching a disc full of work being erased in just a few seconds.

I had just finished assembling a program from the spring 1984 issue of Orchard Computing called Auto Number and as I had

been having problems with my assembler, I decided to exec the source file with MON C,I,O in operation to check that this had not been corrupted in any way.

I expected the string of syntax errors which occurred, but I wasn't ready for the result of line 4 of the listing which read:

```
INITIALISE PHA
```

On reading the label INITIALISE my micro decided that I wanted it to re-initialise the disc, which it did very efficiently, creating a bootfile named IALISE PHA, and wiping the disc.

In retrospect, I suppose I should have considered the possibility of there being some Applesoft commands in the text file, but unfortunately this did not occur to me at the time.

Luckily the only files on the disc (apart from TBSAVE) were back-ups which I was able to copy from other discs.

May I therefore ask that program writers consider the effect of all parts of their work on an unsuspecting Apple, and avoid any combination of letters which in some circumstances, might cause a lot of work to be irretrievably lost. - **Russell Ridout, East Sussex.**

● Good advice!

Adding functions

IS there any way in which I can add functions I have coded myself to the limited list of functions available in Applesworks Spreadsheet on an Apple IIe?

@SUM, @AVG, etc do not do all the things I want on a spreadsheet and I need some way of adding new functions such as cube roots.

Alternatively, is there another spreadsheet that does give me this facility, or a book to which I can refer for the information? - **R. Hegerty, Colchester.**

● Geoff Wood replies: I do not

know any way to add functions to Appleworks. The files are binary so unless you are an expert programmer, there is no easy way.

The range of functions in Appleworks is limited because it is an integrated program with a word processor and a database as well as a spreadsheet, so some things had to be left out in order to keep the program small enough to fit into the IIe.

More functions are available in other spreadsheets such as FlashCalc, Multiplan, SuperCalc and VisiCalc.

Software Arts who designed VisiCalc have been bought out by Lotus and I understand that VisiCalc has been withdrawn, though some dealers may still have stocks. (If you happen to come across a ProDOS version of VisiCalc, please let me know.)

In any case, FlashCalc is cheaper than VisiCalc and very similar in operation. It is ProDOS based and much faster at loading and saving files.

In addition to the functions offered in AppleWorks - ABS, AVG, CHOOSE, COUNT, ERROR, IF, INT, LOOKUP, NA, NPV, MAX, MIN, SQRT and SUM - FlashCalc offers AND, COL, COS, EXP, FALSE, FV, IRR, ISERROR, ISNA, LN, LOG10, NOT, OR, PERIODS, PI, PMT, PV, RATE, ROUND, ROW, SIN, TAN and TRUE. The LOG10 function should meet your need for cube roots.

Another advantage of FlashCalc is that you can create larger files because it offers 82k of RAM against the 55k offered by AppleWorks. You can expand your AppleWorks with the Ramworks card up to 101k or even 736k, but you can also expand FlashCalc with extra memory cards.

You can load files created by FlashCalc into AppleWorks because AppleWorks thinks that they are VisiCalc files. However if you use any of the functions of FlashCalc that AppleWorks does not have, you will get error messages in the cells.

Conversely, you can save an AppleWorks worksheet as a DIF file and load it into FlashCalc but the formulae will be converted to values.

Another good spreadsheet is Multiplan. It offers more functions than VisiCalc and Flash-

Want to write to Feedback? The address is: Feedback, Apple User, Europa House, 68 Chester Road, Hazel Grove, Stockport SK7 5NY.

Calc, including ATAN, DOLLAR, FIXED, INDEX, LEN, MID, MOD, REPT, SIGN, STDEV and VALUE.

It offers only 37k of RAM so you cannot create large files but you can "link" files together so that as a file loads in, it can automatically extract data from other files.

Multiplan can sort the rows into alphabetical/numerical order but FlashCalc and VisiCalc cannot.

The latest spreadsheet for the Apple II range is SuperCalc 3a, but it runs only on the Apple IIc or an enhanced Apple IIe with the 65C02 chip.

It has most but not all of the functions of Multiplan and a few others such as DATE, DVAL, ISDATE, ISNUM, ISTEEXT, JDATE, MONTH, TODAY, WDAY and YEAR.

The date functions are handy but you can get other spreadsheets to perform the same operations by using lookup tables.

A big advantage of SuperCalc 3a is that it can create graphs very quickly from spreadsheet data. It also has a so-called database, but it is more awkward to use than the AppleWorks program.

Apple III software

I RECENTLY bought an Apple III (hard luck I hear you say) and it seems I am the only person in the world using one.

There is no mention in magazines of such a beast and even advertisers don't seem to want my money.

Yet it seems a very nice machine with a good operating system, it is such a waste to use it only in II emulation mode.

Can anyone tell me where I can get software for the III? I need a word processor and a communications program (to access bulletin boards) or is there a CP/M card for the III to open up that source of

software? Any information would be eagerly received. - B.W. Mantell, Winchester.

● In 1983 Apple (USA) published a book with the catchy title of "Will Someone Please Tell Me What an Apple III Can Do?" Given the general lack of support for the Apple III, this is probably as good a sourcebook as you will find.

Not all that E-Z

I HAVE a very good graphics utility called E-Z Draw by Sirius Software and I am trying to use it as a CAD for doing printed circuits.

My intention was to draw the circuit on the screen and then dump it on to an Epson printer using a Cirtech parallel card.

Unfortunately every time I press the Esc key to dump the drawing I get the message "No Silentype or Trendcom AIIIG found".

I would like to know how I can modify this disc to recognise the Cirtech card. I have the inspector utility but can only read Track 0 sector 0 and sector D with it.

Also, every time I press the Ctrl-Reset the disc boots so I can't examine memory.

How do I stop it rebooting and how do I alter the disc to get around this problem? - E. Hickey, Birmingham.

● You might try substituting a normal DOS master before you press Ctrl-Reset.

The system will still reboot because E-Z Draw will have altered the Reset vector.

However a normal DOS will be booted. In all probability the hi-res screens will still be intact. You can check page 1 using CALL -3100 and then check page 2 using POKE -16299,0.

Assuming the pages are there you can then dump them using the firmware on the Cirtech card - assuming it supports a graphics dump.

It's not an elegant solution

but it might just work for now.

Given Murphy's Law, you won't be surprised to learn that Sirius Software no longer exists so we can't ask them how to drive different printers.

Sideways Appleworks

I RECENTLY got Sideways for printing Visicalc files sideways.

Unfortunately it does not do the same with Appleworks, but with the method below you can print out Appleworks spreadsheet files.

Load in your spreadsheet model via Appleworks and do all the appropriate calculations.

Press Open-Apple O for options and change the platen width to the maximum 13.2 inches. Then make the left and right margins 0 inches.

Use Open-Apple P for print and choose print to the clipboard for the word processor.

When it has, create a new file for the word processor - any name will do.

Then go to the printer options (Open-Apple O), change the printer options pw to 13.2 inches lm 0 rm 0, Open-Apple M to move the file on the clipboard to the new word processor file, use Open-Apple P to print the file to an Ascii text file on disc stating the appropriate pathname. If your Appleworks data disc is named "data", for instance, enter this /DATA/FILE NAME.

When the file has saved to disc you then have to change the disc's format from ProDOS to DOS using the System Utilities disc off the IIc or the ProDOS "Kit" disc.

It then can be accessed by Sideways. The only restrictions you have are that you can only print out a file of maximum width of about 140 characters - about two screens - because of the way the word processor handles the text and the size limitation on the clipboard of 250 lines. - David Palmer, Blackpool.

● We don't have Sideways to test this, so the tip is presented "as is".

With Appleworks selling so well, perhaps the Sideways people will issue a ProDOS/Appleworks version.

