Windfall

Volume 2. No. 7 January 1983 £1

MEMORY ON TAP:
Six pages of advice
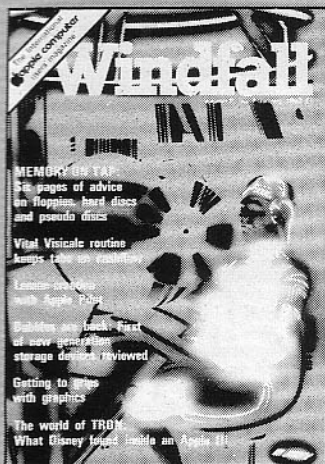on floppies, hard discs
and pseudo discs

**Vital Visicalc routine
keeps tabs on cashflow**

**Lesson-creation
with Apple Pilot**

**Bubbles are back: First
of new generation
storage devices reviewed**

**Getting to grips
with graphics**

**The world of TRON:
What Disney found inside an Apple III**

A.U.G.E. exemplar

LISTING

By David Creasey

The games kings fight for crown

COMPUTER games fever has been gripping the USA. In gladiatorial style, creators of some of the most popular disc based games have been battling against each other in a marathon knockout contest over coast-to-coast TV.

Among them are writers of Apple games that have been highly rated in Windfall. They include Dan Gorlan (Choplifter), David Sneider (David's Midnight Magic), Jay Zipnick (Pigpen), Ken Williams (Threshold) and Paul Stevenson (Swashbuckler).

The contest was launched on an unsuspecting America at a no-expense-spared party, held in the glittering, legendary Sands Hotel and Casino in Las Vegas. Windfall's Derek Meakin, who was in the gambling capital to cover the world's biggest computer trade show, reports that most of the contestants were teenagers — and rich teenagers at that. Their output is so prolific they're well on the way to becoming millionaires.

Mark Turnell, who bought his first computer at 15 and quickly designed top-selling Free Fall, Beer Run and Sneakers for the Apple, now drives a Porsche 924.

His 19-year-old friend Dan Thompson (Twerps, Lemmings) when he isn't conjuring up new games lives a life of luxury, alternating between gambling, skiing and beating up the freeway in his Trans AM.

Steve Wozniak (right) is welcomed onto ALS's board of directors by engineering vice-president Chuck Mauro.

A name to note

AS it comes naturally for Americans to do everything on the largest possible scale, the opulence of the event came as no surprise. Reports Meakin: "What was unexpected, however, was the identity of the hosts — a software distribution company which only came into existence little over a year ago.

"Not that they should have had any problem picking up the tab afterwards. Their sales are growing at the phenomenal rate of 20 per cent a month, and their last month's figures topped 27 per cent. Now they are planning to open an office in the UK and expect to repeat their success over here.

"The name of this high-flying newcomer is Software Distributors. Remember it. We should be hearing a lot more about them before long."

CP/M Plus for Apple

A BIG boost for CP/M on the Apple is expected following a deal signed between Advanced Logic Systems and Tunbridge Wells-based Vector Ltd.

They are setting up a new company in Belgium called Vector Advanced Logic, and their first product — announced in Las Vegas — is the first plug-in CP/M interface card with CP/M Plus, the newest and most powerful version of CP/M, for the Apple.

Unimaginatively called simply CP/M Card, it will retail in the USA for $399, and will cost around 10 per cent more in the UK to cover shipping costs. First supplies are expected in the middle of this month.

The card offers an additional 64k of memory, for a total of 112k fully usable by CP/M Plus in a 48k Apple. It has a 6 mHz Z80B processor, allowing users to run CP/M applications at up to three times faster than with other CP/M cards.

Advanced Logic Systems have also announced a further coup — the recruitment of Apple co-founder Steve Wozniak to their board of directors. "Steve's investment in our company is a note of confidence in our future success", said ALS president Dick Ribas.

And Steve himself said he was "delighted to be able to contribute to the research and direction of products that support Apple's product line".

Comdex records

SUCCESS stories were legion at the Las Vegas computer show, Comdex. Even the show itself broke new records. During four frenetic days it attracted more than 50,000 manufacturers, distributors and dealers.

The number of exhibitors — 1,106 — was also an all-time record for a computer industry trade show. And next November's event will be even bigger. Determined not to miss the boat, exhibitors have already booked 4,300 stand units, completely filling the massive Las Vegas Convention Center as well as substantial areas of four adjoining hotels.

New products for the Apple were well in evidence, much to the delight of UK Apple distributors Pete and Pam Fisher, who had an energetic buying spree.

Corking product

APPLE has manufacturing plants in Dallas, Los Angeles, Singapore and Cork. All the company's 240 volt machines (for the European market) are produced in Cork, where an Apple comes off the production line every 20 seconds, according to Apple's Dave King.

The machines are individually tested for four days before release. "We have a very reliable product," he said. "Less than two per cent of our annual turnover is devoted to maintenance — and when something does go wrong the average time to repair an Apple is 2.5 minutes."

Currently 550,000 Apples have been sold (51,752 in the UK) and in two months' time the company expects to become the first in history to have achieved a $1 billion turnover within five years of its inception.

Apple's current projections anticipate a $4 billion turnover by 1985. More than 750 companies now design and build hardware add ons for the Apple II and III and more than 2,500 companies produce applications software for the machines.

Software for Apple III

THERE was more good news at the Las Vegas meeting for those Apple III users who have been complaining about the lack of software for Apple's flagship machine.

Software writers and manufacturers of peripherals were offered valuable incentives and tools to help them create new products to run on the Apple III.

Apple executive Paul Dali told them: "We are out to show the many advantages of developing products for the Apple III. We realise the importance of providing developers with the best tools possible and of dealing fairly and openly with their needs."

Two hundred people sat down to lunch to hear about the enhanced capabilities of the Apple III. They were the first to hear of new products, like the high density disc drives UniFile and DuiFile, which are reported on Page 39.

And Tim Gill, head of Quark Engineering, told them about a new interpreter switcher program, Catalyst, which he has developed. This allows an Apple III user to move rapidly from application to application, utilising the power and speed of ProFile. Another of its advantages is that it maintains the copy protection features of programs stored on the hard disc.

Nestar's Harry Saal described his company's recent breakthrough that allows the Apple III to connect to Nestar networks. A special driver adds the Apple III to Nestar's Cluster/One Model A and the Plan 4000 network systems. Both systems provide local networking capabilities and shared mass storage for both the Apple II and III.

The guests left clutching lots of goodies and offers that should get them thinking hard about how they can turn the capabilities of the III to their own advantage. When the results come through, you'll be reading about them first in Windfall.

Micro matchmaker

AN enterprising Jersey company has come up with an ingenious scheme to harness programming talent and market it to businesses requiring tailormade packages.

It is a type of micro match-making scheme thought up by Terry Crouch of Number One Computers.

He is compiling a register of Apple enthusiasts and plans to bring them into contact with other users who know what they want their Apples to do, but who lack the time or in-depth knowledge to do it themselves.

Wouldbe part or full time programmers, systems people or Visicalc experts who register with The Apple Corps before the end of this month can join free of charge (subsequent registrations will be levied a £5 fee.)

Anyone wanting to meet people on the register has to pay an initial fee of £25 for the details of the six most appropriate names. Subsequent enquiries cost £18.

"We are looking for the enthusiast who wants something to do and would like to earn some money in the process," says Crouch.

"On the other side we have Apple users in business who don't want to get down to the technicalities of programming but who want their Apple to do something that they can't buy off the shelf.

"They want a specific application but they don't want to work it out themselves and they don't want it to cost a lot.

"At present if you went to a local consultant it would cost you £25 for one hour's advice. This way you get a choice of six consultants, all who have specialist knowledge of your field."

People on the register will be notified if their details are passed on — and will also be allowed to specify any organisation or individual to whom their details should not be sent.

Light fantastic

A UNIQUE long range local area network system which incorporates fibre optic cables has been developed by Symbiotic.

The company MD, Rodney Cox, says it is the fastest system available "and will remain so unless someone develops one that can best the speed of light."

Called Symbnet, the system can cover a range of up to nine kilometres between any two Apples on the network (four Apples could cover 36km). It incorporates a Winchester $5\frac{1}{4}$in Symbfile hard disc and uses a high intensity semi-conductor laser to transmit data.

The company has Symbnets at 30 test sites and will starting meeting orders this month.

Cables can be laid by the shortest route, as opposed to the more conventional use of flat ribbon or coaxial cable which is sensitive to electrical noise from fluorescent lights and photocopiers.

"Apples can be plugged into the system anywhere you can see a cable," says Cox, "and the use of fibre optics ensures that you can't lose the data signal — as it needs the force of gravity before you can bend the light.

"The government is investing £25 million in fibre optic development and Britain is a world leader in the field. We intend to see that it stays there," he added.

Big screen, small screen

YOU'VE read the book. You've seen the film. Now play the computer adventure! The transition from silver screen to computer screen is a natural one for adventure movies, according to one of the leading creators of computer adventure games, Roberta Williams.

Roberta, who devised the classic Wizard and Princess which we review on Page 30, has been working on a Sierra On-Line game based on The Dark Crystal — the new fantasy film from Muppet creator Jim Henson.

"The film making techniques and special effects used in The Dark Crystal are revolutionary," said Roberta. "None of the actors are human; they're all creations of Henson's.

"A computer adventure is a natural extension of the movie, a fantasy revolving around the age-old battle between good and evil.

"Henson's animation techniques are a first, and so is Sierra On-Line's adventure game. It's the first hi-res computer adventure strictly based on a feature film."

Through logic, luck and a series of commands, the computer adventurer guides the Gelfling hero Jen to his destiny — the return of a shard to the Dark Crystal before the planet Thra's three suns align.

If he fails, the evil reign of the Skesis will continue forever. If he succeeds, goodness will return to the planet.

"I'm thrilled to be working on this project," says Roberta.

So how long will it be before the industry turns full circle and starts making movies based on successful computer games?

Big slice of the cake

APPLE will have 60 per cent of the personal computer market within the next two to three years. That optimistic prediction comes from a man who knows more than most about the company's future plans — Mark Wozniak, brother of Apple

Roberta Williams ... "I'm thrilled"

founder Steve Wozniak.

And he says that IBM — still to launch its personal computer in the UK — will be the second biggest seller, with Tandy trailing in third place.

Mark's surprising prediction, which flies in the face of growing industry criticism of Apple's delay in bringing out any new products, shows the company is unworried by the continual spate of new computers appearing on the marketplace.

What Apple have up their corporate sleeve is the subject of growing speculation in the American computer press. One industry analyist, Michele Preston, is widely reported as confirming this January as the launch date for two new Apple micros. One she described as the company's "high end product", so different from existing computers that few dealers would be qualified to carry it. The other would take over from the Apple II as the company's volume leader.

If you want to know what's really going to happen — watch this space!

Apple link to IBM

WATCH out for a hardware/software interface that allows Apples to communicate with IBM mainframe computers — a development that will be welcomed by many large businesses.

The device, a 3270 cluster controller emulator, enables an Apple to emulate the functions of an IBM terminal. What this means is that business executives, data processing professionals and other office workers will be able to use the Apple on their desks to take full advantage of mainframe functions such as on-line data entry, inquiry and response, electronic mail, remote database access, and program development. And it can, of

course, provide normal personal computer applications when not in use as a terminal.

The move, announced at a meeting of dealers held in Las Vegas last month, will be a shot in the arm for Apple sales.

The device will be available in a SNA/SDLC (System Network Architecture/Synchronous Data Link Control) version which emulates the IBM 3276 control unit display station, and a BSC (Binary Synchronous Communication) version which emulates the IBM 3271 cluster controller.

The emulator will support up to seven directly-connected Apple computers or a larger number of Apples connected via modems. Printers can be connected either to the emulator directly or to one or more of the Apple computers.

★ ★ ★

OUT of this world is the best way to explain a hi-res arcade game currently released in the United States and ostensibly written by an alien.

The game is called

ꙅɘ꒤ꊼⱾ

and if that leaves you standing, then you're likely to have problems with the instruction manual — it is written entirely in alien text.

Southwestern Data Systems say they wanted the game to allow the Apple user to see things from the alien's point of view, with the earthmen as the aggressors. Part of the game's challenge is to decipher the rules.

The company has sponsored a nationwide competition and is offering prizes for the first person to successfully translate the advertisement for the game and for the first person to translate the complete game package and screens.
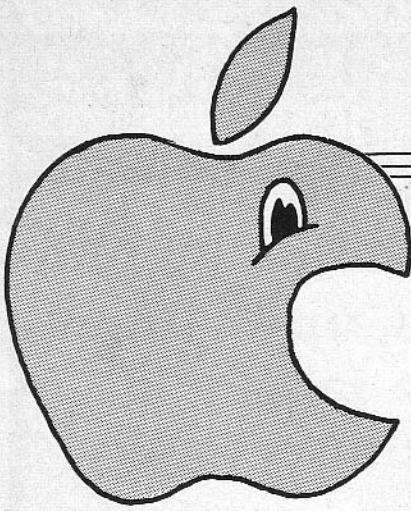
THINK TANK
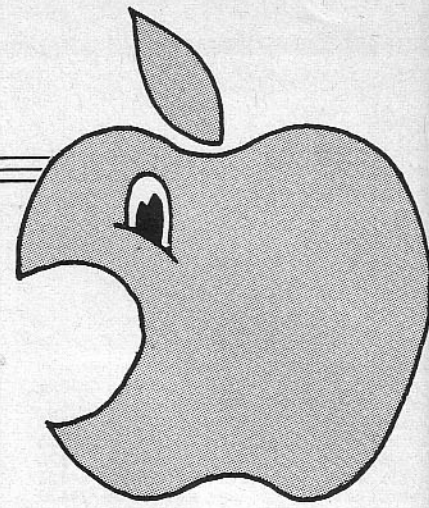
Fast routine for hi-res clear

IN last April's Windfall a routine was published which rapidly clears selected areas of the hi-res screen, writes **Max Parrott**. The routine uses a look-up table rather than calculating the addresses of each hi-res line and hence can only clear areas in discrete jumps. Here is a fast routine which calculates addresses rapidly and so allows any section of the screen to be cleared.

It is accessed from Basic by the command &CLEAR x,y TO x',y' where x,y are the values of the top left hand corner and x',y' are the values of the bottom right hand corner of the area to be cleared. The ampersand vector is set up when the program is BRUNned. Although that portion of code may now be overwritten and lost by any lengthy input, the main code is safe.

As it stands, the program only clears to black and can give a slight fringe when clearing an area coloured other than white or black. Perhaps someone can add a routine to prevent this? It is easy to access

the routine via a CALL if you don't want to use the ampersand command.

The address of any pixel in the X-direction is easy to calculate by dividing its number (0-279) by 7. The quotient gives the byte (0-39) and the remainder is the bit in that byte. This is accomplished by the XCALC routine.

The base address of any line on the screen is harder to calculate but is clear from the YCALC routine. Note that changing the value of byte $3AE to #$40 will clear page 2.

The program works by calculating the base address of the bottom line requested for clearing and the two bytes of the X-values requested. The appropriate bits of the left hand byte are reset and bit 7 reset (for black). Whole bytes are then cleared to zero until the right hand byte is found when its appropriate bits are reset. The line count is decreased and the next line up cleared until the top line is found.

The YCALC and XCALC routines should be useful in any drawing program to plot points on either screen.

```
!A
**END OF PASS 1

**END OF PASS 2

0B00          1    ***********************
0B00          2    * HIRES SCREEN CLEAR  *
0B00          3    *                     *
0B00          4    * USES &CLEAR...TO...: *
0B00          5    ***********************
0000          6    Y1       EPZ $0
0001          7    XQ1      EPZ $1
0002          8    XR1      EPZ $2
0003          9    Y2       FPZ $3
0004         10    XQ2      EPZ $4
0005         11    XR2      EPZ $5
0006         12    B        EPZ $6
0007         13    Q        EPZ $7
0008         14    TEMP     EPZ $8
00BB         15    TXTPTR   EPZ $BB
00B1         16    CHRGET   EPZ $B1
00BD         17    CLEAR    EPZ $BD
00C1         18    TO       EPZ $C1
03F5         19    AMPSND   EQU $3F5
DEC0         20    SYNCHR   EQU $DEC0
F6B9         21    HFNS     EQU $F6B9


LISA 2.5

0B00         22    ***********************
02F0         23             ORG $2F0
02F0         24             OBJ $B00
02F0 A9 4C   25             LDA #$4C
02F2 8D F5 03 26            STA AMPSND
02F5 A9 00   27             LDA #START
02F7 8D F6 03 28            STA AMPSND+1
02FA A9 03   29             LDA /START
02FC 8D F7 03 30            STA AMPSND+2
02FF 60      31             RTS
0300         32    ***********************
0300         33    START:
0300 A9 BD   34             LDA #CLEAR        ;IS NEXT CHAR
0302 20 C0 DE 35            JSR SYNCHR        ;CLEAR?

0305 20 B9 F6 36            JSR HFNS          ;GET COORDS
0308 85 00   37             STA Y1            ;& STORE THEM ETC
030A 20 80 03 38            JSR XCALC         ;CONVERT
030D 85 02   39             STA XR1           ;&STORE
030F A5 07   40             LDA Q
0311 85 01   41             STA XQ1
0313 A9 C1   42             LDA #TO           ;IS NEXT CHAR
0315 20 C0 DE 43            JSR SYNCHR        ;TO?
0318 20 B9 F6 44            JSR HFNS          ;GET COORDS
031B 85 03   45             STA Y2            ;&STORE THEM
031D 20 80 03 46            JSR XCALC         ;CONVERT
0320 85 05   47             STA XR2           ;& STORE
0322 A5 07   48             LDA Q
0324 85 04   49             STA XQ2
0326 C5 01   50             CMP XQ1           ;IS IT TO THE RIGHT?
0328 F0 02   51             BEQ CHECK
032A B0 0B   52             BCS CHECK1
032C         53    CHECK:
032C A5 05   54             LDA XR2
032E C5 02   55             CMP XR1
0330 F0 4D   56             BEQ END
0332 90 4B   57             BCC END
0334         58    CHECK1:
0334 A5 03   59             LDA Y2            ;IS Y2 BELOW Y1?
0336 C5 00   60             CMP Y1
0338 90 45   61             BCC END
033A         62    ***********************
033A         63    * NOW CLEAR SCREEN    *
033A         64    ***********************
033A         65    LOOP1:
033A 20 A5 03 66            JSR YCALC         ;GET BASE ADDRESS
033D A9 FF   67             LDA #$FF          ;CHECK IF AT TOP OF SCREEN
033F C5 03   68             CMP Y2
0341 F0 3C   69             BEQ END
0343 A4 01   70             LDY XQ1           ;Y INDEX
0345 A6 02   71             LDX XR1           ;FIRST FILL PART BYTE
0347 A9 FF   72             LDA #$FF          ;HAVE TO CALC BITS
0349 4A      73    LOOP2:   LSR
034A E8      74             INX
034B E0 08   75             CFX #8
034D 90 FA   76             BCC LOOP2
034F 31 06   77             AND (B),Y         ;NOW CLEAR BITS
0351 91 06   78             STA (B),Y
0353 A9 80   79             LDA #$80
```

```
LISA 2.5

0355 11 06    80          ORA (B),Y      ;SET COLOUR BACK          03B1 A5 03    140          LDA Y2
0357 91 06    81          STA (B),Y                                 03B3 29 07    141          AND #$7
0359 A9 00    82          LDA #0         ;NOW CLEAR WHOLE BYTES     03B5 0A       142          ASL
035B F0 02    83          BEQ D0                                    03B6 0A       143          ASL
035D          84  LOOP3:                                            03B7 65 07    144          ADC B+1
035D 91 06    85          STA (B),Y                                 03B9 85 07    145          STA B+1
035F C8       86  D0      INY                                       03BB A5 03    146          LDA Y2
0360 C4 04    87          CPY XQ2        ;REACHED END OF LINE       03BD 29 C0    147          AND #$C0
0362 90 F9    88          BCC LOOP3                                 03BF 4A       148          LSR
0364 A2 FF    89          LDX #$FF       ;NOW NEXT PART BYTE        03C0 85 06    149          STA B
0366 BA       90          TXA                                       03C2 4A       150          LSR
0367          91  LOOP4:                                            03C3 4A       151          LSR
0367 0A       92          ASL            ;FORCE 0 INTO RHB          03C4 05 06    152          ORA B
0368 EB       93          INX                                       03C6 85 06    153          STA B
0369 E4 05    94          CPX XR2                                   03C8 A5 03    154          LDA Y2
036B 90 FA    95          BCC LOOP4                                 03CA 29 08    155          AND #8
036D 31 06    96          AND (B),Y                                 03CC F0 06    156          BEQ DONE
036F 91 06    97          STA (B),Y      ;SET BYTE                  03CE A9 80    157          LDA #$80
0371 A9 80    98          LDA #$80       ;& SET COLOUR BACK         03D0 65 06    158          ADC B
0373 11 06    99          ORA (B),Y                                 03D2 85 06    159          STA B
0375 91 06    100         STA (B),Y                                 03D4 60       160  DONE    RTS
0377 C6 03    101         DEC Y2         ;USE IT AS COUNTER         03D5          161          END
0379 A5 03    102         LDA Y2
037B C5 00    103         CMP Y1                                    ***** END OF ASSEMBLY
037D B0 BB    104         BCS LOOP1
037F 60       105  END    RTS            ;BACK TO CALLER            HIS IS A HEXADECIMAL DUMP.
0380          106  ***********************
0380          107  XCALC:
0380 B6 06    108         STX B          ;LO BYTE                   ]CALL-155
0382 98       109         TYA            ;HI BYTE
0383 A0 07    110         LDY #7         ;DIVIDE BY 7 TO            *2F0.3D4
0385 84 05    111         STY XR2        ;CALC #BYTES
0387 38       112         SEC                                       02F0- A9 4C 8D F5 03 A9 00 8D
0388 E5 05    113         SBC XR2                                   02F8- F6 03 A9 03 8D F7 03 60
038A 08       114  LOOP   PHP                                       0300- A9 8D 20 C0 DE 20 B9 F6
038B 26 07    115         ROL Q                                     0308- 85 00 20 80 03 85 02 A5
038D 06 06    116         ASL B                                     0310- 07 85 01 A9 C1 20 C0 DE
038F 2A       117         ROL                                       0318- 20 B9 F6 85 03 20 80 03
0390 28       118         PLP                                       0320- 85 05 A5 07 85 04 C5 01
0391 90 05    119         BCC ADD                                   0328- F0 02 B0 0B A5 05 C5 02
0393 E5 05    120         SBC XR2~                                  0330- F0 4D 90 4B A5 03 C5 00
0395 4C 9A 03 121         JMP NEXT                                  0338- 90 45 20 A5 03 A9 FF C5
0398 65 05    122  ADD    ADC XR2                                   0340- 03 F0 3C A4 01 A6 02 A9
039A 88       123  NEXT   DEY                                       0348- FF 4A E8 E0 0B 90 FA 31
039B 10 ED    124         BPL LOOP                                  0350- 06 91 06 A9 80 11 06 91
039D B0 03    125         BCS LAST                                  0358- 06 A9 00 F0 02 91 06 C8
039F 65 05    126         ADC XR2                                   0360- C4 04 90 F9 A2 FF 8A 0A
03A1 18       127         CLC                                       0368- E8 E4 05 90 FA 31 06 91
03A2 26 07    128  LAST   ROL Q                                     0370- 06 A9 80 11 06 91 06 C6
03A4 60       129         RTS                                       0378- 03 A5 03 C5 00 B0 BB 60
03A5          130  ***********************                          0380- 86 06 98 A0 07 84 05 38
03A5          131  YCALC:                                           0388- E5 05 08 26 07 06 06 2A
03A5 A5 03    132         LDA Y2         ;CALC BASE ADDRESS OF LINE 0390- 28 90 05 E5 05 4C 9A 03
03A7 29 30    133         AND #$30                                  0398- 65 05 88 10 ED B0 03 65
03A9 4A       134         LSR                                       03A0- 05 18 26 07 60 A5 03 29
03AA 4A       135         LSR                                       03A8- 30 4A 4A 4A 4A A9 20 85
03AB 4A       136         LSR                                       03B0- 07 A5 03 29 07 0A 0A 65
03AC 4A       137         LSR                                       03B8- 07 85 07 A5 03 29 C0 4A
                                                                    03C0- 85 06 4A 4A 05 06 85 06
LISA 2.5                                                            03C8- A5 03 29 08 F0 06 A9 80
                                                                    03D0- 65 06 85 06 60
03AD 09 20    138         ORA #$20   ; CHANGE TO #$40 FOR PAGE 2.   *
03AF 85 07    139         STA B+1
```

Standard 6502 opcodes?

I WAS disturbed, and (very) slightly annoyed by Max Parrott's reply to A.M. Oldacre's letter in November's issue of *Windfall*, writes **D.M. Miller**. I have had similar experiences with some assemblers.

Surely no matter how much reading is done, the user cannot be expected to become accustomed to the unusual quirks of some assemblers labelled "Standard 6502 Assembler Opcodes" which either have been changed to ease the programmer's task, or, in the case of Synertek's assembler, have been replaced by different mnemonics borrowed from the instruction set of another microprocessor.

The 'BLT' and 'BGE' instructions are not standard to the 6502 (Synertek have used them as alternatives for the 'BCC' and 'BCS' instructions respectively), and if you look in the 6502 instruction and tutorial books (well, the ones that I have

seen) you will not see any reference to 'BLT' or to 'BGE'. These two instructions belong to the 6502's distant father, the M6800 and its family.

Surely in such a field as ours standardisation is most important and such non standard examples should be discouraged. I, with my limited experiences of using assemblers and emulators, have not seen such great variation in the mnemonics used in other microprocessors for example:

● In M6800 assembly code there is only one form of: LDA $30

● In Z80 assembly code there is only one form of: LD A,30H

● But in 6502 assembly there are at least five: LDA &#30 LDA #$30 LDA #30H LDAIM $30 IMLDA $30

All these opcodes will load the accumulator (accumulator A in the M6800) in the immediate addressing mode, with the hex value 30.

Here are some more unstandard monstrosities:

Standard 6502		Very non standard 6502	
LDA	$07	LDAZ	$07
ADC	$190F	ADCA	$190F
STA	$0129,X	STAX	$190F
SBC	$0DDF,Y	SBCY	$0DDF
JMP	($0036)	JMPIN	$0036
LDA	($01,X)	LDAIY	$01
STA	($03),Y	LDAIX	$03

Thankfully the last set of examples of nonstandard 6502 opcodes are very uncommon indeed!

It can be quite confusing for the relative beginner to assembly language, and laborious for the programmer, to have to learn a new set of nonstandard opcodes for each different 6502 assembler.

Surely in this day and age (Information Technology, Prestel, and the ZX81) we can expect just a little 'bit' of standardisation between assemblers!

BOOK REVIEW

Revealed – all the secrets of Apple graphics

I RARELY feel envious of another person's expertise, but Jeffrey Stanton has written a book – Apple Graphics and Arcade Game Design – I would like to have written myself. What is worse, he has given away all those secrets of Apple graphics which have taken me years to uncover, and has done it in a clear and lucid style. I might just have managed to have swallowed my pride and accept this, but Jeffrey has rubbed salt in the wound, in that I learnt a lot from his book as well!

It has something for everyone – whether you are programming in Applesoft, a machine code beginner or even someone who thought they knew a reasonable amount about the Apple's internals and graphics routines.

It starts with a review of the graphical commands available from Applesoft and deals with routines for page flipping, and production of shape tables and their use in animation. Typical of the book's style is the fact that a useful graphics EXEC routine is given within the first five pages.

The coverage of the character generator and ANIMATRIX routines in the DOS 3.3 Toolkit shows the broad sweep this book covers (and all by page 33!)

Although Applesoft graphics are handled in a more than adequate manner this is not a book about Applesoft and the remainder is devoted to the use of machine code (or assembler) routines, which are necessary "for the speed, flexibility, colour and smoothness of quality arcade games."

The section starts with an introduction to assembler fundamentals, comparing them to similar Basic routines. This is consolidated with a lo-res "Breakout" type game written in assembler. This section is explained in an easy style, but I would recommend a beginner to refer to a 6502 text book as well.

The book really moves into its own when it starts to show the machine code routines that drive the Apple hi-res system. Jeffrey Stanton gives you all the pointers and entry points for using all these routines, and shows how they can be used to animate a simple bird-like shape. He also lists the benefits over Basic.

I liked his explanation of the way

By Ed PEACH

colours are displayed on the hi-res screen, not only for its clarity but also because it was the only area in which I found a small omission. He doesn't mention "dithering" – the apparent colour effects which can be obtained by stacking different colour layers above one another.

The pace then hots up, and the reader is introduced to raster (or block-shape) graphics, a method of plotting shapes even faster than using a shape table and machine code.

Once these basic methods are grasped the reader is encouraged to use them in routines for moving guns, dropping bombs, shooting bullets and interfacing with the paddles. The routines are then incorporated in a simple invaders type game.

Having digested all this the reader still has one third of the book to cover. Sections still remaining cover such things as steering spaceships, collisions, explosions, score-keeping and moving backgrounds. All these items are linked together in a Pegasus II type game.

If there are any errors in this book I haven't found them yet. On the other hand I haven't finished typing in all the 90-plus pages of assembler code routines and games. The publisher says that copies of these routines can be bought from them by all purchasers of the book. This is the best book on Apple graphics I have seen and I thoroughly recommend it.

Apple Graphics and Arcade Game Design is published by The Book Co, Los Angeles, price $19.95.

Anyone who has implemented my lowercase monitor (see October Windfall) may like to know of a little patch that makes the shift key available to Pascal, writes **J.P. Lewis.**

Hidden inside System.Apple, there is a routine that checks the paddle button 3 location ($C063), and uses it as a shift key while the system is in the **CTRL-E** mode for lowercase.

By running the little program on the right, you can alter this so that it checks the shift key at my location of $C066. If you have a lowercase character generator, the other changes to System.Apple will make the screen show upper and lower case rather than the standard inverse/normal characters. (This will work even without the shift key mod.)

With this change the special characters above M,N,P are obtained by holding both **SHIFT** and **CTRL**; furthermore, the 'soft crash' **CTRL-SHIFT-P** is disabled.

I haven't yet perfected the idea, but nevertheless what it does could be very handy.

Shifting in Pascal

```
program SHIFT;
var
    BUF:packed array [0..31,0..511] of 0..255;
    F:file;
    I:integer;

begin
    reset(F,'SYSTEM.APPLE');
    I:=blockread(F,BUF,32);
    close(F);
    BUF[4,61]:=102;    (* To get BIT $C066. *)
    BUF[5,401]:=97;    (* Comparisons to identify *)
    BUF[5,405]:=123;   (* lowercase input. *)
    BUF[5,409]:=9;     (* This pair changes *)
    BUF[5,410]:=224;   (* EOR £$40 to ORA £$E0. *)
    reset(F,'SYSTEM.APPLE');
    I:=blockwrite(F,BUF,32);
    close(F);
end.
```

Paddles prompter

THIS program can be incorporated into a game to see if paddles are present.

Make sure you switch off the Apple before plugging in the paddles.
Michael Osborne

```
10  REM   MICHAEL OSBORNE 1982
20  REM   PROGRAM TO DETECT PADDLES
30  P =  PEEK ( - 16287) > 127
40  Q =  PEEK ( - 16286) > 127
50  IF P AND Q THEN  HOME : INVERSE : PRINT "YOU HAVE NO PADDLES "
    : NORMAL : END
60  REM  REST OF GAME
70  HOME : PRINT "YOU HAVE GOT PADDLES PLUGGED IN NOW "
```

If a long program gets corrupted when you use the hi-res graphics page try this routine to stop your program being overwritten by the graphics page(s):

```
HGR PAGE 1
10 POKE 103,1
20 POKE 104,64
30 POKE 16384,0
40 PRINT CHR$(4);"RUN PROGRAM"

HGR PAGE 2
10 POKE 103,1
20 POKE 104,96
30 POKE 24576,0
40 PRINT CHR$(4);"RUN PROGRAM"
```

More free disc space on tap

When a DOS 3.3 disc is nearly full it is often useful to free the unused sectors 5-15 on track 2 to gain extra space. The following simple procedure enables you to do this very quickly from the monitor, without having to run a ZAP program or otherwise interfere with normal user RAM:

1. **CATALOG** the disc to be modified. (Sets correct slot and drive and reads the VTOC into its DOS buffer.)

2. ***B3FB: FF E0** (Modifies the VTOC buffer.)

3. ***AFFBG** (Calls the DOS routine which writes the VTOC buffer back to disc.)

Alternatively, equivalent instructions can be issued in immediate mode from either Basic: (1) **CATALOG** (2) **POKE-19461,255** (3) **POKE-19460,224** (4) **CALL-20485.**

Michael Clark

Recursive Applesoft for speedy sorting

One of the advantages that Apple Pascal has over Applesoft is the way it allows you to set up subroutines that call themselves. If you have read my article on the Quicksort (Windfall, September '82) you will appreciate how convenient this can be.

Here, to show that recursion can be done in Applesoft if you are prepared to look after your own stack, is the Applesoft version of that same routine.

The key to handling self-calling routines is to set up an array (or set of arrays) indexed in some way by the depth of recursion (the variable N in this program.) This array is then used to store the current values of all the important variables just before the routine calls itself. For ease of handling, the following four steps should occur:
(i) Stack the current values
(ii) Set the "ingoing" parameters
(iii) **GOSUB** to the routine
(iv) Unstack the saved values immediately after coming back.

Although the Quicksort in Applesoft is far less efficient than it is in Pascal, it is still five times as fast as a bubblesort, and only 10 per cent slower than an optimised shellsort over 100 numbers. (At the 500 mark, the 10 per cent swings the other way.) The real value of the Quicksort however lies in sorting large lists of strings, as it does far fewer swaps than the shellsort, thus producing far less garbage.

If you intend to use this program for large lists you will need to change the size of the stack array. As a rough guide, the stack length should be slightly greater than the log to base 2 of the number of items, e.g. stacksize 12 for 500 items, 15 for 2,000 items.

J.P. Lewis

```
10   GOTO 1000
20   REM              J.P.Lewis
30   REM          Quicksort 2/10/82
35   REM  .
40   REM      Set the low/high point
     ·ers for a single pass.
50   JLOW = LOW:KHIGH = HIGH
55   REM      Pick a pivot item (the
     middle)
60   PIVOT = A((JLOW + KHIGH) / 2)
65   REM      Reentry point for one
     pass;  if the pointers have
     crossed, the pass is over
70   IF JLOW > KHIGH THEN 200
75   REM      Sweep down from top un
     til you hit something less
     than or equal to the pivot.
80   IF A(KHIGH) > PIVOT THEN KHIG
     H = KHIGH - 1: GOTO 80
85   REM      Sweep up from bottom
     until you hit something grea
     ter than or equal to the piv
     ot.
90   IF A(JLOW) < PIVOT THEN JLOW =
     JLOW + 1: GOTO 90
95   REM      If your pointers have
     not crossed, they are point
     ing at a pair that needs to
     be swapped.
100  IF JLOW < = KHIGH THEN TEMP
     = A(JLOW):A(JLOW) = A(KHIGH
     ):A(KHIGH) = TEMP:JLOW = JLO
     W + 1:KHIGH = KHIGH - 1
105  REM      Go back to continue
     the pass.
110  GOTO 70
195  REM      Stack the pointers to
     the ends of the top half of
     the list.
200  N = N + 1:STACK(1,N) = HIGH:S
     TACK(2,N) = JLOW
205  REM      If necessary, quick-
     sort the bottom half.
220  HIGH = KHIGH: IF LOW < HIGH THEN
     GOSUB 50
225  REM      Unstack the pointers
     to the top half, and quick-
     sort it if necessary.
230  LOW = STACK(2,N):HIGH = STACK
     (1,N):N = N - 1: IF LOW < HI
     GH THEN  GOSUB 50
240  RETURN
1000 BIG = 100: DIM A(BIG),STACK(
     2,10)
1010  FOR K = 1 TO BIG:A(K) =  INT
     (10 + 90 *  RND (0.3)): PRINT
     A(K);" ";: NEXT
1020 N = 1:LOW = 1:HIGH = BIG
1030  GOSUB 50
1040  FOR K = 1 TO BIG: PRINT A(K
     );" ";: NEXT
```

GAMESMANSHIP

Fairy tale for adults, too

NO WAY.
THERE IS A PRINCESS FOLLOWING YOU.
YOU ARE IN A TINY ROOM.
------------------- ENTER COMMAND?

By CLIFF and DENISE McKNIGHT

THE wicked wizard Harlin has abducted King George's fair daughter, the Princess Priscilla, and the king has offered half his kingdom to anyone who can bring her back.

You are a "happy wanderer" passing through a village in the kingdom of Serenia when you hear a proclamation to this effect, and since you love an adventure you decide to find her.

All you know is that the wizard's castle is in the mountains to the north and all you carry are a flask of water, a knife, a loaf of bread and a blanket. Now read on . . .

This is the setting for Wizard and Princess from On-Line Systems. You control the direction of travel (N, S, E, W, up, down, etc) and can inspect objects in the

frame in more detail. If you don't do this you'll miss vital clues, but a closer look at a snake can be fatal!

On your travels you meet dangers, find various objects and even learn a few magic tricks besides having to use your common sense and inventiveness.

The input, apart from directions, is mainly in two-word "sentences" such as

LOOK ROCK or GO DOOR. Sometimes you have to be quite persistent to get your instructions across, yet at other times the program almost anticipates you and fills in the missing actions.

The program allows you to initialise a "save game" disc and you can save the game at any point under letters A to O. Hence there is the possibility of saving up to 15 games. This not only enables you (and 14 friends?) to play the game over several days or weeks but also enables you to try different combinations of possessions and tactics in meeting the various challenges.

When you are killed off or hopelessly stranded, you can go back to where you last saved the game instead of starting from scratch. This can be vital because sometimes you don't get a second chance, and things can have disappeared by the time you return for a second look.

There are six main scenarios to the adventure. The first starts when you boot the disc and find yourself in the village in Serenia. Following your only hint that Harlin's castle lies north, you step north and meet a snake which you cannot pass or go around.

Fortunately, enclosed with the disc was a folded slip of paper marked "Hint to get past the rattlesnake" and subscripted "Read only if absolutely necessary". Well, we tried, and finally we read that we needed to go south into the desert and find a rock.

We're not really giving anything away by saying this because the hint could well have appeared in the instructions without spoiling the game; the slip of paper had the air of an erratum slip. There are

YOU ARE IN FRONT OF THE CASTLE. THERE IS A MOAT AROUND THE CASTLE FULL OF CROCODILES.
------------------- ENTER COMMAND?
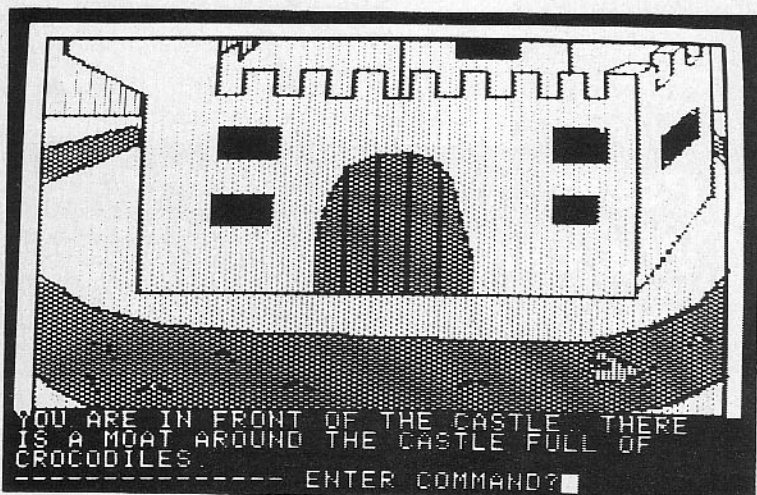
several rocks in the desert and your first task is to find one which isn't guarded by a scorpion.

The desert is an identical-frame maze (On-Line seem fond of these) and is one of the hardest parts of the adventure, although don't expect plain sailing if you get through it.

The worst part of the maze is that normal concepts of map-making go awry. You can go north one frame and then south and *not* end up where you started; or you might go south and west for several frames and then a single step east can get you back where you started. This can be very frustrating, but is not impossible to map.

Each of the six scenarios has a "gate" which gets you from one to the other and which is the only way past — no cheating and going round the back way.

The rattlesnake is the first gate into a second desert, then there is a ravine to cross into a wood. The fourth scenario is an island (the gate to which is a leaky boat) and the fifth is the mountains.

The last scenario is Harlin's castle, which was suitably complicated for the culmination of a couple of weeks' playing.

In order to succeed in rescuing the princess, you have to have solved all the preceding puzzles and collected everything you could.

The only way to check that you have been everywhere is to map constantly and try every direction from every frame. We drowned twice doing this, but fortunately we had saved the game.

On-Line's hi-res graphics are very good, and objects which you drop appear as part of the picture as they would if you dropped them while walking. One consequence of this is that if you drop several items, some appear mysteriously suspended in mid-air.

Where text needs more than the four lines available at the bottom of the picture, a beep (which we soon found irritating) prompts you to press "return" to get the rest of the message and the "enter command" prompt.

Wizard and Princess is a nice adventure game and can be recommended if you like fairy tales. We're still waiting for our half-kingdom reward to arrive — they must be having trouble wrapping it.

QUICK SPINS

Kabul Spy. *You are an agent searching Afghanistan to rescue Professor Paul Eisenstadt before he is interrogated. Once you cross the border you are on your own. Let's hope you are as clever as you are ruthless.* (Sirius Software)

Crisis Mountain. *A high-res arcade game requiring quick reflexes and strategic thinking. Your mission is to defuse bombs planted by terrorists in the treacherous caverns of an active volcano. To reach them you must leap tumbling boulders, crawl through claustrophobic tunnels and cope with other hazards such as bubbling lava streams and radioactive bats. If you don't reach the bombs in time they and the volcano will blow up.* (Synergistic Software)

Mastertype. *A typing instruction system in a hi-res game format. Players learn to type (there are 17 progressive lessons) as they battle wave after wave of attacking enemy words.* (Lightning Software)

Bolo. *A strategic, arcade-action tank warfare game in which you have to manipulate your tank through a maze that is 132 times larger than the monitor screen. As the maze scrolls left and right you have to search out and destroy the enemy tank bases while fighting off their defending tanks. There are five types of enemy machines, each with a different speed and firepower. Players can choose from nine levels of play and five levels of maze density.* (Synergistic Software)

● **Fuller reviews of some of these games will appear in future issues.**

Escape from Rungistan. *You wake up in jail in Rungistan to learn that you are to be shot at sunrise. Can you break out, cross Rungistan with its severe climate and topography and make it to the border? Only the very clever will live long enough to demand their money back from Honest Bob's Discount World Tours.* (Sirius Software)

Pill Box. *You are the last hope of the Allied forces. You must face and destroy the endless stream of enemy tanks, but you have only a limited number of shells. From your pill box hidden in the hills you must track them on your radar screen, take aim, and wipe them out!* (Software by H)

County Fair. *Enjoy the excitement of a county fair in your own home. Reproduced for you is the county fair's most popular attraction, the shooting gallery. It's simple to play, simple to score, but not so simple to win.* (Datamost)

Tubeway. *You've been out on a quiet mission patrolling the frontiers of space when suddenly the universe itself seems to crack and tremble. Before you can react you find yourself trapped on the tubeway . . . a strange creation of beings from a geometric universe.* (Datamost)

Aztec. *The legends speak of the fabulous golden idol hidden within the mysterious lost pyramid. You have discovered the pyramid and your job is to descend into the temple, search the secret rooms, avoid the deadly traps, find the priceless idol and escape alive.* (Datamost)

Type Attack. *One day on the planet of Lexicon the Alphabet began to attack from the sky. The citizens of Lexicon found that the attacking letters and words could be repelled by typing the same letters and words on a keyboard aimed at the sky. Now Earthlings can learn to type quickly and accurately and have fun too.* (Sirius Software)

Wayout. *Step right up! Who'll be brave enough to take this compass and mapmaker and walk into this maze? You will? Good! Now, I'll just lock this door here and you can try to find your way out. If you do, I've got 25 other mazes for you to try — all in 3-D.* (Sirius Software)

Repton. *The Quarriors have attacked Repton! They're building a base station and are draining our power supply. They're guarding Repton with everything they've got. You and your squadron, armed with a laser gun, nuke bombs, a radar screen and an energy shield, are our last hope.* (Sirius Software)

Snack Attack. *You've got it again! First, that empty feeling, then the pangs, finally you feel it . . . you are having a snack attack. Your only hope for relief is to gobble up as many gumdrops as you can grab away from the greedy Gumdrop Guards.* (Datamost)

Princesses really shouldn't be let out on their own..

SOMEWHERE in the distance a clock strikes twelve and once again your intrepid reporter sets out to rescue another princess. Honestly, princesses these days just don't know how to look after themselves, getting captured by any passing wizard. It's not like it used to be when I was a lad.

As you might guess from the title, the setting for Transylvania from Penguin Software, is straight out of a Hammer horror movie, complete with vampire and werewolf. There are also a few non-Hammer inhabitants like goblins and frogs, and one you might not expect. Until you work out what to do with them, the vampire and werewolf stand a fair chance of driving you mad.

The vampire inhabits the castle and the werewolf roams free everywhere else. Their trick is to appear at random, and once they appear you can't do much more than move to another frame. Just when you think you've worked out what to do about the cat, for example, and race there to try out your idea, there's that damn werewolf again!

The clock strikes twelve at the beginning of the game and you have until dawn (5 o'clock in Transylvania) to rescue Princess Sabrina. If you fail by that time, you find her lying in a pool of blood. I would guess that the "time" is based on the number of moves, so it becomes important to work out what to do with the werewolf because dodging him can be "time-consuming".

I've mentioned the clock striking, but in fact Transylvania makes no use of sounds whatsoever. Clocks (which chime on the hour), screams and general things that go bump in the night are all represented in the text at the bottom of the graphics screen. The squeamish among you will be glad to learn that there is no X-certificate stuff in the graphics. Pools of blood and your hair being set on fire are represented in the text only, like the sounds.

There is a liberal sprinkling of random spooky sounds and events which means that the text on any particular frame is not totally predictable. Even if you think you know where you are and where you are going, you may suddenly find yourself whisked off elsewhere.

The graphics are very good with some interesting attention to detail, although no matter how much I tweaked my TV I couldn't reproduce the colours in the illustrations on the back of the manual. This could be a UK/USA colour system difference, and the colours were quite nice anyway. The colour-fill system seemed faster than some others that I've seen recently.

The trouble with choosing a setting like Transylvania is that there are so many clichés around. To this extent, some parts of the game might seem easy. However, there are a few twists and a few surprises

STAIRS LEAD UP
THERE IS A TREASURE COFFER.
THERE IS A VAMPIRE.
COMMAND :

to keep it interesting. Overall, I'd say it was of average difficulty.

There is a save game facility which allows an infinite number of games to be saved. The game is saved to any initialised DOS 3.3 disc and the file is named in the normal way. Hence the limit on number of saved games is the amount of free disc space you have. Unlike many games, you can't restore a saved game from within the current game, you have to reboot (or be killed off) in order to be given the option to restore. This isn't a problem really, more a point of convenience.

Transylvania was fun to play, and I found the setting a refreshing change from the run-of-the-mill adventure games. Subtly embedded within it is an advert for another game from Penguin, which opens up a whole new range of possibilities. For example, suppose advertisers paid to have their product mentioned; assuming a constant profit margin (I *know* I'm naive), the cost of the game to the buyer could be reduced. Perhaps princesses in the future will require feeding with a plate of baked beans in order to awaken them.

Cliff McKnight

...and especially not in horror movie territory

YOU ARE IN A CEMETERY. PATHS GO N W S.
THERE IS A GRAVESTONE.
THERE IS A MENACING WEREWOLF.
COMMAND :

Background to the big back – up business

COMPUTERS are all about memory. All you get when your cheque passes over the counter for an Apple is a 6502 processor and a variety of memory devices. These are an electronic/magnetic version of the filing cabinet information storage and retrieval systems of the traditional office.

The hobbyist might use a cassette tape for storage, but for the rest the choice lies between removable floppy discs, hard discs or pseudo discs. The floppy has two standard sizes – $5\frac{1}{4}$ and 8in – and while a smaller floppy is currently being marketed no decision has yet been made as to an industry standard. But it seems likely that the major manufacturers will settle on $3\frac{1}{2}$in.

Hard discs for the Apple have traditionally been "fixed". That is, they have a much greater storage capacity than a floppy, but they are permanently encased and once "full" a decision must be made as to whether to delete little-used files to provide more room for storage, or to buy a further hard disc unit to expand the system.

Removable hard discs offer large capacity with the floppy system's flexibility, and the latest trend is the development of removable mini discs.

The most realistic way to assess the cost effectiveness of an initial investment is to relate the pounds spent to the amount of memory purchased.

The Apple itself uses random access memory (RAM) to store programs as data. This shouldn't be confused with ROM, or read only memory, which can be compared to a text book in that it can be called up for use at any time, but cannot easily be changed by the user.

RAM may be envisaged as a large variety of pigeon holes (about 48,000) in which the data is stored.

The contents of any individual pigeon hole can be speedily accessed and changed during the execution of a program considerably faster than similar changes can be effected on external storage systems. The contents of this memory however are not permanent – all the pigeon holes are cleared on switching off power.

Floppy discs operate at a much slower speed but when power is switched off, the data remains stored for future use. Floppy disc storage is up to 1,000 times slower to access than the Apple's core memory. The busy Apple user cannot afford to spend the time waiting for floppy discs to LOAD or SAVE files however and so the search is always on to speed access.

Many approaches have been made to the problem. The first answer, the hard disc, is probably the best in terms of speed of access and size of memory made available but it involves the greatest cash outlay — although that does not necessarily mean it is the most expensive when costs per unit of memory are calculated.

Eight inch floppy discs offer more memory than the conventional $5\frac{1}{2}$in floppies but not a great deal more. And as they cost more and offer a minimal speed increase they are a rather unattractive option.

Commercially available software changes have been made to the standard DOS 3.3 operating systems so that the "skewing" of the sectors is physically changed on the disc surface. This can offer quite significant speed improvements, perhaps factors of two or three dependent on the operation, but reliability may suffer.

Another approach is to set aside an area of RAM to be used as a pseudo disc; this is offered commercially as a plug-in card bearing 128k of RAM for the Apple II.

The resulting speed increase is fantastic but there is one serious objection. The memory is volatile – files are lost when power goes off – so one has to transfer the contents of the pseudo disc to a real one to give permanent memory.

We investigate in this section two 128k RAM cards, as well as another development in disc emulation – a bubble memory device.

More than 750 companies worldwide make hardware add-ons for the Apple II and III. The table opposite gives details of some of the disc drives and disc drive subsystems currently available in the UK.

In looking for a system a user should establish that it is reliable, easily-maintained and offers good back-up. Other factors are storage capacity, quietness in use, and compatibility with existing operating systems.

Make and model	Supplier	Price	Hard or floppy	Formatted mbyte capacity removable	fixed	Disc size (inches)	Stackable	Expandable	Disc format and systems supported	Back-up supported	Other comments
SS10R	Rohan	£2395	hard	10		3.9		yes	2 discs (5mbyte each). CP/M, Pascal	yes, removable.	Only slightly larger than standard Apple disc drive
SS15R		£2600	hard		10	3.9 and 5¼		yes	CP/M and Pascal	yes, integral	
SS5F		£1500	hard	5	5	5¼	yes	yes	CP/M and Pascal	yes	
SS5R		£1800	hard		5	3.9	yes	yes	CP/M and Pascal	yes, integral	
REAL R3SD	Realdata	£1395	hard		3	5¼	yes	yes	DOS 3.3 (single or three times floppy volumes) CP/M (logical drive size up to 8mbyte in units of 1.5mbyte). PASCAL (one to four volumes of 1.5mbyte).	yes, tape streamer or floppies	No extra RAM needed by drivers. Use-definable "SPLIT" operating systems DOS/CP/M/ PASCAL
REAL R6SD		£1690	hard		6	5¼	yes	yes			
REAL R12SD		£1995	hard		12	5¼	yes	yes			
REAL R20SD		£2495	hard		20.4	5¼	yes	yes			
Contour 3	C/WP	£995	hard		3	5¼	no	yes (by daisy-chaining up two drives)	One disc subject to volume limitations DOS : 400k. CP/M : 8mbyte. Pascal : 6mbyte. BOS (MPSL) 8mbyte. Any of these OS can coexist on the same disc.		
Contour 6		£1195	hard		6	5¼	yes	yes			
Contour 12		£1595	hard		12	5¼	yes	yes			
Contour 21		£1995	hard		21	5¼	yes	yes			
Contour 42		p.o.a.	hard		42	5¼	yes	yes			
X-COMP PHD	Pete and Pam	£2495	hard	20	16	5¼	no	no	Supports CP/M, Pascal, DOS and networking	built in	Can be added to Apple II, Apple III and other Z80-based micros.
Capricorn	HAL	£1750	hard		5-22	5¼			All systems used with Apple Basic DOS 3.3, Pascal (Fortran (1.1 and later) and CP/M.	to floppy streaming tape Amlyn 8mbyte disc	
Aquarius		£3299	hard		5-11	5¼					
Scorpio		£3299	hard		5-11	5¼					
SyMBfile	Symbiotic	£1350	hard	3	21	5¼			All support DOS, Pascal and CP/M. DOS looks like sets of floppies	2nd SyMBfile/floppy	
Model FD8/1D	Eicon Research	£1250	floppy	1		8	yes	yes	IBM Format (256k sectors). DOS enhancement – 1mbyte file volumes. Pascal and CP/M volume size 1 mbyte	yes	Compatible with all Apple software (BIOS is standard size) Auto recognition of single and double density discs.
Model FD8/2D		£1750	floppy	2		8	yes	no		yes	
DisCache 10		to be advised	hard		10.5	5¼	yes	yes	256K sectors. Volume Manager Program allows user to set own volume sizes up to 8 mbytes in any combinations.		Intelligent controller with 64K RAM for buffering, caching, back-ground print spooling, etc.
DisCache 20			hard		21	5¼	yes	yes		yes	
DisCache 40			hard		40	5¼	yes	yes			
DAS-11,	Tamarisk Design	£225	floppy		125K	5¼	yes	up to 4	Supports CP/M and Pascal	yes	
Data Box	X-Data	p.o.a.	hard		5 30	5¼	yes	yes	DOS CP/M Pascal	yes	Stand alone Winchester or Winchester floppy combinations
Kitten				10.6		8	no		Supports DOS 3.3, Pascal 1.1, CP/M 56K		Other interfaces available
Sphynx				10.6		8	yes		IISD reads as 4 x 0.25 mbyte		Other interfaces available
DP100				10.6		8			IIDD reads as 2 x 1mbyte		Removable disc cartridge
DP400					42.4	8			SOS 1.1, Business Basic, Pascal		
5006					5	5¼			Reads as 2 x 0.5 mbyte or 4 x 0.25 mbyte		
5012					10	5¼			DOS 3.3, Pascal 1.1		
5018					15	8			Reads as two drives in both cases. SOS 1.1		
RFD480			floppy	0.32							
RFD960			floppy	0.65							
RFD4000			floppy	1.0							
Megastor IISD	Vlasak		floppy		1	8					Two double sided drives
Megastor IIDD			floppy		2	8					
Megastor III			floppy		1	8					
Mios			floppy		1.25	5					For Apple II and III
Elite One	RANA	$849	floppy	163K		5¼	yes	up to 4 drives	DOS 3.3, Pascal 1.1, CP/M 2.20B		Write protect feature. Elite 2 + 3 have double sided heads.
Elite Three			floppy	652K		5¼	yes				
RO 101	ICE	£1350	hard		3	5¼			DOS, CP/M and Pascal	Image tape streamer sub-system (£1595)	Rotary head positioner, microprocessed control with on-board diagnostics. 2 chamber construction. Available for Apple II or III.
RO 102		£1600	hard		6	5¼					
RO 103		£2200	hard		12	5¼					
RO 200		£2500	hard		21	5¼					
Corona 406 Starfire	Corona Data	$2495	hard		5	5¼			Apple DOS CP/M, Pascal (up to 16 operating systems) Variable size volumes (up to 400K in DOS)	Floppy (with file compacting) auto-disc sequencing	9K interface card. Read-after-write and auto-back sector reallocation.
Corona 412 Starfire		$2995	hard		10	5¼					

THE invention and promotion of bubble memories some time back generated great excitement in the computer world. They are now appearing in the Apple market — one of the first being a disc emulator from MPC, a San Diego board manufacturer. MAX PARROTT and PETER BRAMELD looked at one of the first MPC boards to arrive in Europe.

Enter the bubble memory

THE MPC card is expensive, no doubt partly a reflection of newness, and it is not as fast as a RAM card. It is a measure of the product that despite having no manual when it first arrived we had absolutely no problem using it. We merely plugged it in slot 5 and accessed it with a PR#5.

Fortunately we had chosen an Apple with a lower case adaptor on board because messages were given in upper and lower case. We later tried it on an Apple without a lower case facility to see what happened. We got what we expected, upper case perfectly visible but lower case presented as a mixture of unintelligible symbols. The card offered a DOS 3.3 emulator with four files in memory. (We understand that the Pascal system and CP/M are also offered in other versions.)

The four files on the pseudo disc were an HELLO program in Applesoft, similar to the well-known system master HELLO in that it tries to LOAD the next file INTBASIC if a language card is present, INTBASIC, a BOOT13 utility, and a file development utility called FUD.

Having booted the pseudo disc, a pair of floppy disc drives on a controller in slot 6 were completely accessible for all normal DOS operations and FUD allowed files to be moved between slots 6 and 5. Round the other way, that is booting the floppies in slot 6 first, the pseudo disc in slot 5 was not accessible.

Attempting to CATALOG it, for example, gave I/O error. Obviously the way to use the device would be to have it in the higher slot number so that it boots on switch on and then have the floppies available as secondary drives in a lower slot. We tried this and all worked well.

The file developing program FUD is not a simple FID look-alike although it is somewhat similar to operate. There was one aspect of FUD we did not like. When copying a file from the pseudo disc to a real disc which already had on it a file of the same name, the program did not query the name but moved from drive 1 (the destination drive) to drive 2 and attempted to write there instead — a potentially dangerous situation.

Apart from this problem with FUD, we were quite happy with the system, although we found one minor curiosity. On attempting to CATALOG "drive 2" of slot 5 we were presented with a list of files identical to that on "drive 1" but with a different volume number. However, there was no "drive 2" and in fact we could SAVE a file to drive 2 and still have it on drive 1.

Clearly the DOS did not actually distinguish completely between drives 1 and 2 on the device. There was no similar problem with the real drives. The correct drives were always accessed except in the use of FUD as described above. All DOS operations were faster on the pseudo disc — the manufacturers claim operations are three times faster with a reliability over 1,000 times greater than that of tapes or floppies.

The memory offered on the card is the same as on a conventional floppy, and as we pointed out above is probably best used as a cold boot drive with your favourite program in its memory, also writing files to its memory and to a standard floppy as back up.

Here, however, is one major problem. The chances are that your favourite piece of software is a spreadsheet or a data handling or a word processor program which would really benefit from the increased speed and reliability of a bubble memory — but which is only available to

you the user on a copy protected $5\frac{1}{4}$ in floppy.

Quite how this problem is going to be resolved we don't know. Perhaps bubble memory devices with their own "protected DOS" will now appear.

While the bubble memory is being accessed the Apple emits a gentle clicking noise. We assume that this is the audible equivalent of the visual warning given by the red light on a conventional drive. We did our best to crash the pseudo disc's operating system but in the time available did not succeed. *(We'll keep trying!)* Clearly the manufacturer's claim of reliability is justified.

Well, where does this pseudo disc emulator sit among its peers? The main advantage is the non-volatile memory. This type of memory is very useful for data storage as it is virtually indestructible and has relatively fast access times. Whether 128k of storage would be useful to you will depend on the size and structure of your data management system. As a rough guide, 128k will store approximately 1,200 names and addresses, each containing 100 characters.

The benefits of having your most used program speedily available are difficult to assess. The DOS has the turnkey facility (the automatic running of a welcome program when the machine is switched on) and this combined with modular programs which access the disc frequently will benefit from an increase in speed as on this device.

Overall we feel that this is an interesting development and if the price does not deter you, worth looking at. As previously mentioned we did not have any instructions for the device and if we have omitted any important features we will report on them in future editions of *Windfall*.

RAM cards show a clean pair of heels

WITH the slow speed of Apple disc drives and their size, I was looking round for a bigger, better, faster drive. I'd heard about 128k RAM cards before and they seemed a good idea, but, at about the same price of a floppy drive, were they really viable? For that price wouldn't I be better off putting up with the slow speed of a floppy drive and keeping my files on non-volatile media?

The RAM cards themselves are generally arranged as multiple banks of 16k so they can be used instead of, or in addition to, a 16k language card. They come with their own driving software, which is the most important part of the package. This is the thing that lets them do work and be useful.

The amount of software supplied varies from make to make. You can have as much RAM as you like, but if it takes 10 years to program it to do anything, it's a waste of time and money.

Not all RAM cards are supplied with software that enables them to be booted and used in all environments. Those that do must generally be rebooted for each environment.

Although the advantages of the cards seem readily apparent it is worth remembering that just because they are RAM cards doesn't mean they work at RAM speed. The driving software takes time to operate, so the speed varies with the efficiency of the drivers. It can range from as much as 20 times to as little as three times as fast as a floppy drive. To put this in perspective a Winchester hard disc is about 10 times faster than a floppy.

Most RAM cards require you to boot an operating system image, then run a program to set up the pseudo disc. Once set up, the disc is still devoid of data and must, if required, have files copied across.

The time taken to simply set up and load the pseudo disc with files will probably be more than the time you save by using the pseudo disc even for a long disc intensive run. At the end of the run the data must then be transferred back to floppy if it is to be saved.

All this extra time could make the RAM disc uneconomic for some users, particularly those who only need short runs. Having said all that, the speed increase with some of the cards can change the use of packages involving word processing and accounts from a long, tedious bore to a quick and efficient operation.

One of the best suites of software to be seen with a RAM card comes with the Saturn cards. It enables the card to operate towards the top of the range at nearly 12 times the speed of a floppy.

The Saturn supports Apple DOS, Pascal and CP/M environments for a pseudo disc, but does need to be rebooted if changing between them. The software can make full use of any RAM cards plugged into the back plane up to a maximum of 256k giving a double-sized, fast disc capability but at about twice the price.

Using Saturn software is reasonably easy, although by no means automatic, except in Pascal. With Apple DOS a program must be run in order to set up the disc and use it. There must be another three files as well, and all are used by the set-up program.

Each time the machine is switched on this set-up program must be run in order to use the RAM disc. The program modifies DOS so that if you have to reboot without switching off, the set-up program must be run again to preserve the RAM disc's contents.

This makes it cumbersome in use but, to be fair, the vagaries of Apple DOS do not help any in this respect. There is also software available which will move Apple DOS onto the RAM card. Utilities such as FID and Muffin work with the RAM card as well as floppy discs.

By T.N. THOMPSON

Using Pascal on the card involves a complicated set-up procedure similar to DOS, but is a once only thing. Once the environment is set the pseudo disc is automatically initialised on boot.

To make Pascal even more automatic, there is a program that on boot will automatically load the RAM card with marked files and which can be inserted before your auto boot program. Then the files are loaded onto the RAM card and your program is run. When finished with the card there is a very fast copy program which dumps the contents of the RAM card to a floppy. On the whole, a very civilised system.

Like Pascal, CP/M's RAM card driver has a once-only set-up program. This is the only software provided by Saturn for CP/M (as they rely on SUBMIT files and PIP to copy files to RAM) which is unfortunate as CP/M would really benefit from a fast copy to RAM facility.

The set up program generates two .COM files: INIT.COM and INIT2.COM. The first initialises the RAM disc and the second recovers a crashed RAM card.

I consider it very worthy of Saturn to provide these back up and recovery programs. However the only trouble I had with the whole system was when I made extensive use of these programs using Apple DOS. I have never had to use the recovery programs with either Pascal or CP/M in three months' heavy use, as both have worked flawlessly.

The manual for the Saturn card is very good, being easily readable by the most inexperienced user and yet not boring to an expert. All the information you could require is included. It is a credit to its author.

Just before this article went to press I had a brief encounter with the Ramex 128k RAM card. The software supplied provides a number of facilities for Apple DOS and a pseudo disc for Pascal, but none for CP/M.

While using the Ramex card as a disc emulator the speed increase is just over 10 times that of a floppy drive.

In Apple DOS the System Master must be booted, followed by the supplied disc. The program which then runs places an enhanced DOS on to the card which can then be saved and booted by INITing a disc.

The new SoliDOS has extra commands which allow the card to be INITed to a pseudo disc, to fast load the card, dump the card's contents to floppy, and write-protect the card. There is a lot more, including the ability to store and recall vast arrays.

I found this new DOS a great improvement over the standard, especially as most of the new commands work with floppies as well as the RAM card. I also thought it a better, more automatic implementation than the Saturn.

As it is an enhanced DOS not requiring extra software to control the card, I had no problems at all with data loss, unlike the Saturn Apple DOS.

The Pascal implementation works in exactly the same way as with the Saturn system, but neither card is compatible.

The Ramex documentation was not up to Saturn standard. I found it tried to explain complicated subjects in a simple manner and simple subjects in great depth. The result — a brief manual that makes heavy reading.

With the availability of CP/M on the Apple and the enormous amount of software available, it is a major failing on the part of Ramex not to include any software for the CP/M environment in their package. I feel they should eliminate this gap as soon as possible to make the package complete.

Both cards have their good and bad points. They are expensive and their performance does not live up to the image of fast RAM. On the other hand some users may find the increase in access speed a boon to their operations.

The cards give the speed performance of hard discs, but much reduced storage capacity, for a lower initial cost.

Don't let back-up get you down..

By PAUL RAYNER

FLOPPY discs are marvellous things. They store hundreds of thousands of characters on a piece of plastic smaller than a postcard. They are left lying around on dusty desks; they are rammed in brief cases; they are consigned to the tender mercies of the Post Office; and still they deliver back our data at 192,000 bits per second.

There comes a time, however, in every disc's life when it stops performing reliably. With some of the cheaper brands this can be after only a few dozen accesses. With expensive brands it can be after many hundreds of thousand accesses. But whatever discs you use, and however well tested your programs, one day you will get a disc error.

Usually you will know pretty quickly. The computer will stop and print an error message, for example in CP/M:

BDOS ERROR ON B
or ERROR EF

When this happens you will find out just how reliable your back-up system is.

The idea behind most back-up arrangements is to take copies of your data files at regular intervals. Then if for any reason your current files get destroyed or corrupted, you can revert to the copies. You will, of course, need to re-enter all the changes since you made the back-up copies in order to bring your files up to date again, but in a well designed system with frequent back-up copies this should not take too long.

Exactly how often to take back-up copies depends on factors such as how much data you are entering, and how much time and money you are prepared to spend on back-up.

For example, if you take copies at the end of each day then you should only have one day's business to re-enter in the event of a disaster. However making back-up copies takes time. Thus you have to choose between the time spent in making back-ups, which hopefully will never be needed, and the time required to get everything back up to date when you do have to use your back-up files.

Another factor to consider is the processing sequence of your programs. If, for example, certain operations only take place at month end, then certain types of error or file corruption may only be discovered at the next month ending. Thus it is essential to keep copies of the files as they existed before each month end run. In these circumstances a daily back-up routine on all files would require more than 30 generations of back-up.

Keeping track of all these discs would

Paul Rayner is managing director of Great Northern Computer Services

probably require a computer system dedicated entirely to this. A more conventional approach in these circumstances would be to take weekly back-ups. Less time would be taken while making copies and the number of discs would be much more manageable.

The process of copying back from your back-up discs is known as "backing down". It is important to back-down rather than just to use your back-up discs in place of your normal ones. It could be a bug in your programs that is causing the error and if you use these programs on what had formerly been your back-up copies you will corrupt those discs as well. When the programmer finally discovers and corrects the fault you will still be in difficulties because you now have no back-up.

Obviously, if you are going to have to re-enter all the transactions since you made your back-up copies you must have some way of knowing what these transactions were. Keeping the original source documents in neat bundles is one way. The more usual way is to bring out audit reports whenever transactions are entered and to file these reports as part of your normal operating procedure.

It is crucial to have some clearly understood system controlling the back up

discs, otherwise when you come to back-down you could end up by backing down from the wrong ones. A normal procedure is to have a series of "generations", often referred to as "son", "father", "grandfather", etc. These discs will be rotated so that the one containing the oldest copy will be over-written with the latest back-up copy. Thus "great grandfather" will become "son", "son" will become "father", and so on.

To control all this you need to be able to easily identify which discs are which. Firstly, attach a sticky label to each stating clearly which files are contained and the date that the copy was made.

Then give each disc a unique identifier (a number or letter) or else use a colour code. Disc suppliers always provide a bundle of multi-coloured sticky labels with their product and you can use these to create a "red" set, a "yellow" set, a "green" set.

Finally you will need to keep a register of back-up copies as you can be sure that problems will only occur when you are on holiday, off sick, mending your son's hang-glider, or whatever; and your temporary replacement needs to be able to see at a glance when the last back-up was taken and which discs it is on.

These broad principles apply whatever type of computer you are working on. On the other hand, the exact details of which files to copy and when to copy them will depend upon the exact nature of your programs and your hardware. Your dealer should be able to advise you.

Mini-Winnies boom

ONE of the biggest talking points at last month's Comdex computer trade show in Las Vegas was the dramatic increase in the variety and range of memory storage systems now available.

Most remarkable of all has been the sudden growth in the 5¼in Winchester disc drive market, which only opened in 1980. The industry now expects these ultra compact drives to make up 50 per cent of total fixed disc drive sales by 1986.

Several companies are now offering 3½in minifloppies. Shugart's stands 1.6in high and occupies about a quarter of the volume of a standard floppy. It provides 500k unformatted capacity, 135 tracks per inch density and six millisecond track-to-track access time, and uses discs protected by a hard plastic shell.

Santa Clara Systems showed their new 3.9in removable Winchester cartridges — new technology which provides a low-cost method of backing up important data with low power consumption, and also allows for virtually unlimited storage capacity.

Half-height Winchesters and floppies were also to the fore. Vista call their Thinline Quad Subsystem the final evolutionary stage in Apple disc storage, offering the capacity of four Apple drives in the space of one conventional drive. This is achieved with two double sided Thinline discs, providing more than 640k of storage.

Rana have brought out a floppy disc drive that is claimed to give four times more storage capacity than Apple's own drive, plus a much faster access time. And that's getting pretty close to hard disc performance.

New drives offer massive storage

TWO high-density floppy disc products which provide new mass storage options for the Apple III will be available in the spring.

Apple say the drives ensure greater integrity of data than other high-density drives by way of a unique, double-sided mechanism.

The products, UniFile and DuoFile, contain the new drives in a single and double configuration; each drive provides 871k of on-line storage for the Apple III.

They are designed to serve the needs of managers of small to medium-sized businesses, professionals, software developers, OEMs and others who need mass storage for applications such as accounting and database management. UniFile and DuoFile complement the 140k disc III floppy disc drive, which is built into the Apple III, and ProFile, a self-contained five mbyte mass storage system introduced by Apple in 1981.

UniFile also is intended to serve as the ideal backup for ProFile, which allows the user to backup to one location information that would fill 35 floppy discs. With a formatted capacity of 871k, it requires only six discs to back up the total ProFile contents.

It also takes advantage of Backup III, a new software utility that selectively backs up and restores ProFile's files. Backup III will now be included in each ProFile system package.

DuoFile provides Apple III users with an additional mass storage alternative. It contains two Apple 871 disc drives, with a capacity of 871k per drive, thus providing built-in backup. Both UniFile and DuoFile offer greater data security over hard disc drives since the discs are removable.

The systems are equipped with a doorless self-ejecting mechanism that eliminates the chance of errors that occur when disc drive doors are opened while writing data. UniFile is 4.39in high by 8.54in wide by 8.81in deep and weighs four pounds. DuoFile has measurements identical to ProFile — 4.39in high by 17.28in wide by 8.81in deep. It weighs eight pounds and is designed to fit between the Apple III and its monitor.

The Apple 871 disc drive is one of the first high-density, double-sided mechanism to be built specifically for personal computer applications. It

DuoFile (left) and UniFile

represents three years of research and development and has undergone extensive testing and design verification during the past year.

"We expect it to make a significant contribution to the personal computer industry," said Barry Yarkoni, marketing manager for Apple's Peripheral Systems Division.

"It combines the best aspects of what has been learned about floppy disc drives over the past 10 years with the latest in floppy disc technology — advanced heads, innovative design, and improved media. Apple's design overcomes some of the fundamental limitations of previous floppy disc drive technology."

The mechanism positions read/write heads on opposite ends and sides of the disc, reducing media wear and increasing data integrity. The system also utilises eight electronically-controlled motor speeds that vary from track to track, maximising data capacity.

Disc drive systems with one speed require the bit density to be less on the outer tracks. The Apple 871 solves this problem by varying the motor speed and using equal density on each track.

The disc drive system utilises an intelligent controller that performs functions such as reading and seeking, and that provides data transfer at a rate of 62k per second, four times faster than conventional floppy disc drives. The controller card plugs into one of the Apple III's expansion slots.

UniFile and DuoFile use Fileware discs, a new 5¼in floppy media developed by Apple in conjunction with major media suppliers. Each contains 62½ tracks per inch, providing six times the capacity of the single-sided, 48 track disc used with conventional 140k drives.

The increase from 48 to 62½ tracks per inch is a conservative boost that provides additional storage without sacrificing data integrity, and that requires a less complex and less expensive positioning mechanism than discs with higher densities.

Fileware accommodates a wider temperature range than many high density discs, further increasing reliability. Because temperature changes can cause the media to expand and shrink, the read/write head on discs of more than 62½ tracks per inch can be sufficiently misaligned to cause reading errors.

"Apple is continuing to enhance its computers with the most advanced and reliable mass storage devices. UniFile and DuoFile complete the first generation of a family of mass storage systems that fully support the Apple III in a wide variety of applications," said Yarkoni.

The Apple 871 disc drive will be manufactured by Apple and a second source. Fileware will be manufactured by Apple and other major media suppliers, including Verbatim and 3M.

UK prices have not yet been finalised, but in the United States UniFile will retail for under $1,000 and DuoFile will sell for less than $1,700.

TRON

From the heart of an Apple springs an electronic odyssey

HAVE you ever day-dreamed about the mysterious, electronic civilisation that might exist deep, deep inside the central processing unit of your Apple?

A whole new micro-world where fantasy becomes fact, where energy lives and breathes and where every law of logic is defied.

Such thoughts came to writer-director Steven Lisberger in 1978. A devotee of video games, he had just produced a multi-million dollar animated film called Animalympics and was looking for new fields to conquer.

One summer's day he mentally assumed the role of a classic character from an earlier Hollywood era, the incredible shrinking man, and imagined himself as a microscopic speck escaping from forces of evil lurking in the heart of the Apple, which was just then beginning its meteoric success story.

After two years' gestation, Lisberger and his partner, Donald Kushner, took the idea to Walt Disney Productions, where it received an enthusiastic welcome. The result was the futuristic adventure TRON, which went on general release last month.

Quite naturally Apple Computers Inc gets a credit line in the film. So do two other big names in the Apple add-on field, Mountain Hardware and the Syntauri Corporation.

The story starts with the hero sitting at his Apple III writing a program that would get the better of an unscrupulous Master Control Program which was trying to take over the world.

The fact that he was not having much luck was either a reflection of the power of the MCP or the inadequacies of the Apple III. Any beginner who has had to struggle with the III's Sophisticated Operating System will know exactly how he felt.

The hero is so frustrated he decides to incapacitate the user-unfriendly MCP. Instead he finds himself blasted into electronic particles and wakes up in another dimension where computer programs are electronic warriors, the alter-egos of the programmers who created them.

Says Lisberger: "Everyone's looking for new fantasies in the movies. Outer space has been done to death. They've gone inside the body and under the sea. We've created this new world in TRON by taking video games and just blowing them out to the point where they become a reality.

"When games met computer graphics something came alive that hadn't been alive before. Video games were the basis for the fantasy. Computer imagery was the means to create it."

Computer imagery had its birth in the computer graphics of the mid-1960s, when workers in aerospace and scientific research found that simulating objects digitally proved more effective than building models. Today it's the hottest tool in cinematic special effects.

TRON is the first feature film to make extensive use of computer imaging, such as a cityscape at night metamorphosing into computer circuitry pulsating with energy.

Describing how it is done graphics pioneer Larry Elin likens it to the way they light the giant electric poster on Times Square in New York. "However that has maybe 8,000 lights, while one of our frames can consist of more than two million pixels of light," he said.

Those two million pixels — each with its own colour and intensity values — multiplied by the 24 frames it takes for one second of film, adds up to nearly 100 million bits of information for a brief period on the screen that would be missed in the blink of an eye.

Fifteen minutes of the 100-minute film consist of wholly computer-generated sequences and cost a quarter of the $20 million spent on TRON.

The film is breaking so many records in the United States that the pundits are already saying that Hollywood will never be the same again. But then they said that when Disney's Steamboat Willie first tooted down the silver screen Mississippi in 1928.

"Sure we've taken risks with this film," said Lisberger. "We've broken with convention. But computer imagery is never going to replace actors. Actors are what I call the ultimate special effect."
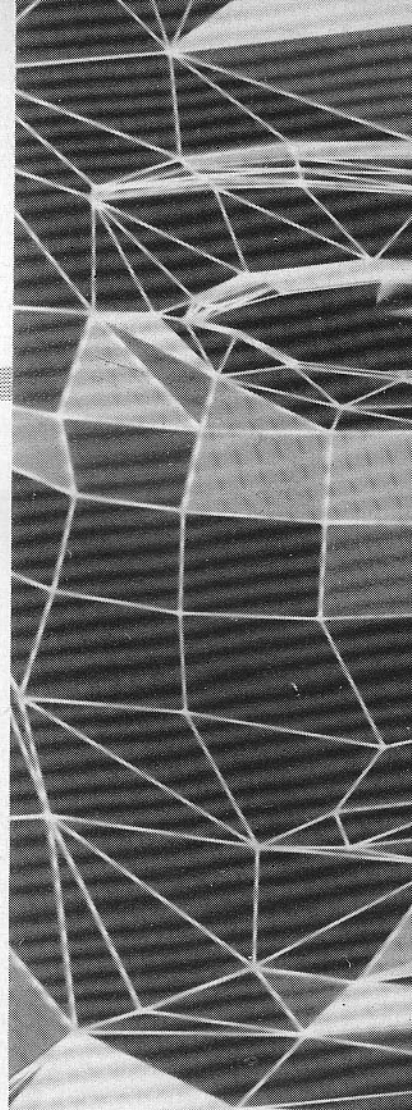
The Master Control Program, sinister

.. and a new line in computer language
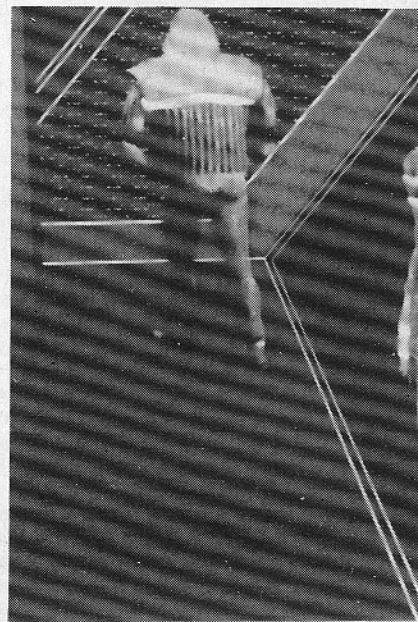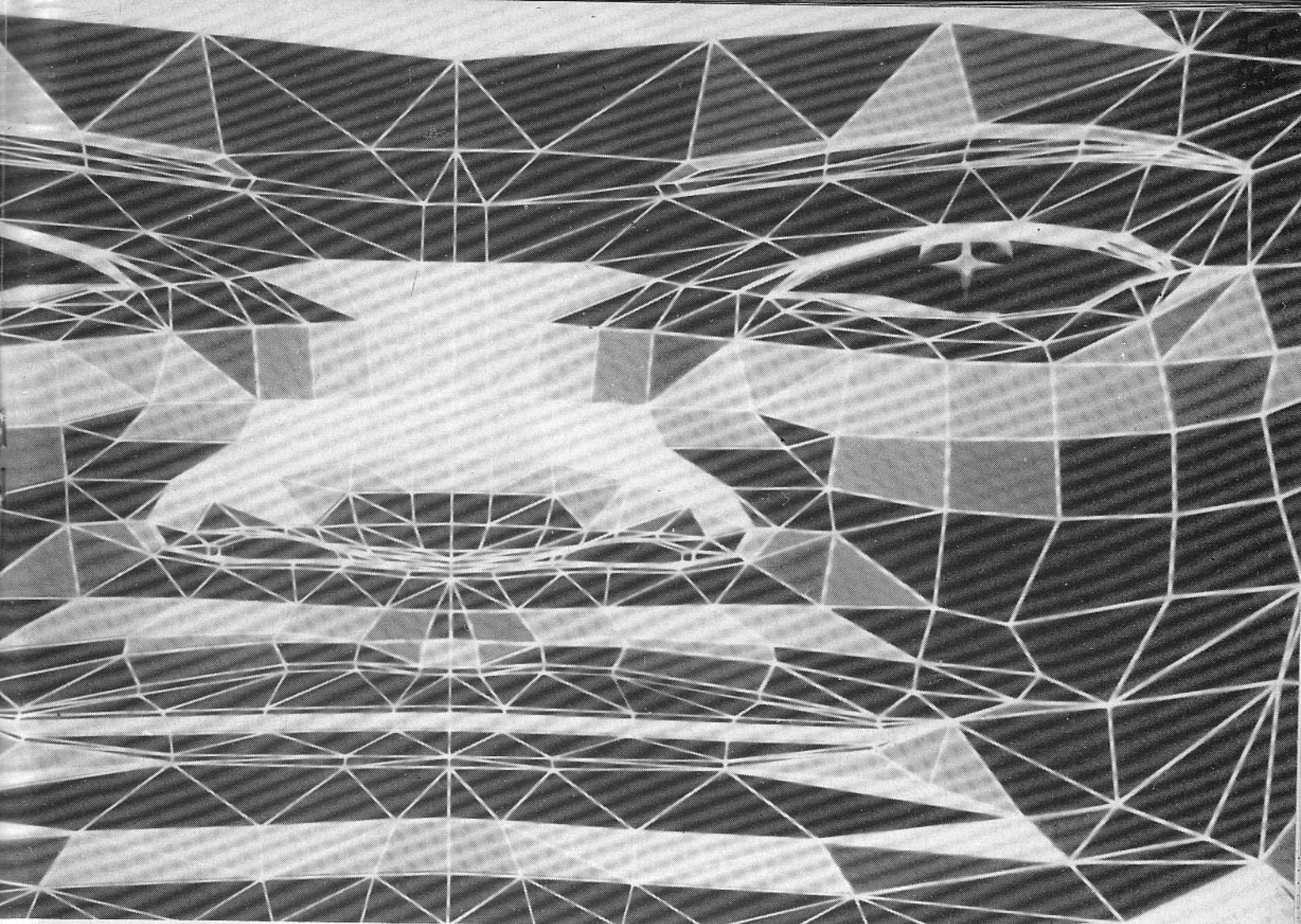
of the world of TRON

CPU – The geographic area within the electronic dimension where the MCP is located.

CROM – An overweight, out-of-shape compound interest program caught and made to play video games by the MCP forces.

DIGITIZE – To enter the measurement of some physical variable into a computer's memory as a series of numbers. Theoretically, it is possible to re-construct a physical entity, a DNA chain or an orange if the measurements are stored in enough detail.

DOMAIN – A geographic area of the electronic dimension.

GAME GRID – The arena where computer programs play video games to the death. The electronic dimension's equivalent of the Roman Colosseum.

GRID BUG – Spidery creatures with the ability to disguise themselves as part of the game grid.

IDENTITY DISC – Used as a means of storing information and thrown as a weapon, a disc is worn on a program's back. A program gets information encoded on a disc by sending it to a User via an I/O Beam.

INPUT/OUTPUT TOWER – The structure where programs communicate with their Users.

LOGIC PROBE – A beam of electricity which can force a program to yield the information it possesses.

MCP – The tyrannical force which demands subservience from all programs in the system.

PROGRAM – In the real world, a set of instructions or coded information entered into a computer system. In the electronic dimension, a being with human qualities created by a User, or programmer. Programs are the alter-ego of the people who create them.

RECOGNISER – A towering robotic entity which flies from domain to domain, looking for programs wanted by the MCP.
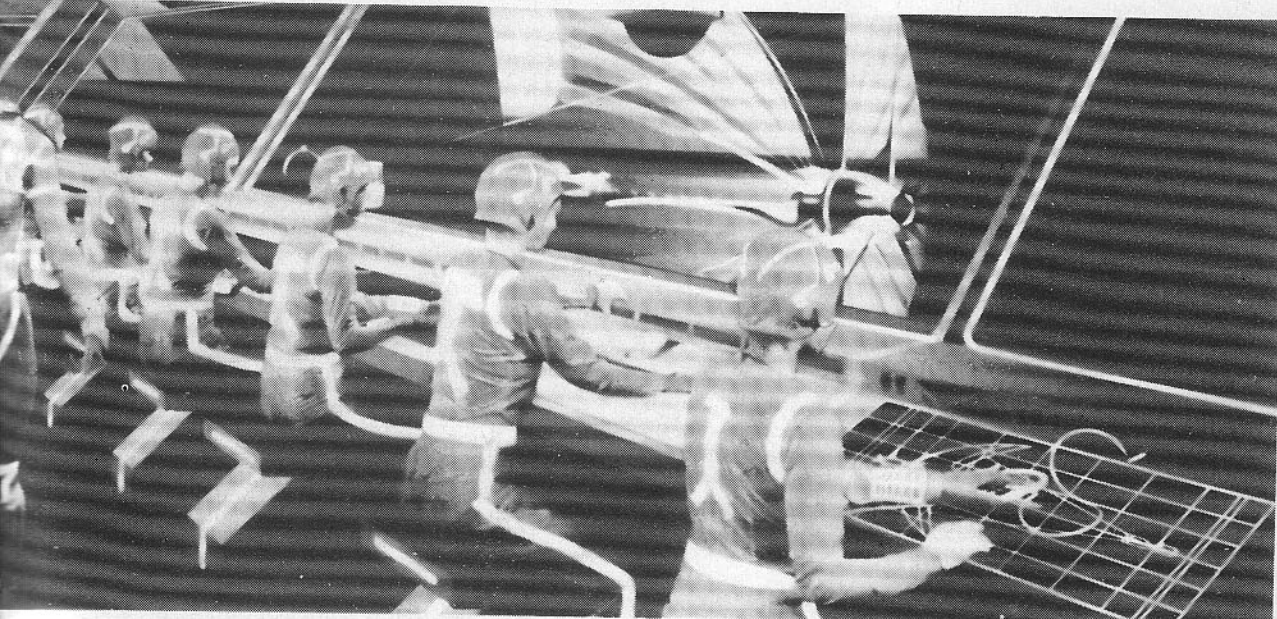
A computer program takes on an entirely new meaning in the shape of Yori (Cindy Morgan), here overseeing construction of a computer-generated solar sailer.

GRAPHMAGIC

It's as easy as pie charts

ANYONE who handles large amounts of numerical data will at some time want to represent it graphically, as a graph can often convey trends and variations in a more informative and lucid manner than a table of figures. However, drawing accurate graphs by hand is time consuming and so any alternative is welcome.

Among the software packages which utilise microcomputers to rapidly transform numerical data into graphs is ISM's GraphMagic which boldly proclaims that "the user-friendly graphics era comes to your Apple."

In common with so much commercial software today, using GraphMagic is rather like conducting a tour of local restaurants. The user is launched into a world of menus and sub-menus which fortunately, as in this case, generally leads to smooth operation of the program.

GraphMagic is supplied on a protected disc and is accompanied by a useful, mainly non-technical user manual. Generating graphs with it is a simple process, and after a quick glance at the clearly written manual I was able to boot the disc and display a pie chart within minutes.

Three types of graph can be drawn: pie chart, histogram and line. Data input is achieved via a flexible input editor, which refers to numerical data as variables which must consist of a name/value combination.

The editor allows new variables to be created or old ones to be loaded from disc and can manipulate up to 40 variables at a time.

To draw a basic graph all that is required is a set of variables. As an example of this, Fig. 1 was created by entering nonsense data and selecting the pie chart drawing option. The program automatically labels each segment of the chart and represents each variable as a per cent

By GRAHAM PLACE

of the total.

Basic graphs can be enhanced in several ways. Two lines of text can be inserted as a title at the top of the graph and may be justified to the left, right or centre of the screen. Users with colour monitors also have the option of displaying graphs in a variety of colours.

One of the more powerful capabilities of the package is its full screen text editor, which allows text to be added to any part of a graph. Since graphs are drawn on the Apple's high-resolution graphics pages, text is created by "turning on" appropriate pixels.

Cursor control at a crude level is essentially as normal, but the text editor also has cursor position fine-tuning com-

mands. These move the cursor one pixel at a time left, right, up or down.

This function can be used to create bold face text by typing it twice with the cursor shifted one pixel on the second time of typing (see Fig. 2). It is possible to create any shape desired with imaginative use of the character font.

Incidentally, the text editor can be operated without actually drawing a graph, and so can be used as a crude word processor.

Histograms and line graphs drawn by GraphMagic can be modified in several ways. The x and y axes can be labelled, statistical data — calculated by the program from the variables in memory — can be superimposed on the graph and the scales of either axis may be altered.

With the latter option, altering the default values (1 to the number of variables) of the x-axis shifts the y-axis to the right, leaving its scale hanging in space. This gives poor presentation (Fig. 2).

Several modes of display are possible with histograms and line graphs (Figs. 2 and 3) including superimposing a grid of horizontal and vertical lines. On doing this I found that the vertical lines did not correspond with the tic marks of the x-axis although this should be an easy bug to rectify.

Graphs or variable sets created using GraphMagic may be saved to a data disc. Before starting any serious drawing, the user specifies which disc drive is to be used for data storage and retrieval. Ideally the system should have two disc drives, although the program will operate with one.

To prevent valuable data from being corrupted or lost, user numbers may be allocated. This is a useful option if several people wish to use the same data disc since the formatting of the file names incorporates the user number and thus avoids file name conflicts.

Hard-copy of graphs may be obtained using a printer capable of graphics. Again, the system set-up has to be specified and this includes indicating the type of printer, slot number of the communication card and the column print width desired.

A limited number of printer types are catered for, although if your particular printer is not specified this should not present a problem. GraphMagic saves graphs by utilising the Applesoft BSAVE routine and so files are easily accessible from the data disc.

For the non-technical user, ISM points out that a number of utility programs are
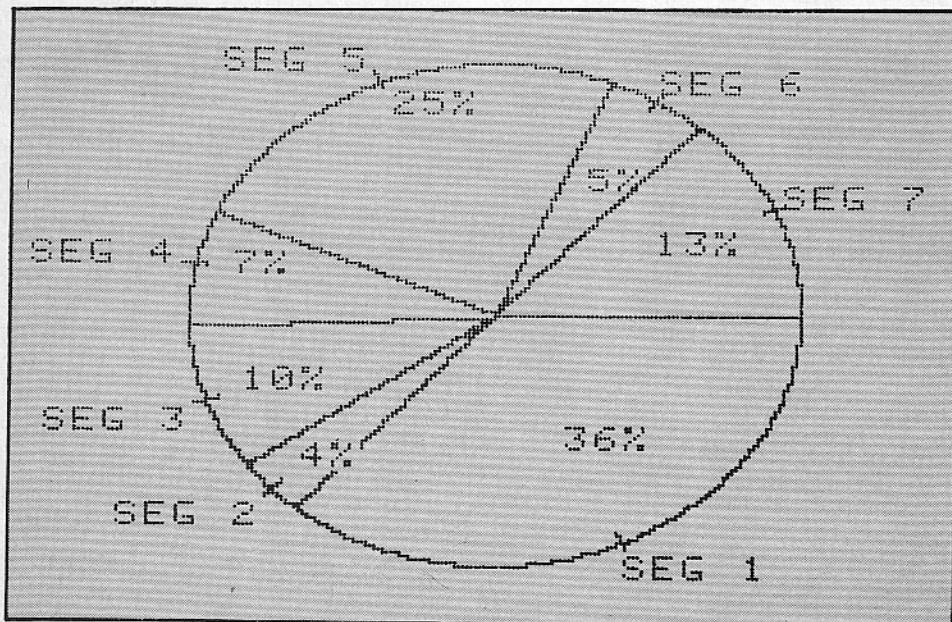
Fig. 1. Basic pie chart drawn by GraphMagic. Highlighting of any one segment of the pie is possible, enabling data of particular interest to be emphasised.

available which will plot a memory image file of graphics to various combinations of controllers and printers.

The default printer of GraphMagic is the Epson MX-80 which was also used for the figures in this review. However, graphs dumped on this printer have a flattened appearance which is particularly unsatisfactory with pie charts because they become squashed into an oval shape (Fig. 1).

This appears to be a common fault associated with the Epson parallel interface card. ISM acknowledges the problem and is apparently trying to solve it at the software level.

GraphMagic can be used in conjunction with other programs, in particular MatheMagic (another ISM product) and VisiCalc. In fact GraphMagic can read any data files saved in DIF format.

An "interface to DIF format" option is provided enabling tabulated data, created by using VisiCalc for example, to be graphed.

With minimum effort, any graphics created with other programs could be enhanced by using the text editor and other options available in GraphMagic — very useful.

What about performance? On the whole I found the package simple enough to operate and it certainly lives up to its user-friendly tag.

Error trapping is excellent and any errors occurring due to user, hardware or software failures prompt the display of appropriate error messages. The meanings of these messages are explained in a comprehensive appendix in the manual
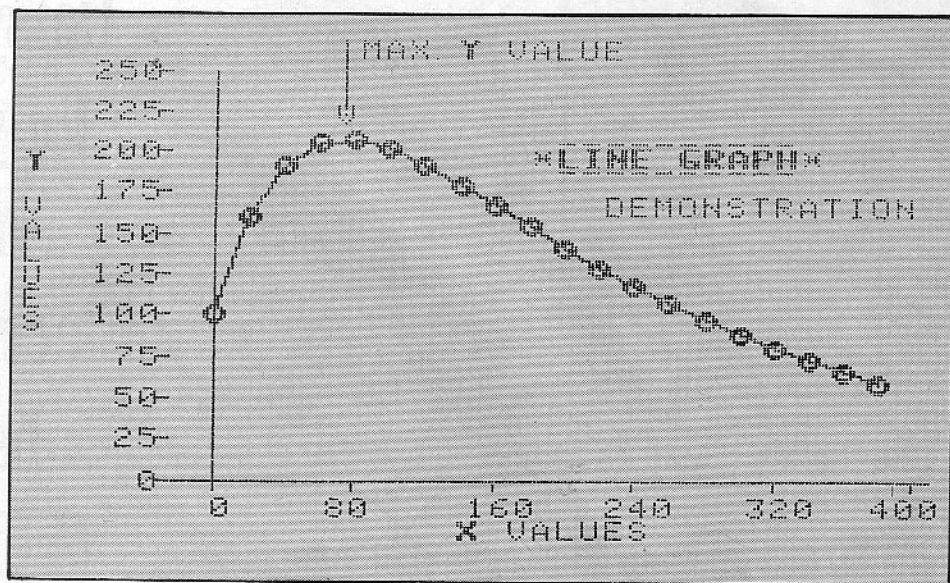
Fig. 2. Line graph representation of a bi-exponential function. This display incorporates text created using the full screen text editor. In addition to plotting points and drawing a line through them, the area between the base of the graph and the line may be filled in.

which also advises on the best routes of recovery.

The formatting of each type of graph is basically very good, although two points should be noted. First, when labelling axes it is often necessary to enclose certain information, such as units, within brackets. With GraphMagic this is simple enough when labelling the x-axis but the mode of display of y-axis labels, i.e. reading from top to bottom (see Figs. 2 and 3), prevents sensible use of brackets.

Perhaps the user could be provided with an option allowing "sideways" printing of y-axis labels, which should be within the scope of the flexible text editor.

I did note that when required to print the % character as part of a y-axis label the program crashed — an odd little quirk since every other character prints perfectly well.

My second point concerns line graph representation which could be improved substantially by permitting x,y data pairs to be plotted.

The nature of GraphMagic's variable input routine, i.e. name/value combination, means that only truly variable y-values can be plotted while x-values simply represent an arithmetic series. The incorporation of a numeric input routine especially for line graphs would be a useful addition to an already extensive repertoire.

These criticisms accepted, GraphMagic is a user-friendly program capable of generating acceptable graphical representations of numerical data.

● *A 12-year-old Stockport schoolboy, John Stevens, also liked the package, and sent us his review under the title "Draw your own conclusions":*

"This is an extremely useful package for anyone who keeps having to draw loads of graphs, because it saves a lot of proper graph paper (which these days is a lot of money.)

"GraphMagic can be used with MatheMagic and VisiCalc, and it enables you to draw information on a pie, bar or line graph, adding colour where required and highlighting certain parts.

"It is a great graph drawer — and I don't really see the need to have the instructions, because it is so easy to use." ●
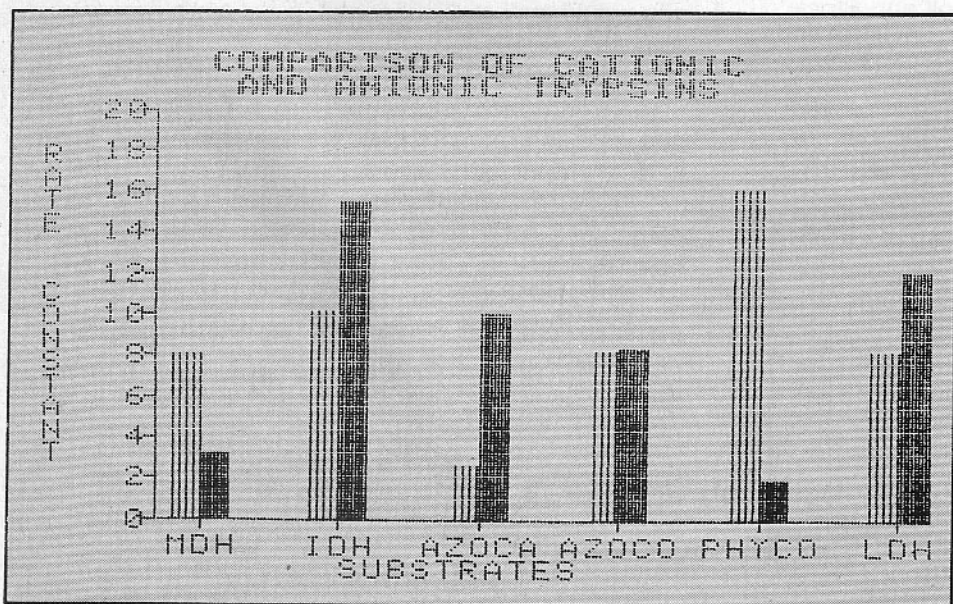
Fig. 3. Bar graph of real experimental data. The bars may be justified left, right or centred, which allows one graph to overlay another.

COMPUCOPIA
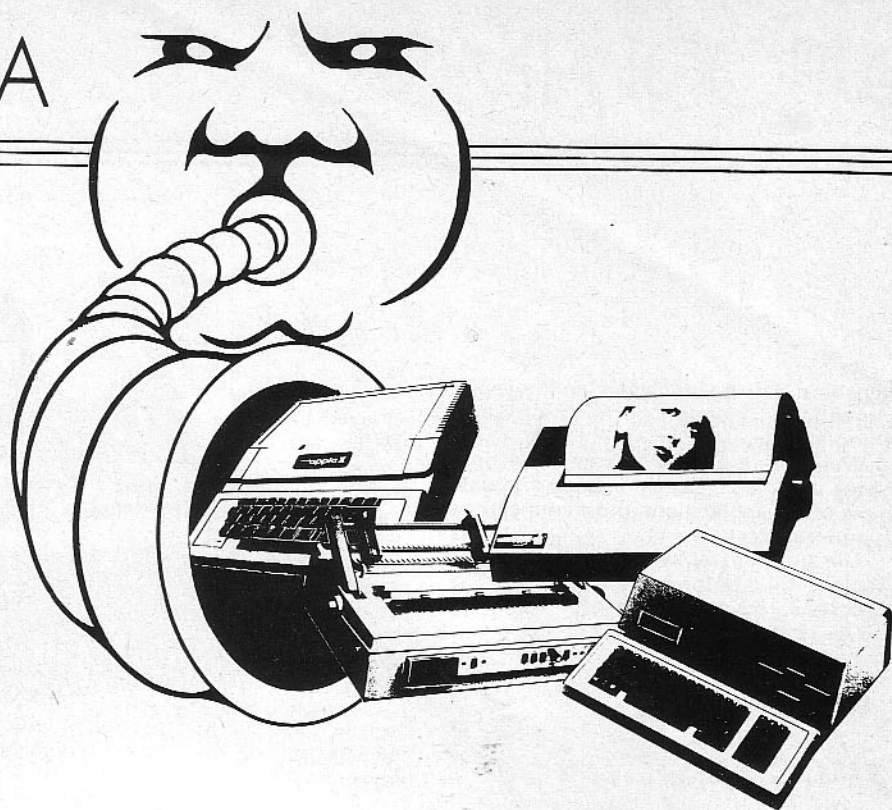
Assistant for budding Brahms

THE American-based Syntauri Corporation, which describes itself as "the pioneer of soft instruments-programmable micro-computer-based digital instruments", has released Composer's Assistant, a powerful software tool which allows automated transcription and hard copy score printing.

In use a composer performs on the AlphaSyntauri keyboard and his music is stored in up to 16 separate polyphonic tracks. It may then be viewed track by track on a video screen or printed out.

The company says it is designed to meet exacting professional standards and provides a fast, accurate and inexpensive way for musicians to translate their ideas into hard copy for reviewing, orchestrating and copyrighting.

Any polyphonic piece performed on the Syntauri keyboard may be transcribed into notation similar to piano scoring. The system also incorporates the newly-released Metatrak 16 track recording system which lets composers work with full orchestral ideas and then produce instrument-by-instrument video displays and printed copy. Each track can be polyphonic and any may be chosen for printing.

Composer's Assistant caters for the fact that personal playing styles vary, and can adjust for performance timing variances such as leading the beat.

Additional features include a 1/16th note and rest resolution, special triplets mode, variable accidentals and measure tie marking. A text-editing facility allows the addition of expressions, special instructions, lyrics and chords.

Californian synthesist/composer Kentyn Reynolds developed the program because, he said: "I wanted direct visual feedback of my keyboard impressions, independent of the sounds. The Syntauri system — the keyboard synthesiser, mul-

titrack recorder and transcriber — lets me capture my musical intuitions, then later review the theoretical aspects of what I was feeling as I played."
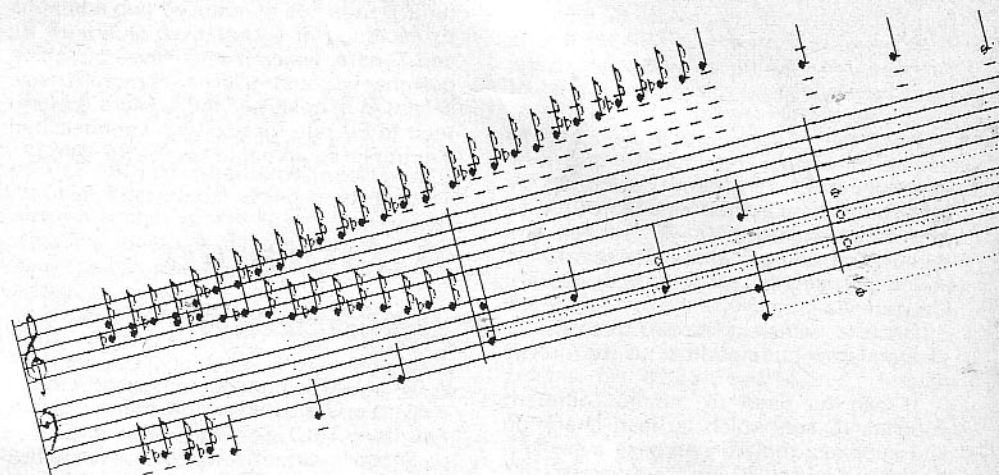
The software costs $295. It is designed for use with Syntauri synthesisers and a 48k Apple system with printer. The full five-octave Composer's Workstation, incorporating the software and equipment, sells for around $2,100 in the US.

★ A review of composer's Assistant and Metatrak is planned for a future issue of *Windfall.* It will also evaluate the rival Sound Chaser system and look at other developments in micro music.

Another spreadsheet

A NEW electronic spreadsheet has been developed by Microsoft. It is called Multiplan, and has many of the standard Visicalc functions and capabilities with a few extra features thrown in for good measure.

As with Visicalc, Multiplan automatically re-calculates a worksheet when new data or formula are entered, but instead of referring to entries by a grid reference code it allows a user to assign names (such as Sales or Costs) to an entry or area of the worksheet.

Program prompts tell the user what to do next, commands are listed in English, and a help function gives access to a reference guide at any time for information on any command or function being used.

Another facility permits linking information in different worksheets to instruct the Apple to make adjustments automatically on related worksheets when changes are made to a primary sheet.

Entries can be made either alphabetically or numerically and results of a formula can be expressed in either mode. Microsoft claim that using the program's conditional capabilities complex rela-

tionships can be analysed and relevant sentences chosen and displayed automatically.

Worksheets can be broken down into pages of variable size and shape and data from other applications programs can be incorporated.

The program is available for standard Apple II (64k) or for Apple II and III CP/M. It costs £169 from Pete and Pam. *Tel: 0706-227011.*

Avoiding hiccups . .

HAVE you ever had your computer or sensitive electronic equipment hiccup, malfunction or lose data? Was it equipment malfunction, or the result of mains borne interference, transients or noise?

Transients and noisy 240v mains supplies can play havoc with computers, disc drives and many other types of electronic equipment.

A portable monitor unit designed to detect transients and indicate that they have occurred, has been developed by Mektronic Consultants. The Sentry plugs into any standard 240v AC socket, and detects transients at three different levels indicating slight, moderate or severe interference.

Once a transient is detected an indicator lamp remains lit until the unit is reset.

It can be used to monitor different supplies to see which is best, check on site supplies, monitor supplies at exhibitions or when giving demonstrations, identify equipment which generates noise and to monitor supplies overnight and at weekends.

It costs £45. *Tel: 061-798 0803.*

. . and sundry static

WITH the development of the electronic office static electricity has become more than just a nuisance — it can be a mortal hazard for the silicon chip.

The combination of central heating and air conditioning has made offices warmer and drier, which are perfect conditions for the generation of static.

Simply walking across a carpet can create a charge of anything between 5,000 and 10,000 volts, depending upon the carpet fibre and the type of soles on one's shoes. Even the combing of a person's hair with a plastic comb can create a potentially chip-killing charge.

Mainframe manufacturers do build static protection into their hardware. Micro-based offices might consider
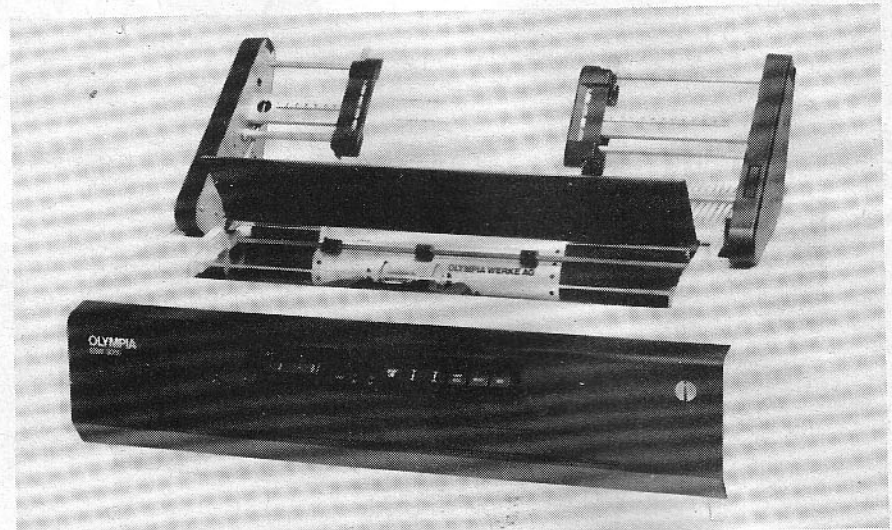
The Olympia ESW 3000

buying anti-static carpets specially engineered to do the job. Carpet manufacturer Heuga has announced two additions to its range of carpet tiles, Quantum XL and Tundra, which it says, gives the office designer or end user a comprehensive selection of textures and colours guaranteed to be safe for use with sophisticated electronic equipment. *Tel: 0296 84631.*

Adaptable database

A DATABASE management system that is easy to operate and simple to learn is File-Fax from TMQ Software of Illinois.

The program offers quick access to files and records and TMQ says it has a unique, powerful and easy-to-use report generator.

It is written in machine language and is completely memory based so that once loaded a user doesn't have to swop discs.

Features include numerous text-editing facilities, help screens that can be called up at any time and an eight-level sort and wide-ranging search capability.

The database itself has no fixed format and can be set up according to individual user requirements. *Tel: (from UK) 0101 312-520 4440.*

Robot for your Apple

A TEACHING tool that can demonstrate the operation and control of an industrial robot — that's the Mitsubishi Movemaster RM-101, a compact and portable micro-robot, only 25cm high and weighing 10kg.

It has six axes of movement and five degrees of freedom. There is a stepping motor positioned on each joint giving

open loop control and the motors are geared down to give repetition of positions to within 0.3mm.

Mitsubishi claim the micro-robot's intelligence is comparable to that of its big brothers, the industrial robots. It consists of a Z80 microprocessor with 16k RAM and 8k ROM. The ROM is programmed with two test modes so that operations can be checked quickly without being connected to a microcomputer.

The RAM enables it to remember 100 positions, but by extending RAM by 6k the number of stored positions can be boosted to 600.

The micro-robot can be programmed in either Basic or Assembler to give direct control or when using Basic, subroutines in the robot's memory can be called up to give internal control.

Maximum load is 500 grams and top speed is 7cm per second at the tip of hand. Computer control gives straight line movement between programmed positions. The robot links to an Apple II using a standard Centronics interface on an Epson printer card. *Tel: 0923-37333.*

Records system

AN incomplete records system from Number One Computers includes comprehensive instructions on how to code records, a data entry program that can store and retrieve up to 5,000 records for a client and a Visicalc template to produce profit and loss account, balance sheet and source and application of funds statement.

As the report section is generated by Visicalc, amendments to the trial balance have an immediate effect on the accounts. Comparative figures for the previous year are shown, and the format of any of the accounts can be altered.

The system costs £115. *Tel: 0534-77268.*

Installing WordStar in the Apple

WHEN WordStar, which I have used for some time now without any problems, first arrived it had already been set up for my machine. Recently however, in reviewing three 80 column cards I had to instal the program myself because each 80 column card was slightly different, each with its own protocols.

Although 20 or 30 installations later I can carry out an installation in under 10 minutes, at that time I was a rank novice, so the first step, I thought, was to read the manual.

I read it, was none the wiser, and read it again. I was still having trouble so I phoned a dealer — "No idea" was the response. I tried another — "Do you have to instal it then?", and a third — "Oh, it can't be done with that 80 column card."

No friend had installed a "non standard" terminal, and no one seemed to be able to help. So, all else failing and refusing to be beaten by a machine, I sat down at the keyboard and went through it step by step. It was then I discovered that WordStar really isn't that difficult to instal! I also found out that most dealer installations leave a lot out — what the manual refers to as fine tuning. The work proceeds in five steps:

Terminal

The terminal is the first, and can be the most difficult step. Fortunately most 80 column cards that aren't listed look, to WordStar, like one of those that are, and can be installed as that terminal. So if your terminal appears in the list or "looks" like one that does, fine, select that. If not, select "none of the above", read thoroughly the section on user patch areas, and patch your terminal manual.

If you still can't find the answers, ring the card manufacturers or importers, who are mostly very helpful. The other alternatives are a letter to MicroPro in the States or experiment by elimination. If the last is necessary, you must have a really weird terminal . . . and my sympathy.

Two of the 80 column cards I had to instal were non standard. One looked like a listed card, and the other was installed after consulting the terminal manual and the card's importer. Both worked perfectly.

Printer

A lot of people have trouble with the printer for some reason. To start with, there are only two types: dot matrix and daisy wheel printers. If your printer works from slot 1 with Apple DOS it must work with WordStar. With all printers the extra facilities available are patched at the end of the instal program.

First dot matrix printers. If you check in the printer manual your printer will either backspace or not. If not select "A any 'teletype-like' printer (ie almost any printer)". If it does backspace, select "C

'teletype-like' printer that can backspace". That's all there is to it.

If you have a daisywheel which is listed select that option. If yours isn't listed then, as with matrix printers, select either A or C. As already mentioned, the fancy bits come later.

Communications protocol

If you have selected a specific daisywheel printer, select "E EXT/ACK Protocol", and set the correct baud rate on the interface and printer (best at 1200). If not, then select "N NONE required (or handled outside of WordStar)".

Simple wasn't it?

Printer driver

Here again, the selection procedure is simple. The CP/M master BIOS is

By T.N. THOMPSON

defaulted so that the printer, the physical device TTY:, slow console, is assigned to the logical device LST:. The version running on the Apple also requires that the slow console be located in slot 1. The only option available for the Apple is, therefore, "L CP/M 'List' device".

If you have a parallel interface do not be deceived by the "Parallel Centronics" option. All interfaces on the Apple are accessed by the 6502 processor and have their driver routines written in 6502 machine code. This parallel option is intended to allow the Z-80 direct access to the interface and would have disastrous results on the Apple as it could not read the 6502 driving routines.

Final tuning

Once all the above have been entered, you will be asked "Are the modifications to WordStar now complete?". Answering 'no' will take you to the patcher. It is here

you tell WordStar what extra facilities your system has, and the patcher's operation is very simple.

It asks "Location to be changed:". You can enter a hexadecimal address or one of the labels named in the manual. I suggest you stick to labels unless you are proficient at Z-80 machine code.

With labels, you simply enter the label name, follow it with a colon, and press the return key. The patcher then tells you the hexadecimal address associated with that label, its current contents, and prompts for the new hexadecimal contents. Pressing return retains the current content. Pressing return again instead of entering a new label increments the address by one.

If before launching into the labels you write down all the facilities you want your system to use, then write down their codes to perform the functions in hexadecimal. You will find the patcher takes no time at all.

One thing to note while patching labels is that in multiple byte labels it expects the first entry to be the number of bytes in the code. As an example, if your terminal needs ESC (to set inverse, then because ESC (is two characters long, Label should contain 02, Label+1 should contain ESC, and Label+2 should contain (.

There are a number of labels detailed in the manual, both in the section Modifications, and in the appendix "Listings of user modifiable areas", and I will detail a few which I have found most useful and might make your WordStar even more user-friendly.

ITHELP: After a short time with Word-Star you should find that you don't need the help menu on screen all the time. If you patch this label to a lower help level WordStar comes in at that level. No length byte is needed with this label.

ITPOPN: Patch this label non zero if you don't need page numbers on printouts most of the time. Page numbering will then be available by means of the '.pn' Dot command.

NONDOC: If you write a lot of non document text files patching this label to FF will bring WordStar up in non document mode.

IVON: If your terminal displays inverse characters in CP/M enter the length of the code, then the code, in the successive locations at this label. While a lot of 80 column cards display inverse in Apple DOS, only a few manage it in CP/M, so don't be too disappointed if it doesn't work. If yours does this is a must, as it makes text editing very much easier.

IVOFF: The reverse of IVON:, and has the same format. Turns the display back to normal.

TRMINI: This label should contain the length and then the codes, in successive bytes, that set your terminal correctly. It can be used to enable lower case, shift mod, etc.

TRMUNI: If patched, does the reverse of TRMINI: at the end of an editing session.

DELCUS: Sets certain delays for slow terminals. Apple 80 column cards are not slow terminals, so this label can be set to zero.

DELMIS: Same as DELCUS:

DEL1: DEL4: These delays can be adjusted to give the required delays for the various functions.

DEFDSK: This label sets the drive that WordStar looks on for its overlay files (that is all the files that have filetype .OVR). This is an important label because it can save a lot of work getting WordStar running. As a demonstration, by patching this label to 02 WordStar looks on drive B:

for the overlays. This means the only file I need on Drive A: is the small WS.COM file. This leaves the remainder of the disc free for text file storage and logs the drive correctly. I put the overlays in B:.

POSMTH: Check that this label contains the correct value for your type of printer.

BLDSTR: Sets the number of overstrikes for boldface printing.

DBLSTR: Same as above, but for double strike.

PALT: This label contains first the length of string, and the hexadecimal codes to set your printer to the alternate character font of your choice, assuming your printer has an alternate.

PSTD: Reverses PALT:

ROLUP: Contains the length and code for moving the paper for superscripts.

ROLDOW: Reverses the above.

There are many more labels than those listed. Studying the two sections that cover the labels may show others that you need to improve your WordStar. It really is easy to tailor it to your individual needs as well as the needs of your system.

Appletip

 If you want to print more than 40 characters on a parallel printer in AppleWriter 2.0 this is how to go about it:

Pull the print parameters menu up on the screen, and using the Top Line command, type in TL/CTRL-I80N/

Alternatively, using CTRL-V and imbedding the commands in the text also works.

Can't someone make plain the potently incomprehensible?
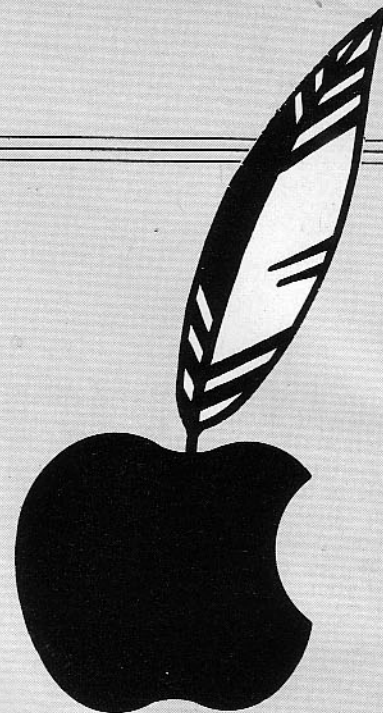
I READ with interest the comments and criticisms made by Derek Mills in the November Feedback about the lack of easily understood information.

As a beginner I have been using an Apple II for about four weeks and although I can now write relatively simple programs I know only too well the frustrations encountered by the poorly written guide books.

The Applesoft Tutorial is a gentle introduction, but in no way comprehensive. The Basic Programming Reference Manual is not a gradual progression from the Tutorial, but jumps so that only those with experience can understand.

I could give you numerous examples of instructions that having deciphered I have re-written in simple English so that I and my colleagues can understand and execute.

It appears from your reply to D. Mills that like nearly all experts in this field you have failed to recognise the problem and cannot therefore consider a solution. (Let a beginner review your Beginner's Dictionary.) After reading there must be some understanding before we can be "doing".

I hope that the book being reviewed "Kids and The Apple" in Windfall will be reviewed by a kid or a beginner so that it can be evaluated by those it is intended for.

As you know, the micro computer industry is growing very rapidly and we beginners will soon outnumber the experts (if we don't already). Perhaps you could devote a couple of pages of your magazine to raw recruits so that more can benefit from the potential at our finger tips. — **L. Gazzard, North Harrow, Middlesex.**
P.S. I didn't understand the Peek and Poke introduction either.

● I did not write the reply to Derek Mills' letter, writes **Max Parrott,** nor do I believe myself to be an expert, but I do agree with the given reply. In fact as far as programming the Apple goes I do not believe there are any experts.

Let me exemplify what I mean. Basic is a programming *language;* note the stress on language, for that is what it is. As time goes by you become more fluent in the language. Using the Applesoft Tutorial will accomplish this. The reference manual is a dictionary. If you have *worked* (not *read*) your way through the tutorial you must be reasonably fluent in the language.

However, that does not mean you are going to sit down and write a brilliant program. Why? Because you have to be imaginative and create a new idea, or indulge in a little plagiarism and rework someone else's ideas.

You then have to work at it, honing it, polishing it, and maybe rewriting the text completely.

In other words, it's like sitting down with a pen and writing an essay in whatever language you are fluent. At the end you've not only produced a fine piece of work but your fluency in the language has increased although you will always find somebody who is more fluent than yourself.

Mr Gazzard is already becoming more fluent because he has translated numerous instructions into simple English. He must realise that to become more so he has to read and re-read about different aspects of the language.

This brings me to a final point. The one aspect of the Basic language which is most frightening to all who are not used to it is the PEEK, and even more so, the POKE. You can do absolutely no harm to your computer, your program or yourself by PEEKing various locations. I urge Mr Gazzard to do so, to re-read the PEEK and POKE introduction and to generally have a hard night's work just puzzling out what's going on.

Read as many books and magazine articles as you can and continue playing with the machine. Read about a few POKEs and try them out. They may well destroy a program in memory or maybe the disc operating system, but you cannot hurt the computer and you won't hurt a disc (especially if you leave the drive door open for safety.)

If it all fouls up, just switch off, wait a minute or so and switch on again. In another four weeks' time it would be useful if Mr Gazzard could compose an article for *Windfall* on the subject of the problems encountered by himself and friends, pointing out the solutions found to each problem. This could benefit the other "raw recruits" but mostly he won't believe what progress he's made. Another idea is to join a local user group and gain some fluency via conversing.

Welcome Windfall

I HAPPENED to have seen the letter in Windfall, March 1982 concerning the "&LOMEM:" utility by Neil Konzen (The Apple Orchard 1980) and Max Parrott's routine for splitting Applesoft programs into two parts just now, reviewing some back copies I'd ordered. Though more than half a year has already passed, I would be very glad if you could still send me a copy of them both.

I would also like to congratulate your team for producing such an excellent computer magazine, beginning just with the always very pleasing front cover!

It's a pity that in Switzerland we are looking too much to the United States and to publications coming from that country, so your magazine is hardly known, though coming from a much nearer region. For me it was really a windfall seeing accidentally one of your advertisements! — **M. Häni, Branzistr.6, CH-8303 Bassersdorf, Switzerland.**

Earth defence

I HAVE experienced some problems in running the arcade game "Earth Defence" (Windfall, July 1982). They occur in the actual game itself. The introduction etc, and ending are perfect.

It is the actual firing and scoring location/sequence that is not appearing and running. And instead the end result is given ie: 0 Androids have been shot down in 0 minutes/time etc.

I wonder if there can possibly be a fault in the printing of the program. If none are known or if you have not been notified of any mistake made by the programmer I

would appreciate any tips to rectify my problem. – **C. Geraghty, Durban, South Africa.**

● As far as we know no one has experienced any problem with the actual running of Earth Defence, so presumably (although it is difficult to say without seeing your listing) you have incorporated a typing error while putting it to memory.

The program has an ONERR GOTO 990 (line 120) which is used to print out the final score, and obviously you are ending up here prematurely.

The remedy is to temporarily make line 120 a REM statement and rerun the program. The Basic should now tell you where the errors are. When you have found them all, restore line 120 and play it again.

If you still have problems remove the GR command in line 1650 and line 2020 and try running the program with TRACE on. See if the correct line sequence is being obeyed. **Max Parrott**

Extra bass in Tempo

AFTER printing out the musical program Tempo, being very tempted to play music on my Apple, I discovered, after many moments of enjoyment, that the Catalog selection/menu Make Music Subroutine was slightly faulty.

After the program had successfully converted my personal musical program to Applesoft Basic, the problem soon came to light when I ran the Basic program.

Firstly it had a completely different "deep" sound to it, and secondly it soon ended with the "RETURN WITHOUT GOSUB error" message in line 14050 where, incidently, the conversion program takes place.

I would gladly accept any advice or, if there was a misprint or any other mistake, I will soon pick it up in the following edition of Windfall. – **Charles Geraghty, Glenwood, Durban, South Africa.**

● I don't know where the "deep" sound came from on using the Make Music subroutine option. I have used this many times and it hasn't happened to me yet. (See program SCRAM in the December issue of Windfall.)

As for the RETURN WITHOUT GOSUB this is because whenever subroutines are used at the top of a program END must be included before reaching the subroutines. Otherwise when your program finishes it runs straight into the subroutine without being told to, hence in this case when reaching the RETURN in line 14050 it is a RETURN WITHOUT GOSUB. You could cure the problem therefore by adding the line 13900 END. **Michael Hambly**

Bug in the TASC compiler

I HAVE found a serious bug in the Microsoft TASC compiler for the Apple II. Alas it only seems to occur with large programs and is very unpredictable.

If you fail to declare a string before it is read in from a disc file using an INPUT statement, you might end up with a load of rubbish. Also if you assign a value to a string using its Ascii code this may get changed to something else half way through running your program, for example: DS=CHR$(4) might get changed to CHR$(94) or even "SL JRLWL"!

I have contacted Microsoft who say that an update is now available from their UK office. The new version is 2.01 (the old one being 2.0). Their address is Microsoft, Bulbourne House, Gossoms End, Birkhampstead HP4 3LP.

They will send you a replacement if you send them your original master disc. – **David Bowles, London NW5.**

U-Term review

THERE were a number of inaccuracies and omissions in the recent review of our U-Term and I trust you will allow us to correct them.

Documentation was criticised for being scarce – although a similar review in Microdecision magazine said that our manual provided all the information really needed. The source listing of the U-Term Eprom is available on disc and I suggest that this was what your enthusiast reviewer missed. Of course printing it in every copy of the manual would have drastically increased the size of the manual but baffled most users.

Assuming that support by us was being criticised, I refuse any implication that we don't support our products. In fact as the only UK manufacturer reviewed I am quite sure that our factory-direct support is superior to that offered through the various here today/gone tomorrow importers of cards from the USA.

Regarding software, I believe that to be really useful your reviewer should have covered standard packages available for 80 column use.

U-Term really scores here being compatible with word processors such as Wordstar, Applewriter II (the only one of the three to be so), Systematics Financial Controller, Ormbeta database and packages, Visicalc and many more. An 80 column card with no software to use it with is an expensive luxury.

I believe that magazines would only gain by giving the manufacturers the chance to review the reviews before going to press. Keep up the good work. – **Dr W.D. Unsworth, director, U-Microcomputers, Warrington.**

● Our reviewer Terry Thompson actually praises the card bought from you, and as it was the first 80 column card he had experienced we would assume his assessment of it is valid. Here is his reply to your letter:

If the source listing is on the disc, there is no mention of this fact in the documentation, and so I feel my criticism to be fair. When I discovered the problem with Pascal I immediately contacted U-Microcomputers direct.

I spoke to a number of people and was, at first, informed by them that the problem didn't exist. I managed to persuade them to the contrary, only to be told that there was no solution with any make of 80 column card.

However the Vision-80 and Videoterm cards do offer a solution to the Pascal problem and U-Microcomputers have, I believe, still to rectify this problem five months on.

From Dr Unsworth's comments, it would seem that he has never installed Wordstar (see Page 50) as the U-Term cannot be installed without special knowledge. Secondly, Wordstar was designed to work best with inverse characters, which are not available in CP/M with the U-Term, but are available on the other terminals, although at extra cost on the Videx.

The other two cards also have an 80 column expansion for Visicalc, but I have not seen this feature on the U-Term.

Applewriter II and the other software are available working on the Vision-80 as is Magic Wand with most working with the Videx.
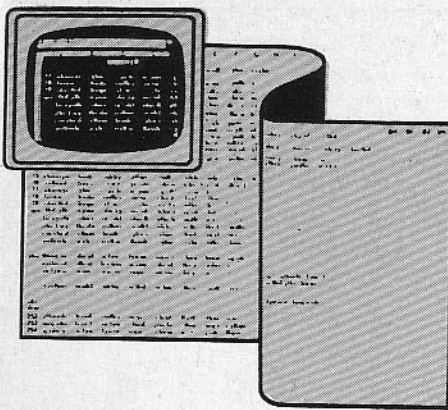
Keep that cash flowing

By NICK LEVY
Principal, Interface Management

HAVE you heard how Visicalc saved a company from going to the wall? When management realised that the company was heading for trouble, they decided, as a last resort, to bring Visicalc to the rescue. So every morning the MD of the company locked himself in his office with Visicalc leaving everyone else in the organisation free to get on with their jobs without him interfering.

Assuming however that you have more conventional applications for Visicalc in mind we shall look this month to see how Visicalc – or for that matter Supercalc – can help you to forecast your business cashflow. But first we have to start with a short preamble about cashflow planning. "Never mind the profit – watch the cashflow". This describes most appropriately what cashflow planning – the best accounting method ever devised – is all about. In other words, if you look after the cashflow (ensuring your company's liquidity at all times), then the profits will look after themselves.

Are you aware that profitability and li-

quidity stand in opposition to each other? That's why companies whose profits are on the increase are most vulnerable to bankruptcies – the more profit they make on paper, the greater the drain on their cash resources. Bear this in mind when your company comes out from the recession, and don't go for profit maximisation as this could force your company to go into liquidation in spite of its rising trend of record profits.

If a company cannot find the cash to pay the weekly wage bill, then no amount of paper profit can save it from going into immediate liquidation. A company which consistently fails to make profits can survive for a long time before crashing, but in the shorter term a profitable company which runs out of cash can collapse overnight. That's why you must watch your cashflow more consistently than your profits (a process which is even more subjective and arbitrary then the method of counting how many unemployed people there are in the country).

If as a precaution, you decide to keep an unduly large proportion of your assets in a liquid or near liquid state, then although you have safeguarded your company from sudden bankruptcy you have also made it vulnerable to takeover bids, because other companies will be after the under-utilised cash or quasi-cash resources held by your company.

However the holding of large cash reserves should not always suggest inadequate financial management. On the contrary, it is quite possible that such cash reserves are kept in readiness to make takeover bids. So watch out for companies with proportionally large cash reserves. Are they prey? Or are they on the prowl?

Considering that cashflow is the lifeblood of any business, why is cashflow accounting not more widely recognised and accepted as the prime accounting method? There are two possible answers.

First because, whereas companies must by law prepare profit and loss accounts and balance sheets, there is no

law requiring companies to prepare accounts based on cashflow.

Secondly because no organisation in the world would be prepared to let any outsider, let alone competitors and potential competitors, find out where they are going to get their money and how they are going to spend it.

But what about the cashflow statements that we see published in public companies' annual accounts? Shouldn't they be treated as classified information? These statements are not really cashflow statements in the sense that you have a projected weekly, or monthly, time series showing how much cash will come in, and how much will be paid out during each of the forthcoming weeks or months.

These published statements have a periodicity of one year, and only refer to the past – they don't project the future. That's why they are more appropriately described as "Sources and application of funds statements."

A company's cashflow actually consists of three separate streams of cashflows, and for any company's projected cashflow statement to be of practical value, the following three streams must be shown summarised separately (see Exhibit I):

1. Cashflow from operations.
2. Capital cashflow.
3. Financial cashflow.

Did you know that in 1978 the operational cashflow results of BL actually showed a £2 million surplus of receipts over payments? Unfortunately however, capital and interest payments have turned this surplus to a £291 million cash deficit. Our next example comes from the world of Concorde: How often have you heard the comment that Concorde does actually produce operating profits but is unfortunately crippled by the non-operational financial burdens? All of which goes to show how important it is to plan and monitor the non operational cashflows as well as the operational cashflow.

In spite of the above preamble, this month's cashflow model on Visicalc should not be regarded as a model statement. The object of the exercise is mainly to demonstate some Visicalc techniques and to provide you with some ideas which, if they appeal to you, could be incorporated in your own statements.

Exhibit II is the operational cash flow statement. Row 73, which is one showing the monthly profits, is not the same as

Exhibit I

row 83, which shows the monthly cash-flow arising from those profits.

Exhibit III shows to what an extent the cashflow is trailing behind the paper profits. Observe the shaded areas: Who would have thought from just looking at the figures that on the month when the profits are approaching their peak (July), the company has to pay out more money than the amount coming in.

What determines the cashflow and why does it lag behind profits? The answer depends on the parameters applicable to your company. So let's look at these parameters and also see to what an extent they can be changed or controlled by your company.

The cashflow parameters appear as a block in Exhibit II ranging from cell A36 to K58. According to the entry in cell D42, 35 per cent of the company monthly sales will be paid in cash during the same month on which the sale was made.

The cash for 50 per cent of what the company sells every month will be received one month in arrear (cell D46), and the cash for 12 per cent of any monthly sales will be received two months in arrear. In other words the cash inflow on any particular month will consist of 35 per cent of the value of that month's sale, 50 per cent of the previous month sales and 12 per cent of the sales made during the previous two months.

These figures assume that the recovery of the cash for 3 per cent of the monthly sales could take a long time. Generally speaking there is very little a company can do with regard to significantly improving on these cashflow parameters. On the other hand if the company does not watch these parameters they can easily deteriorate, to the detriment of the company.

The remaining cashflow parameters contain elements over which the company can exercise a certain amount of control. As you can see, 75 per cent of what the company buys (including the cost of labour) is paid for within the same month as the purchase was made (see cell J39), 20 per cent is paid for a month later (see cell J45), and the remaining 5 per cent two months later (see cell J50).

In other words, the company makes every month cash payments for 75 per cent of its purchases during the current month, 20 per cent of the purchases made on the previous month and 5 per cent of the purchases made two months ago.

Now let's see what would happen if we change some of these parameters. Assuming we cannot influence the rate at
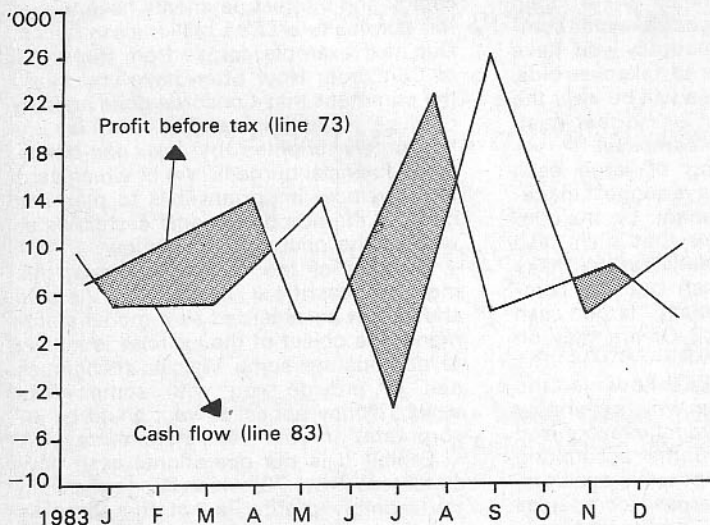
Exhibit II
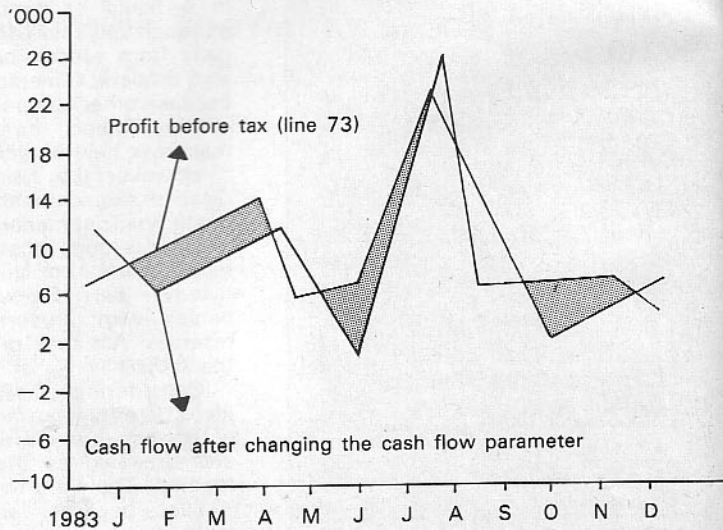
Exhibit III

Exhibit IV

which money comes in, but we can control the rate of cash outflow, let's change the entries in cells J39, J45 and J50 to 40, 35 and 25 respectively. The results will be reflected in Exhibit IV, which shows narrower gaps between profits and cashflow compared to Exhibit III.

Remember that by themselves the cash flow calculations (row 83) as well as Exhibit III and IV could be misleading, because they do not include any cash flows of a capital nature (arising for example from changes in the level of stock holding), nor do they show the financial cash flows (e.g. corporation tax — if at all paid by the company — or the receipt or payments of interest and dividends).

Next month we shall propose a Visicalc template showing how the combined results of our three streams of cash flow forecast should be presented. In the meantime let's look at some of the formulae used in this month's operational cash flow template (Exhibit II).

Exhibit II is designed so that when you load it for the first time your screen will look like Exhibit V. This is in order to hide the forecast from any unauthorised person loading it and looking at the figures. If such a person knows his way around Visicalc, and provided that he is given the time to explore the unreadable spread sheet, he or she could find a way to restore the figures into all those cell showing 〉〉〉〉〉〉〉

How was that done? First we format the following cell into graphic mode using the command /F*: P48, D60, F53, F54, F55, F56 and F57. Next we enter the following figures and formulae in their respective cells:

In P48 enter .0123 or any other number less than 1 which you would like to use as a secret code number. Note that after you have entered your secret code number the cell appears blank.

In F53 enter 825945 or any figure which represents your total annual sales. Do not be surprised to see the results of your entry looking like this: *******. In F54 enter 70 or any figure which represents the weighted average percentage of the cost of goods sold by your total direct organisation (in other words, express your cost as a percentage of the annual sales).

In F55 enter @SUM(035 . . . 047), and in F56 @SUM(049 . . . 059). In F57 enter 45, or whatever is the appropriate effective corporation tax for your organisation. Note that all your figures will come out as a series of *******.

In view of the size of the model it would be a good idea to format it to cell width of seven characters. So enter /GC7. Next format cell J53 to interger (/FI) and enter the following formula: @IF(D60 = P48,F53,1000000000).

The formulae compares the entry in cell D60 with the entry in cell P48 — where you put your secret code number. If the entries in both cells are identical the formulae will copy the value hidden behind the ******* in cell F53. If they are not identical the formulae will write the number 1000000000. But the size of cell J53 is too small to contain 1000000000, so instead you get a display of

Exhibit V

〉〉〉〉〉〉〉〉. As soon as you enter your secret code number in D60 cell J53 will display the value hidden in F53.

Next replicate that formula from J54 to J57 (all relative), and you can start to see how your secret code comes into operation. Always make sure that cell D60 is either blank or contains the wrong code number whenever you save the model or leave it unattended.

With the exception of the above formulae, the part of the model from row 22 to row 65 can be copied as shown in Exhibit II. Following are some of the key formulae used between row 66 and row

84. They should help you to complete the model. If you find that you cannot get on with that exercise, write to Windfall and ask for a complete listing of the January 1983 operational cash flow model, enclosing a stamped addressed envelope.

Finally note that there is a cost item in Exhibit II which appears twice, and which should never be included in a cash flow statement. It was left there because its inclusion would not have a significant effect on the cash flow. Can you spot that cost item? If not it will all be revealed next month.

To be continued

```
D66:/FI+J5

E66:/FI+D66*E63/100. Replicate E66 from F66 to P66, I shall leave it to you to

determine when to enter (R)elative and when (N)n change.

D67:/FI+J54/100*J53

E67:/FI+J54/100*E66. Replicate E67 from F67 to P67.

Row 69should be self explantory.

D70:/FI+J55

E70:/FI+D70/@COUNT(E63...P63). Replicate E70 from F70 to P70.

D71:/FI+J56

E71:/FI+D71/@COUNT(E63...P63). Replicate E71 from F71 to P71.

D73:/FI(D69-D70-D71). Replicate from E73 to P73.

D75:/FI+J57/100*D73. replicate from E75 to P75.

Row 77 should be self explanatory.

E80:/FI+C27*D50/100+(D27*D46/100)+(E66*D42/100)

F80:/FI+D25*D50/100+(E66*D46/100)+(F66*D42/100)

G80:/FI+E66*D50/100/(F66*D46/100)+(G66*D42/100). Replicate G80 from H80 to P80.

E81:/FI+C28+C29+C30*J50/100+(D28+D29+D30*J45/100)+(E67+E70+E71*J39/100)

F81:/FI+D28+D29+D30*J50/100+(E67+E70+E71*J45/100)+(F67+F70+F71*J39/100)

G81:/FI+E67+E70+E71*J50/100+(F67+F70+F71*J45/100)+(G67+G70+G71*J39/100).

Replicate G81 from H81 to P81.
```
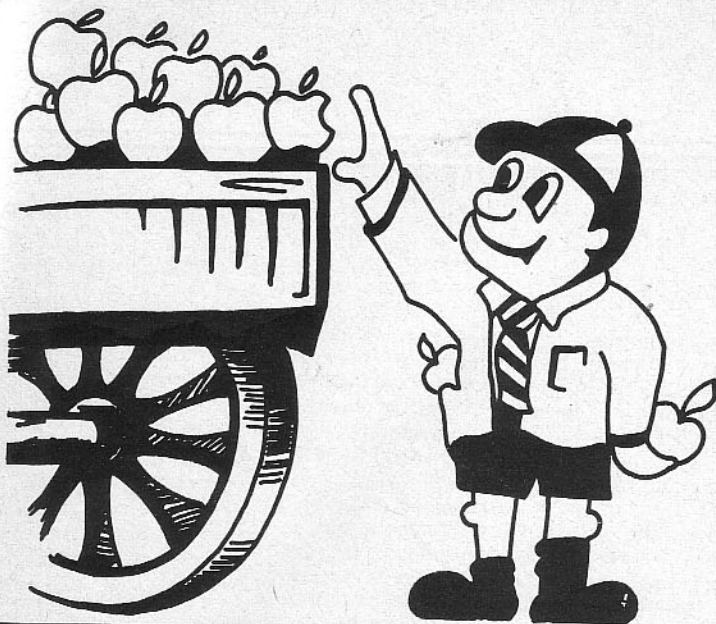
Pilot — a language for high flying teachers?

By BRIAN S.
RUSHTON

**Department of
Biology,
New University
of Ulster**

WITH the appearance of microcomputers in both schools and colleges, CAI (Computer Aided Instruction) is now becoming a distinct practical possibility. You know the sort of thing — sit Johnny down in front of a microcomputer and monitor, he switches it on, loads a tape or disc and is then presented with information which he works through at his own pace. The machine might ask Johnny questions and if successfully answered would lead him on to other delights; if wrongly answered then maybe a gentle reprimand, back to the basic information and ask him the question again. And so on.

If you have contemplated this approach and if you have tried to produce your own CAI lessons using one of the standard microcomputer languages such as Basic or Pascal you might well have come across several problems.

Using a standard language it is easy to put information on the screen. For example, a Basic program would use a series of PRINT statements so that as the program is run the information in the PRINT lines would be output to the screen, e.g.

```
10 PRINT "THIS IS A SAMPLE CAI
   PROGRAM"
20 PRINT "TODAY, WE ARE GOING TO
   LEARN"
30 PRINT "SOMETHING ABOUT THE PILOT
   LANGUAGE"
   etc.
```

All the information contained in quote marks would be output to the screen when the student runs the program. Simple so far, but from then on it gets much more difficult. If, for example, you want the student to interact with the machine — to give the machine answers to questions which if right causes more of the program to be executed or if

wrong for part of the program to be repeated — then Basic cannot cope easily with this. At best Basic would be cumbersome; at worst it might be impossible.

Difficulties would certainly arise when in answer to the question "How many legs has a donkey" a student answers 4 while another student answers Four. Basic cannot easily handle graphics displays, and many microcomputers do not allow for graphic and extensive textual information to be combined on the same screen display — unfortunate since much science is presented in this form. And there is nothing more boring than screen after screen of text with no visual break.

To the aid of teachers comes Pilot (Programmed Inquiry, Learning or Teaching), a computer language developed specifically for the production of computer-assisted lessons. It was developed in the late '60s and the original language had only eight basic instructions. However, it has been considerably revised since the original and is now a language with a wide range of different instructions, but remains in essence a simple language to learn and use. You can write relatively sophisticated lessons using only five commands: Text, Accept, Match, Jump and End — but more of these later.

Pilot is available for several microcomputers — the one we have been testing at NUU and the one on which the following is based is for Apple II machines — Apple Pilot. The package consists of two manuals, an editor's manual and a language reference manual, together with an author disc and a lesson disc.

The former is used by the teacher to prepare and test lessons using the Pilot language on the disc.

The lesson disc stores the created lesson and is the one ultimately used by the student. To write lessons using the author and lesson discs requires two disc drives; for the student to run the lesson, only one disc drive is necessary. The lesson disc is copyable. This provides a means of disseminating lessons to other users without the need for every user to have the Pilot package — therefore if you have produced a good, workable, interesting lesson you could mass produce it, sell it and help swell the school funds.

So how do you actually write lessons? The author disc has available four editors. The lesson text editor is the one that actually creates the lesson; the other three editors produce special facilities that the lesson text editor uses. The three are the graphics editor, character set editor and sound effects editor. Using the lesson text editor the author prepares his lesson. Generally, this takes the following form:

Give the student some information
↓
Test his/her comprehension
↓
If student performs satisfactorily in the test then go on to give him/her more information. If the student does not perform satisfactorily then either repeat the information or give him/her the information in another form or give him/her an electric shock to improve his/her concentration (!)

There are three basic types of instruction which carry out this lesson program:

☐ Text instruction
☐ Response instructions
☐ Control instructions

The text instruction allows information to be displayed on the screen when the lesson is run. They are preceded in the program by T:. There are two important response instructions — ACCEPT, abbreviated to A: in the program and MATCH, abbreviated to M:. When the lesson is run the computer waits at an ACCEPT instruction until the student has typed in an answer. A MATCH instruction in a lesson causes the computer to match whether the answer given by the student is an acceptable answer — it matches the answer given to the correct one. There are two important control instructions. END, abbreviated to E: is used to signify the end of a lesson, and JUMP, J:.

The latter causes the computer to JUMP through the program to a particular point, usually depending on the outcome of the student's answer. Thus in our example above, if the student's answer were correct then the computer needs to JUMP to new information in the program; conversely, if incorrect, then the computer needs to JUMP back to repeat old information.

The JUMPS are made to pieces of program identified by LABELS. A LABEL in the program consists of an asterisk followed by a name for that bit of program, e.g. * CORE. The JUMP instruction is a means of creating branches in the program and is, of course, the equivalent of GOTOs or GOSUBs in Basic.

If all this is somewhat baffling, perhaps a sample trivial program might help. But before that, could I add one complicating factor — many of the simple instructions can be altered by conditioners. The following program has a conditioner with the JUMP instruction, a Y or N. These are set by the MATCH instruction. JN: causes a JUMP if the MATCH was incorrect (i.e. the student got the wrong answer), JY: causes a JUMP if the MATCH was correct (i.e. the student got it right!).

In the former case, if the student had got the answer right, then no JUMP would have been executed and the program would continue to the next statement. In the latter case, if the student had got the answer wrong, then no JUMP would have been executed and again the program would continue to the next statement.

A sample program lesson is shown below.

The Pilot lesson program	Comments
R: A lesson about flowers	*This represents a comment or remark statement which is ignored by the computer and is inserted to remind the author of program structure.*
T: Hello! Today we are going to learn **T:** something about flower structure. **W:**	*Text displayed on the screen when the program is run.* *A pause of 10 seconds while the student reads the introduction.*
*FLOWER	*This provides a label for the part of the program which follows. It is not output to the screen.*
G: ES	*Clears the screen.*
T: A flower is usually composed of four **T:** distinct parts: the calyx, corolla, **T:** androecium and gynoecium. The **T:** calyx is usually a protective covering **T:** which surrounds the more delicate **T:** inner structures. The corolla com- **T:** prises the petals, the androecium **T:** the stamens, and the gynoecium the **T:** carpels. **T:** From your knowledge of the Greek **T:** origin of words, which part of the **T:** flower do you think is the male?	*Text displayed on the screen ending with a question which the student is expected to answer.*
A:	*Accepts the answer from the student.*
M: Androecium **JN:** FLOWER	*Did the student type Androecium? If the student did not type Androecium then the program jumps to the section labelled FLOWER and the text about flower structure is repeated ending with the same question. Otherwise (i.e. if he did type the correct answer) continue on to the next bit of information, labelled COROL. Note that this simple program example demands an upper case A followed by lower case letters in the answer. Apple Pilot provides control options to overcome this kind of problem.*
*COROL **G:** ES	*Another label.* *Clears the screen.*

T: The number of petals and their
T: arrangement within the corolla are
T: very variable. For example, a
T: Buttercup flower usually has 5
T: petals all of similar size and shape.
T: Petals of the Whin flower are also 5
T: in number but are of different size
T: and shape. Some flowers have more
T: petals, e.g. Purple loosestrife while
T: some have only two, e.g.
T: Enchanters nightshade.
T: From what you know of the petals
of Buttercups would you expect the
flower to be relatively symmetrical
in shape?

A:
M: Yes
JY: ANDRO

*More text again ending with a
question.*

Accepts the answer
Did the student type 'Yes'?
*If the student gets the question
right, then the program jumps to
the section labelled ANDRO. If
wrong, then in this case there is
no jump instruction and therefore
instead of repeating the informa-
tion a reprimand is given in the
next text statement and the
lesson also passes on to the
ANDRO section.*

T: No!!! Think about this. If all the
T: petals were all the same size and
T: shape then they would probably be
T: symmetrically arranged.

ANDRO

etc.

E:

*Text only displayed if the pre-
vious question was incorrectly
answered.*

Another label

The end instruction.

The lesson text editor allows complex branching and question/answer sessions to be built up from relatively simple instructions like these. One useful feature is that multiple answers are available in MATCH statements. Thus:

M: Calyx ! calyx would 'capture' the student who capitalised the first letter and the one who did not. Similarly

M: YES! YEP! YEAH! OUI! Y would allow for five different but equally correct answers. The ! is the equivalent of "OR". The MATCH instruction can become more complex — you can have multiple answers using the & sign and a 'wild card' or 'joker card' to trap spelling mistakes, such as:

M: B*C*CLE would trap mis-spellings of bicycle (or is it bycicle?).

Of course the lesson text editor only allows for text. However, Apple Pilot also has special instructions which allows the lesson to call files created by the three other editors. Suppose you wanted to display a Whin flower in the above lesson. Using the graphics editor you could easily construct a labelled diagram on the screen and save this in a file called WHIN. The program could then contain two extra ins-

tructions:

G:ES This clears or erases the screen

GX:WHIN Causes the program to call the file WHIN and draw the flower on the screen. Suppose you wanted to precede this drawing with a merry little jingle — Ring around the Roses? Then using the sound effects editor a file can be created containing the notes of the tune which when executed in the program will play it back. You don't need to be a Bach or Lennon to do this either. You can reduce sheet music to a series of letters and numbers! Our program would now contain one extra instruction:

SX:ROSES Plays Ring around the Roses composed by the SOUND EFFECTS EDITOR and stored in a file called ROSES

G:ES
GX:WHIN

The remaining editor, the character set editor, is perhaps potentially much more useful. It allows a user to define his own set of characters which can be called from within a program e.g. $^2, \Sigma, \neq$, various chemical formulae — OH groups etc., or even small pictures that may be required over and over again. These can be included in the character set and used whenever required in the program.

There are many other facilities in the language which cannot be dealt with in a short review like this. However, one further feature should be mentioned — it concerns file handling from within a running program. One expected feature of CAI is that the teacher should be able to monitor a student's progress. A way of accomplishing this would be to record the student's name and the responses he makes to questions on a file on the lesson disc as he uses the lesson. This 'hidden' information can then be accessed by the teacher at a later date and the student's work evaluated. Apple Pilot easily performs this.

The language is also able to deal with conventional numeric and string variables — interestingly these are relegated to the "Advanced Programming Techniques" section of the manual.

So that's it. What does it cost? — current prices range from £75 to £85 but you must have an Apple II 48k with two disc drives and DOS 3.3. There is also an animations package available (about £40) for use with Apple Pilot (animated Buttercup flowers?) but as yet I have no experience of it and cannot comment.

So what are the disadvantages? There are several associated with CAI *per se* — time commitment to preparing lessons, updating, lack of machines — but these have been aired elsewhere. So far, there appear to be only two problems with Apple Pilot itself — editing is cumbersome compared with Basic, but is probably largely a question of practice. Disc wear might also be a medium term problem since the author disc and in particular the lesson disc are fairly heavily worked.

This series by SEAN OVEREND takes the lid off assembly language and machine code programming, by describing the purpose and structure of a sophisticated assembler written in Basic.

So this is what it's all about..

AN assembler allows the programmer to use readily understandable instructions as source code, which is then "assembled" into object code. Assembly language consists of lines containing symbolic labels, mnemonics for opcodes, and operands written in a conventional way (which specifies the addressing mode), allowing for the use of the symbolic labels, as well as purely numeric addresses.

The level of abstraction afforded by an assembler can be illustrated by analogy with instructions to a painter. Each of the painter's possible jobs has been placed in a list, and each has been given a number. Figure 1 shows part of such a list. (Thus the order to paint a front door appears as number 13 on the list). Figure 2 shows the house numbers of part of the estate.

There are three ways of telling the painter to paint the front door of house number 5:

13 5;

PFD 5 (using the mnemonic PFD instead of the list number 13); and PFD Whiteacre (specifying the name of the house instead of its number).

As a further variation, an instruction to paint the hall of number 4 could be given as PH Whiteacre -1 or PH Applegarth+2.

Similar levels of abstraction can be used to describe 6502 operations and addresses in assembly language.

Each line of assembly language program contains the same information as the machine code equivalent of one machine code opcode and its related operands (if any). The general format of an assembly language line is shown in Figure 3.

Features of assembly language

Apart from the advantage of being able to use the mnemonic — and therefore memorable — opcodes (e.g. STA = STore Accumulator contents in memory, LDY =

NR	INSTRUCTION	MNEMONIC
"		
"		
12	Paint back door	PBD
13	Paint front door	PFD
14	Paint hall	PH
"		
"		
"		

Figure 1

1	2	3	4	5
APPLE GARTH				WHITE ACRE

Figure 2

LoaD the Y register) the main features of assembly language relate to the manner of expressing an address.

Numeric addresses

Memory addresses can be referred to numerically, and 10, $A and %1010 all mean the same thing in decimal, hexadecimal and binary. The precise method of distinguishing each numeric base varies according to the particular assembler. Generally it should be possible to dispense with leading zeros, the only assumption being that the number in question is right-justified.

The assembler should also calculate whether the binary numeric representation of the address (not the contents) can itself be stored in one byte of memory. If it can, then the address is within the "zero-page" of memory.

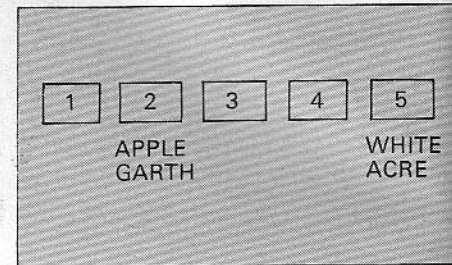Address number 255 (decimal) is the highest "zero-page" address. This becomes obvious when converted into hexadecimal ($FF), as any higher address requires more than eight binary bits to store it in.

Symbolic addresses (labels)

Addresses can also be represented as symbols by "labels", or "identifiers". The great advantage of this is that the programmer can assign a numeric value to a label, knowing that the assembler will substitute the correct numeric address whenever the label is encountered in the source code. There are two ways of assigning a numeric value to a label:

(a) By stipulating its value, directly. Many assemblers use the "pseudo-opcode" EQU for this purpose, e.g.:

Label	Opcode	Operand
SPEAK	EQU	49200

This line is an instruction to the assembler to associate the address 49200 (decimal, or hexadecimal $C030) with the label SPEAK when assembling the source code. The address is that of the Apple speaker switch). A subsequent line such as:

LDA SPEAK

means "load the accumulator with the contents of the address 49200". (See Reference Manual p.20 for the effect this will have.) The assembler translates this as:

Opcode	Operand
AD	30 C0

(Addresses which require two bytes to contain them are always presented to the 6502 in lo-byte, hi-byte order.)

EQU is a "pseudo-opcode" because it merely defines the address of the label associated with it, despite appearing in the opcode position (or "field") of the assembly language line. It does not specify a 6502 operation, as does a true assembly language opcode.

(b) By assigning a value to the label in-

Assembly language format

LINE NR.	LABEL	OPCODE	OPERAND	COMMENT

Line number: Used solely to facilitate input/editing of source code. Interactive assemblers allow commands such as LIST, PRINT, ASSEMBLE, END

Label: May be omitted. The characteristics of the assembler will dictate maximum length and whether alpha/numeric characters allowed

Opcode: One of the 50 assembly language opcodes that specify the type of 6502 operation alternatively, a pseudo opcode such as EQU, DW, DFB, ASC or =

Operand: Addressing modes are implicit from the format:-

absolute :	no preceding #
	no ( )
	no ,
indexed :	contains ,
indirect :	contains ( )
immediate :	preceding #

Comment: Cf the Comments after REM in BASIC. Some assemblers insist on a preceding ;

Figure 3

directly, as a matter of calculation. In the following example:-

Label	Opcode	Operand
START	LDX	#$FF

the label START is associated not with a pseudo-opcode EQU, but with the opcode LDX. The assembler translates LDX #$FF into its machine code equivalent A2 FF, and also works out what address to associate with START.

If the machine code operand A2 was located in memory address 769 (decimal, or $0301), any subsequent reference to START in the source code would result in its conversion to the hexadecimal equivalent of 769. If the next line of assembly language was:

Opcode	Operand
JMP	START

this would become

4C   01 03

(again remembering the inversion of two byte machine code numeric addresses). Figure 4 shows the two lines side by side, in assembly language and in machine code.

Arithmetic feature

Most assemblers allow simple arithmetic operations on labels to identify labels displaced from the named address by the amount of the offset. Replace the second assembly language line in figure 4 with:

JMP   START+1

and the assembler will translate it to:
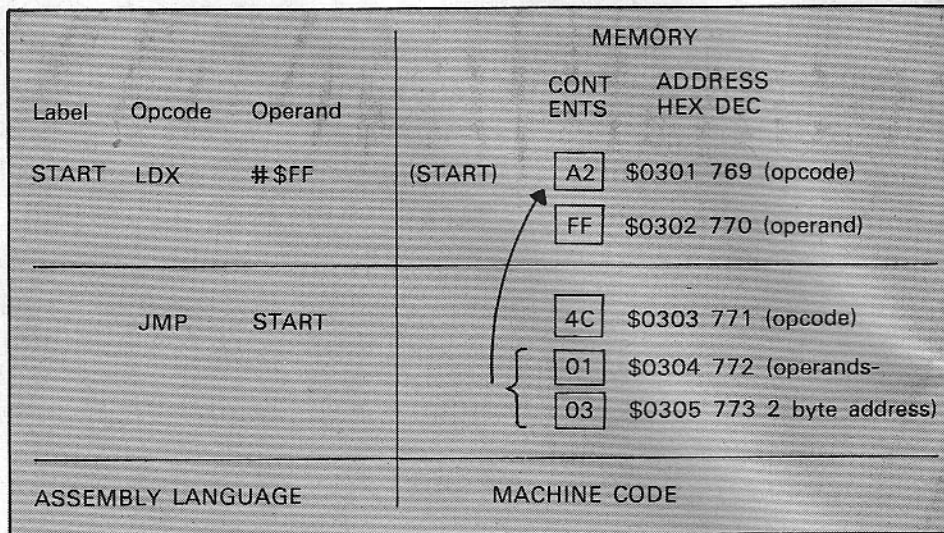
4C   02 03

because START is given the address

Figure 4

$0301, to which 1 is to be added, making $0302.

Summary

Assembly language can be defined as a series of lines of source code, each of which corresponds with one machine code opcode and its related operands (if any). A line of assembly language has five fields, two of which can be empty (the label and the comment fields), and at least one of which can be used for additional purposes, e.g. the pseudo-opcode.

Recommended reading: Zaks, Programming the 6502, pp 356-362.

The author's interactive two pass, disc-based editor-assembler is too long to publish in the series. Copies of the assembler may be obtained from the author in either Basic or compiled code (the latter being considerably faster in execution). Disc price £20, inclusive p&p (48k DOS 3.3 Applesoft firmware only).

Alternatively, send your own initialised disc and £15 handling charge. Specify whether Basic or compiled version required to Sean Overend, 22 Highland Road, Amersham, Bucks HP7 9AX.

If you have access to the Tasc Compiler, full instructions for compiling the Basic version are given.)

● **Next month's article covers the assembly process and its fundamental structure and looks at input editing and assembled output.**

Your Apple questions answered

● **HOW can I get text to appear on the graphics page?**
One way is to use the hi-res character generator from the DOS 3.3 toolkit or use shape tables (slower!)

● **WHY does TRACE not work with DOS commands?**
All DOS commands must be preceded by a RETURN, so to make TRACE work put a CHR$(13) into your DOS command (ie PRINT CHR$(13); CHR$(4);"OPEN FILE") OR D$ = CHR$(13) + CHR$(4) : PRINT D$ "OPEN FILE".

● **WHY do GET statements not work with DOS commands?**
This is similar to the last question. GET commands do not have a return on the end, therefore it must be added. Use the example from the last question.

● **HOW can I tab over 40 characters on my printer?**
You cannot use the TAB command. The command to use is POKE 36,X (where X = the number of spaces that you want to tab.)

● **WHY am I getting an "OUT OF MEMORY" error when there is plenty of memory space left?**
If you are getting OUT OF MEMORY errors that are not caused by running out of space the cause is a stack overflow. There are six things that cause this:
Too many FOR NEXT loops; too many GOSUBS; excessively complex mathematical or string functions; GOSUBS with no RETURN; improper recovery in ONERR GOTO routines; calls or interrupts that do not restore the stack.
These effects are cumulative and a stack overflow can be the result of two or three of them.

● **THERE is a program on the system master disc called 'INTBASIC'. Why can I not use it?**
This is a file that the HELLO program loads into a language card if there is one present in the system. If there is no language card present this file is not used.

● **WHAT 80 column card works with Applewriter 11?**
The Vision 80, the SUP-R-term and Videx (with boot disc).