C16 C116 PLUS/4

COMMODORE COMPUTING MONTHLY

THE ONLY COMMODORE C16 C116 & PLUS/4 NEWSLETTER THAT IS MONTHLY!

Software:

Odds
&
Sods

Hardware:

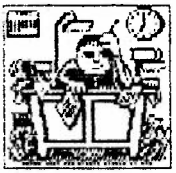
256 KB C16+4
C16 + 4
MOUSE ?

Hoky Poky!

Where's
The
Pokes?

CONTENTS

VOLUME 2 ISSUE 2 MAY 1990



ED'S NOTES

HOT STUFF FROM THE EDITOR'S DESK!

Hello once again to another issue,
Again I apologise for the delay, this was because I've been running about trying to get a computer show organised down in London for you lot. When I first phoned the nice chapo said I could have a £150 stand for nothing then a week later when I had got all the transport oraganised the nice chap rang back and said due to my club only having around 40 members I would have to pay ½ price £75, so I had to call the show off but hopefully next year I will get one, this is'nt aload of old bull, because Dave Brighton and Peter Crack and some other members where told, so sorry for the delay but I thought it would be worth it.

SUMMER DELAYS

Due to the school closing this week for 6 weeks summer holidays, you will not recieve the June/July & August issue until September 5th, sorry but I can't do it any quicker, anyway it means I can catch up as well. You will See in this ish, the Order Form for July issue, please add £1 for the August issue so instead of the £1 on the order form please send £2 for July and August ish's, 'cause Junes have been paid for.

COMPETITION

Remember the competition I set last issue, well I have not had one entry, what are you doing, just to make ait a bit more worthwhile, I'm now offering a full years Subscription free as the prize, yes 12 full months of not having to pay for a mag, now get coding, remember you can send more than one entry, but it must be sent on Disk/Tape and printed/handwritten, with descriptions of what the program does/. Comp. Address in last issue.

QUIXAVER

Quixaver is a unigue Fast Save for C16/+4 tape users, unlike other fast save systems, it does'nt use any of the RAM from $0800 - $FFF0, so you can fast save a full 64K program with no hassle, more details next issue, the program will be for sale to all members, and I think its a Priceless Utility for Tape Users, cheers Eric Jones for Coding it, its Brill!!!!!

The little black splodge in the left of the Ed's Notes last month was to be a photo of myself, but it turned gross when photocopying it, I'm working on it, because some members wanted to see what I looked like, (GOD KNOWS WHY!).

Till next issue, Bye.

Roy Robinson, 112 Cliff Road, HORNSEA, N.Humberside, HU18 1JE. Tel 0964-534611

Mr Roy Robinson,
112 Cliff Road,
Hornsea,
N. Humberside, HU18 1JE.

89 Old Road,
East Peckham
Tonbridge,
Kent, TN12 5EN.

29th May 1990

Dear Roy,

I would like to start a series in your magazine on the subject of graphic art. As an introduction, I would like club members to make the graphic of a microscope on the Graphic 3 screen and then subsequent articles will deal with on-screen modifications with my 'Graphic Editor' program, preparing the graphic for the printer by making the existing graphic into separate pictures which are the Color1, Color2 and Color3 constituents, and finally sending the data to the printer and making a multi-colour,triple overlaid image on the paper having made a small alteration to the Star LC-10 Colour printer which I see some members have like myself.

I am working with Peter Crack to speed up some of the parts of my programs, but we should be able to have the programs ready for the magazine since it will be a series.

I hope this idea meets with your approval and have enclosed the 'Microscope' program for you to put in the next edition of the magazine.

Yours sincerely
Rob
Rob Marshall

PROG OVER

```
10 REM ***********************************
20 REM *           MICROSCOPE            *
30 REM *         BY R.MARSHALL           *
40 REM ***********************************
50 COLOR0,7,4
60 COLOR1,8,7
70 COLOR2,14,7
80 COLOR3,3,0
90 GRAPHIC3,1:COLOR4,7,4
100 KEY1,"(WHT)GRAPHIC0"+CHR$(13)
110 CIRCLE3,105,57,8:CIRCLE3,105,57,2:PAINT3,110,57
120 DRAW1,72,85TO105,25TO113,40TO111,44
130 DRAW1,98,66TO80,101TO72,85
140 DRAW2,107,26TO110,19TO116,30TO112,37:PAINT1,92,72,1
150 DRAW2,72,85TO64,110TO67,116TO80,101:DRAW3,74,83TO81,
98:DRAW3,72,85TO80,101
160 PAINT2,108,27,1:PAINT2,75,95,1:PAINT2,105,57,1
170 CIRCLE2,105,57,4:DRAW3,104,27TO111,42:DRAW3,106,26TO
112,38
180 CIRCLE3,85,105,25,40,28,195
190 CIRCLE3,81,100,40,55,53,188
200 DRAW3,75,154TO78,143:PAINT2,76,153,1
210 DRAW2,98,67TO90,84TO91,86TO99,69TO98,67
220 DRAW1,45,100TO73,156TO75,154TO47,98TO45,100:PAINT1,4
8,103:DRAW2,49,96TO75,148
230 DRAW3,34,136TO60,146TO60,153TO34,143TO34,136:PAINT3,
36,142:DRAW2,45,148TO47,159
240 BOX1,30,160,115,168:PAINT1,31,161
250 CIRCLE1,51,95,1,,,,45:DRAW2,34,136TO60,146
260 CIRCLE1,85,148,5:PAINT1,85,148
270 BOX1,45,168,40,175:BOX1,105,168,100,175:PAINT3,43,16
9,1:PAINT3,104,169,1
280 CIRCLE3,85,148,5:CIRCLE3,85,148,2,1
290 CIRCLE3,85,148,3:DRAW3,84,159TO86,159
300 BOX3,32,162,113,166
310 FORI=0TO15
320 CIRCLE3,35+(I*5),164,1,1
330 NEXTI
```

```
*******************************************************************************
****************************** BLOOPING BUG ***********************************
***************** By Peter and Melanie Crack.   part five *********************
*
* This month I am going to start to fill in the missing subroutines which
* will create the sprites. First load in yor previous programme, enter A4CA0
* press 'space bar' and then type in the first command of the listing.
* Here is the explanation of what it does.....
*
4CA0-4CA9 Creates the sprite.
4CAC-4CB1 Load 'Y' register with sprite number and gosub 'print the sprite'.
4C00-4C07 Transfer 'X' register (this is the sprite no.) to 'A' register and
*         store it in correct position in sprite active list (all sprites
*         are dealt with in rotation sprite no.$$00 is first, sprite $$0F is
*         last, non active sprites in the list are marked as $$FF)and return
4C08-4C0C Clear $D0 and $D1 (set to $$00)
4C0E-4C10 Load 'Y' register with sprite number,load 'A' register with
*         correct page number where this sprite reload data is to be stored
*         ($49D0-$49DF contains these numbers one for each sprite),each
*         sprite has three pages of data alloted to it,the first is the
*         area of screen which is under the sprite and has to be put back
*         into the screen as the sprite moves on,but because in this prog.
*         more then one sprite will be on screen at the same time and more
*         importantly they will collide with each other this page of data
*         is used to remove the old sprite and only the old sprite from the
*         screen any pixels which remain set after this has been done must
*         be part of another sprite,thus sprite collision can be detected
*         unfortunatly this programme cannot tell if the set pixel is
*         background or sprite,so in this programme no background details
*         are used in the sprite area of the screen.The second page is the
*         sprite working area,this data is shifted left or right as the
*         sprite moves,the third data page is the sprite reload that is to
*         say it is the same as it is stored in $B000-$B7FF,this way you can
*         have more than sixteen sprite definitions as the others could be
*         stored in any of the spare areas from $C000 to $FEFF.
4C13-4C17 Store 'A' register in $D3 and $D5 and return from gosub.
4C28-4C29 Set interupt disable and tell computer to get all data above $8000
*         from RAM.
4C2C      Set 'Y' register to $$00
4C2E-4C30 Transfer data from area pointed to by $D0 and $D1 to area pointed
*         to by $D4 and $D5. In this instance $D1 was set to either $$B0 or
*         $$B7 at $4116 and $4124, $D0 and $D4 were set to zero at $4C0A and
*         $4C0C, $D5 was set to sprite area reload at $4C15, so here, we are
*         transfering the initial sprite definition from eithr $B000-$B0FF
*         or $B700-$B7FF to this sprites reload page.
4C32-4C35 Increase 'Y' register by one and check if it is zero if not then
*         branch to $4C2e and transfer another byte, if it is then return.
4C40-4C9A This routine is quite complex, I have set aside two areas of data
*         $4600-47FF and $4800-49FF, the latter is filled with fixed data
*         which I will explain in detail when I send it, the former contains
*         data taken from $4800-$49FF but changes as the game proceeds. O.K?
*         so here goes!!!!!!!!!.
4C40-4C42 Load $D5 with start of fixed data page address (hi-byte).
4C44-4C4D Load 'A' register with value stored in $D1(page number of original
*         sprite data area this will be in the range $$B0 to $$B7, as in
*         this game only eight sprite definitions are used more of course
*         are possible), remove the four leftmost bits and save a copy on
*         stack,shift all bits left four times (to explain $$B4 becomes $$04
*         and finally $$40) and store the result in $D4,$D4 and $D5 now form
*         the address where the sprite registers can be found.
*******************************************************************************
*********************************************.....CONTINUED.....
```

```
**************************************************************************
*
4C4F-4C55 Load 'A' register with sprite number(remember, the sprite number
*         in register $E5 and the sprite data page number can be different
*         in this way you can change the shape of the sprite but still print
*         it in the same order as before, to explain, if you have, as in
*         this game, a special sprite whose condition and position you wish
*         to test from time to time it is much easier to keep its sprite
*         number the same and simply load in different data to change the
*         shape), shift all bits left four times (the reason for all the
*         shifts is because there are sixteen pointers for every sprite
*         definition the first from $4800 to $480F the second from $4810 to
*         $481F and so on, so if we want sprite definition #$01 then the
*         start address is $4810, $D5 has already been set to #$48, $D4 had
*         to be set to #$10, the same applies here only this time the result
*         is transfered to the 'X' register.
4C56-4C58 Clear 'Y' register and load one byte from fixed data area.
4C5A-4C5E Check to see if 'Y' register is equal to #$0D if yes then discard
*         value in 'A' register and substitute value in $D1 (the original
*         sprite data page number in this case it will be in the range #$B0
*         to #$B7 this way we can change the shape of the sprite as we go).
4C60      And store it in $4600 offset by the 'X' register, $4600 to $47FF
*         is the data area that is changed as the game goes on.
4C63-4C64 Increase both 'X' and 'Y' registers by one.
4C65-4C67 Have we transfered sixteen bytes of data? no then branch to $4C58
*         yes then.....................................
4C69-4C6F Increase $D5 to point second page of fixed data area, pull $D1
*         back off the stack and shift it left three times, these shifts are
*         needed because there is yet more data pointers for each sprite in
*         $4900 to $497F eight bytes for each sprite, so again the data for
*         sprite number #$01 is found in $4908 to $490F, and #$01 shifted
*         left three times becomes #$08, now store 'A' register in $D4 thus
*         creating the new address.
4C71-4C77 Load 'A' register with pointer $E5 (sprite number) shift it left
*         three times and push a copy onto the stack then transfer it to the
*         'X' register.
4C78-4C83 Clear 'Y' register load first byte into 'A' register (offset by
*         'Y' register) and store it in $4700 offset by 'X' register,
*         increase both 'X' and 'Y' registers then check to see if 'Y'
*         register has reached #$08 (have eight bytes been transfered) if no
*         then branch to $4C7A if yes then ...................
4C85-4C9A Pull 'A' register off the stack (remember this was the value of
*         $E5 after three shifts left as at $4C76) and transfer it to 'X'
*         register, load 'A' register with $D3 (see explanation for $4C0E to
*         $4C17), store 'A' register in $4702 offset by 'X' register,do this
*         two more times reducing $D3 by one each time also changing the
*         position at which this data is stored by one each time and finaly
*         return from subroutine.
41EF-4261 Oh dear another biggie!!!!.This routine fetches all data pointers
*         for a particular sprite and distributes them to their proper
*         places around the programme.
41EF      Transfer 'Y' to 'A' registers this is done because some routines
*         have the sprite number in the 'Y' registers and gosub $41EF and
*         some in the 'A' register and gosub $41F0.
41F1-41FA Save 'A' register on stack, shift 'A' register left four times and
*         transfer 'A' to 'Y' registers, clear the carry flag ready for
*         addition, add #$0E (dec 14) to 'A' register and store it in $E4
*         ($E4 now contains the start point for our load routine plus 14).
41FC-4207 Clear 'X' register, load first byte from $4600 offset by 'Y' reg.
*         store it in $D4 offset by 'X' register increase both 'Y' and 'X'
*         registers and compare 'Y' register with $E4 (have 14 bytes been
*         loaded), if no then branch to $41FE if yes then.............
**************************************************************************
*************************************************.....CONTINUED.....
```

```
*
4209-4213 Store next byte in $430A increase 'Y' register and store next byte
*         in $4318.
4216-421A Pull 'A' register off stack (sprite number), shift it left three
*         times and transfer it to 'Y' register, thus creating the correct
*         offset for the next series of fetches.
421B-4261 Fetch seven bytes from sprite fixed data area and place them into
*         various routines around the programme (these routines will be
*         explained as I send them) and return from subroutine.
4340-4346 Disable interrupts tell computer to ignore ROM and take all data
*         from RAM, store 'Y' register in $E5 and then transfer 'Y' to 'A'
*         registers, the last two commands may not make sense but this
*         routine is used by different parts of the programme.
4347      Store 'A' register in sprite list ($47E0 to $47EF) this tells the
*         programme that a sprite is 'active'.
434A      Gosub get sprite pointers.
434D      Gosub transfer sprite reload data (sprite shape), to sprite
*         working data area.
4350      Gosub get random up or down movement delay (when two sprites
*         collide they move apart at slightly different angles).
4353      Gosub print sprite.
4356      Gosub store sprite data.
4359-435D Tell computer to take all data above $8000 from ROM, clear
*         interrupt disable and return from subroutine.
*         And that ends this months section of the programme, please do not
*         expect to understand this in one go,just read it through a few
*         times and it should help you to make sense of the listings.
*         as always any problems or suggestions write in to the mag or
*         phone me on 081-367-3152 if you cannot get me then leave your
*         name, phone number and problem and I will phone you back.
*************************************************************************
*********************************.....PETER CRACK.....********************
*************************************************************************
```

```
*
*
* Dear Ed.
* Now that all three parts of TRAP THE KING and the missing part have been
* printed could those members who have tried the programme write a review
* comparing it to other games they have tried and give me an idea how well
* or otherwise it plays, I am most interested in how clear the display
* and instructionswere, how good was my choice of colours bearing in mind
* that some colours look better on different types of televisions, and any
* other comments club members wish to make.
* those were just a few suggestions but of course any views or points are
* welcome.
*
*
*                                   PETER CRACK.
```

```
. 41EF  98       TYA
. 41F0  EA       NOP
. 41F1  48       PHA
. 41F2  0A       ASL
. 41F3  0A       ASL
. 41F4  0A       ASL
. 41F5  0A       ASL
. 41F6  A8       TAY
. 41F7  18       CLC
. 41F8  69 0E    ADC #$0E
. 41FA  85 E4    STA $E4
. 41FC  A2 00    LDX #$00
. 41FE  B9 00 46 LDA $4600,Y
. 4201  95 D4    STA $D4,X
. 4203  E8       INX
. 4204  C8       INY
. 4205  C4 E4    CPY $E4
. 4207  D0 F5    BNE $41FE
. 4209  B9 00 46 LDA $4600,Y
. 420C  8D 0A 43 STA $430A
. 420F  C8       INY
. 4210  B9 00 46 LDA $4600,Y
. 4213  8D 18 43 STA $4318
. 4216  68       PLA
. 4217  0A       ASL
. 4218  0A       ASL
. 4219  0A       ASL
. 421A  A8       TAY
. 421B  B9 00 47 LDA $4700,Y
. 421E  8D 19 45 STA $4519
. 4221  8D 1C 44 STA $441C
. 4224  8D 21 44 STA $4421
. 4227  8D 6A 44 STA $446A
. 422A  8D 6F 44 STA $446F
. 422D  B9 01 47 LDA $4701,Y
. 4230  8D 48 45 STA $4548
. 4233  8D 50 45 STA $4550
. 4236  8D 46 44 STA $4446
. 4239  8D 92 44 STA $4492
. 423C  B9 02 47 LDA $4702,Y
. 423F  8D 43 44 STA $4443
. 4242  8D 8F 44 STA $448F
. 4245  B9 03 47 LDA $4703,Y
. 4248 *85 E2    STA $E2
. 424A  B9 04 47 LDA $4704,Y
. 424D  85 E3    STA $E3
. 424F  A5 D4    LDA $D4
. 4251  85 D2    STA $D2
. 4253  A5 D5    LDA $D5
. 4255  85 D3    STA $D3
. 4257  B9 05 47 LDA $4705,Y
. 425A  85 E6    STA $E6
. 425C  B9 06 47 LDA $4706,Y
. 425F  85 E7    STA $E7
. 4261  60       RTS
. 4262  EA       NOP
```

```
. 433F  EA       NOP
. 4340  78       SEI
. 4341  8D 3F FF STA $FF3F
. 4344  84 E5    STY $E5
. 4346  98       TYA
. 4347  99 E0 47 STA $47E0,Y
. 434A  20 F0 41 JSR $41F0
. 434D  20 3F 44 JSR $443F
. 4350  20 22 43 JSR $4322
. 4353  20 32 45 JSR $4532
. 4356  20 A0 41 JSR $41A0
. 4359  8D 3E FF STA $FF3E
. 435C  58       CLI
. 435D  60       RTS
. 435E  EA       NOP


. 4C00  8A       TXA
. 4C01  9D E0 47 STA $47E0,X
. 4C04  EA       NOP
. 4C05  EA       NOP
. 4C06  EA       NOP
. 4C07  60       RTS
. 4C08  A9 00    LDA #$00
. 4C0A  85 D0    STA $D0
. 4C0C  85 D4    STA $D4
. 4C0E  A4 E5    LDY $E5
. 4C10  B9 D0 49 LDA $49D0,Y
. 4C13  85 D3    STA $D3
. 4C15  85 D5    STA $D5
. 4C17  60       RTS
. 4C18  EA       NOP
. 4C19  EA       NOP
. 4C1A  EA       NOP
. 4C1B  EA       NOP
. 4C1C  EA       NOP
. 4C1D  EA       NOP
. 4C1E  EA       NOP
. 4C1F  EA       NOP
. 4C20  EA       NOP
. 4C21  EA       NOP
. 4C22  EA       NOP
. 4C23  EA       NOP
. 4C24  EA       NOP
. 4C25  EA       NOP
. 4C26  EA       NOP
. 4C27  EA       NOP
. 4C28  78       SEI
. 4C29  8D 3F FF STA $FF3F
. 4C2C  A0 00    LDY #$00
. 4C2E  B1 D0    LDA ($D0),Y
. 4C30  91 D4    STA ($D4),Y
. 4C32  C8       INY
. 4C33  D0 F9    BNE $4C2E
. 4C35  60       RTS
. 4C36  EA       NOP
```

.....CONTINUED.....

```
. 4C3F  EA       NOP
. 4C40  A9 48    LDA #$48
. 4C42  85 D5    STA $D5
. 4C44  A5 D1    LDA $D1
. 4C46  29 0F    AND #$0F
. 4C48  48       PHA
. 4C49  0A       ASL
. 4C4A  0A       ASL
. 4C4B  0A       ASL
. 4C4C  0A       ASL
. 4C4D  85 D4    STA $D4
. 4C4F  A5 E5    LDA $E5
. 4C51  0A       ASL
. 4C52  0A       ASL
. 4C53  0A       ASL
. 4C54  0A       ASL
. 4C55  AA       TAX
. 4C56  A0 00    LDY #$00
. 4C58  B1 D4    LDA ($D4),Y
. 4C5A  C0 0D    CPY #$0D
. 4C5C  D0 02    BNE $4C60
. 4C5E  A5 D1    LDA $D1
. 4C60  9D 00 46 STA $4600,X
. 4C63  E8       INX
. 4C64  C8       INY
. 4C65  C0 10    CPY #$10
. 4C67  D0 EF    BNE $4C58
. 4C69  E6 D5    INC $D5
. 4C6B  68       PLA
. 4C6C  0A       ASL
. 4C6D  0A       ASL
. 4C6E  0A       ASL
. 4C6F  85 D4    STA $D4
. 4C71  A5 E5    LDA $E5
```

```
. 4C73  0A       ASL
. 4C74  0A       ASL
. 4C75  0A       ASL
. 4C76  48       PHA
. 4C77  AA       TAX
. 4C78  A0 00    LDY #$00
. 4C7A  B1 D4    LDA ($D4),Y
. 4C7C  9D 00 47 STA $4700,X
. 4C7F  E8       INX
. 4C80  C8       INY
. 4C81  C0 08    CPY #$08
. 4C83  D0 F5    BNE $4C7A
. 4C85  68       PLA
. 4C86  AA       TAX
. 4C87  A5 D3    LDA $D3
. 4C89  9D 02 47 STA $4702,X
. 4C8C  C6 D3    DEC $D3
. 4C8E  A5 D3    LDA $D3
. 4C90  9D 01 47 STA $4701,X
. 4C93  C6 D3    DEC $D3
. 4C95  A5 D3    LDA $D3
. 4C97  9D 00 47 STA $4700,X
. 4C9A  60       RTS
. 4C9B  EA       NOP
. 4C9C  EA       NOP
. 4C9D  EA       NOP
. 4C9E  EA       NOP
. 4C9F  EA       NOP
. 4CA0  20 00 4C JSR $4C00
. 4CA3  20 08 4C JSR $4C08
. 4CA6  20 28 4C JSR $4C28
. 4CA9  20 40 4C JSR $4C40
. 4CAC  A4 E5    LDY $E5
. 4CAE  20 40 43 JSR $4340
. 4CB1  60       RTS
. 4CB2  EA       NOP
```

CONT NEXT MONTH,

DUKES MARKETING & SOFTWARE FOR SALE LIST

C16 + 4 - ENTERTAINMENT SOFTWARE			
25002	BRIAN CLOUGH F'BALL (+4ONLY C D S		7.99
26086	OMNIBUS 1	(+4) OPEN IN	6.99
26087	OMNIBUS 2	(+4) OPEN IN	6.99

C16 + 4 - BUDGET SOFTWARE			
28815	3D CLOUPER	AILANTIS	1.99
27983	ACE 1	ARTRONIC	2.99
27984	ACE 1	(+4ONLY) ARTRONIC	2.99
29949	ACE 2	(+4ONLY) ARTRONIC	2.99
28406	AIRWOLF	ENCORE	1.99
33076	AIRWOLF 2	ENCORE	1.99
28911	ANTHROPOID	ALTERNATIVE	1.99
25378	AUTOZONE	PLAYERS	1.99
26892	BMX SIMULATOR	CODEMASTERS	2.99
29000	BOMB JACK	ENCORE	1.99
31996	BOMB JACK 2	ENCORE	1.99
23182	BUBBLE TROUBLE	PLAYERS	1.99
28236	C16 COMPILATION	MASTERTRONIC	2.99
31555	CAVE FIGHTER	BLUE RIBBON	2.99
29685	COMMANDO	ENCORE	2.99
25549	DIZZY DICE	PLAYERS	1.99
20349	FOOTBALL MANAGER	(+4) ADDICTIVE	2.99
22344	FOOTBALLER OF YEAR	(+4) KIXX	2.99
26826	FORMULA ONE SIMULATOR	MASTERTRONIC	1.99
28256	FRANK BRUNO'S BOXING	ENCORE	1.99
21992	G MAN	CODEMASTERS	2.99
33302	GHOST & GOBLINS	ENCORE	2.99
32450	GREEN BERET	HIT SQUAD	2.99
23404	GUNLAW	MASTERTRONIC	1.99
27734	HOLLYWOOD P.(PLAYERS!+4ON	PLAYERS	1.99
24063	INDOOR SOCCER	(+4) ALTERNATIVE	1.99
21822	INTERNATIONAL KARATE	(+4) PRISM	2.99
28626	INVASION FORCE (T.D.1)	(+4) ALTERNATIVE	1.99
29692	JOE BLADE 2	PLAYERS	1.99
22120	LEAGUE CHALLENGE	(+4) AILANTIS	2.99
22652	MASTERCHESS	(+4) MASTERTRONIC	1.99
23701	MICRO VALUE	TYNESOFT	3.99
25837	MICRO VALUE 2	TYNESOFT	3.99
27703	MICRO VALUE 3	TYNESOFT	3.99
25480	ON CUE	MASTERTRONIC	2.99
33426	PAPERBOY (BUDGET)	ENCORE	2.99
N 33984	REAL STUNT EXPERT	ALTERNATIVE	2.99
22127	SABOTEUR	(+4) ALTERNATIVE	1.99
32486	SCOOBY DOO	ENCORE	1.99
24063	SOCCER BOSS	(+4) ALTERNATIVE	1.99
21254	SPEED KING	MASTERTRONIC	1.99
19932	STEVE DAVIES SNOOKER	(+4) BLUE RIBBON	1.99
32765	SYSTEM 8	BLUE RIBBON	1.99
32270	TASK FORCE	PLAYERS	2.99
31284	TOM CAT	PLAYERS	2.99
30780	TRIPLE DECKER 2	ALTERNATIVE	1.99
25771	TRIPLE DECKER 3	ALTERNATIVE	1.99
23357	TRIPLE DECKER 3	(+4) ALTERNATIVE	1.99
31611	TRIPLE DECKER 7 (T.B.)	ALTERNATIVE	1.99
31731	TURF TOWN	BLUE RIBBON	2.99
21828	VIDEO POKER	(+4) MASTERTRONIC	1.99
27903	WAY OF EXPLODING FIST	MASTERTRONIC	1.99
30263	WAY OF THE TIGER (4!XX)	KIXX	2.99
33026	WHO DARES WINS 2	ALTERNATIVE	1.99
32455	YIE AR KUNG FU	HIT SQUAD	2.99

112 Cliff Road
Hornsea
N. Humberside
HU18 1TE

155 Ladbrooke Square
London
N1 7UB

28th April 1990

Dear Roy

I'm writing to inform you that the people at 'DUKES MARKETING' (mail order) have changed their discount & Ordering system, the discounts are now at "20% off RRPs" (instead of 30%) and all orders must be accompanied by an order form. I have enclosed an order form (with 'How to order' information on the reverse side), This is their latest (most up-to-date) form.

Sorry for all the confusion caused, but this was unavoidable because a few days after I sent off the old form to you, I recieved the new updated one. The C16/+4 software List that I sent before; has not changed - all codes and RRPs are the same.

I have also enclosed an old advert of the Plus/4 - you might like to publish in the mag?

Best Wishes

Andy Ting (London)

P.S. Thanks for supporting the C16/+4.

P.P.S. I just heard of these re-releases: Fighting Warrior (Mastertronic) and Illuri Warriors (Encore), both cost £2.99.

Dukes Marketing (Mail Order)

25 Market Street,
Bridgend,
Mid Glamorgan,
South Wales.
Tel:- (0656) 667444 / Fax (0656) 667444

Dear _____,

Thank you for sending for our catalogue.

How To Order.

1) Quote Title, Code No. Machine Format And Price Of Software And Or Hardware.

2) Make Cheque Payable To "Dukes Mail Order."

3) Or Quote Access/Visa Credit Card No. Expiry Date, And Holders Name And Address. State Which Card Used.

4) Deduct 10% Off Sega Titles And 20% Off all other Computer Titles. (This Discount Only Applies To Software Not Hardware)

5) Overseas Customers Add Postage Charge Europe £1.50, Outside Europe Add £3.00.

6) Send Order To:-

Dukes Marketing (Mail Order),
25 Market Street,
Bridgend,
Mid Glamorgan,
South Wales.

If You Have Any Problems Regarding Dukes Mail Order Please Write To Peter Midgley At Above Address.

For Credit Card Orders Phone (0656) 667444
Quote Your Credit Card Number, Name + Address + Expiry Date.

Use Reverse Side Of This Form To Order

A New Order Form Will Be Sent To You With Every Order.

Note 1) Any Orders Not Sent On An Order Form Will Not Be Accepted.

Note 2) If Any Mistakes Have Been Made While Calculating Total Of Order, The Order Form And Accompanying Monies Will Be Sent Back With A Letter Of Explanation.

Dukes Marketing Mail Order

25 Market Street,
Bridgend,
Mid Glamorgan,
South Wales.

Tel:- (0656) 667444 / Fax:- (0656) 667444

Please Print In Block Capital Letters.

Your Name:- _ _ _ _ _ _ _ _ _ _

Your Address:- _ _ _ _ _ _ _ _ _ _ _ _

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Order Form.

	Title	Code No.	Machine	Format	Price
Eg.1	Dynamite Dux	33114	C64	Tape	9.99
2)					
3)					
4)					
5)					
6)					
7)					
8)					
9)					
10)					

Total With Discount £

Grand Total + P&P £

Note:- Discount Must Be Taken Off All Items Separately And Then Totalled Before Adding Post And Packing.

Game Review

Game Reviewed: More Adventures of Big Mac the Mad Maintenace Man
Publisher : Mastertronic
Price : £1.99
Reviewer : Matthew Newton-Lewis, West Sussex

Big Mac is very rarely seen about, although, like all Mastertronic games, it is relatively old. It is the sequel to Mr. Puniverse, with roughly the same idea. You have to get to the exit at the end of the screen before you run out of air, while jumping about platforms, dogding guns shooting you and working out which way to go.

In some harder rooms you have to use you reflexes a lot, especially when running against moving conveyer belts. Some of the graphical effects are really amazing with the screen spliting in either directions and some simply excellent scrolling of messages.

You're movements are a bit unrealistic, like when you walk in mid air while jumping, however this doesn't matter much as all the movements need to happen in order to make the game work. The animation in the enemies and in you are good.

The sound is pretty good. Their is quite an effective tune on the tittle screen and basic but effective sound effects throughout the game.

The coding is not altogther faultless with little mistakes along the way. There are no serious errors in any part of the game.

You do not just jump about the place, you have to open doors, get on escalators and use timing and skill to enjoy the game.

Overall this game is well worth buying. It's one of those games that sits around and you start to forget about it, but when you load it up it will take a lot to draw you away again. I think you have to give this game time to be really satisfied with it.

Ratings out of 10:

GRAPHICS : 9
SOUND : 9
PLAYABILITY : 9
V.F.M. : 10
OVERALL : 9

PLUS 4 ON THE SPOT

Commodore is set to aim its Plus/4 at first-time buyers for serious home and introductory business use. Barry Miles reports on the moves behind this release.

Offering fundamentally the same operating system as the Commodore 16 (recently reviewed in *PCN*), the Plus/4 is aimed at the first-time buyer, for serious home use and introductory business use. Priced at £299.95, it competes on price with the Commodore 64 (£199 in most places), MSX machines at around £200 and the Sinclair QL at £400.

The Commodore 16, which has the same operating system, but one quarter of the memory, is bundled with a cassette unit (retail £44), Introduction to Basic Part I (£13.95) and four games.

For serious use, you need at least the cassette unit, making the Plus/4 look rather expensive at £350. Clearly Commodore is following its usual pattern of high prices to begin with, and progressive reductions over time.

This is a marketing method proven by its profits and the world-wide sales, currently said to be two million Vics, already; a similar number of 64s is expected to be sold by the end of 1984.

A design approach, similarly tried and tested by Commodore, is to say in effect: 'We are the volume producers, so our design is the standard.' Accordingly we have non-standard cassette connections, and cassette units, non-standard joystick connections, and serial disk connection.

A whole range of peripherals, including a fast disk drive and a new monitor is planned for the machine, all in the same tasteful shade of charcoal grey, and rather attractive too, according to the photographs.

So will it sell, at the price quoted? I think so. The sales of the Vic are against the natural law of the market. It is astounding that a machine with under 4K of memory has continued to sell, against a market background of 16-bit, and perhaps 32-bit machines, with memories of 128K and 256K becoming commonplace. But is it so strange really? Now salesmen are told, 'sell benefits, not features'.

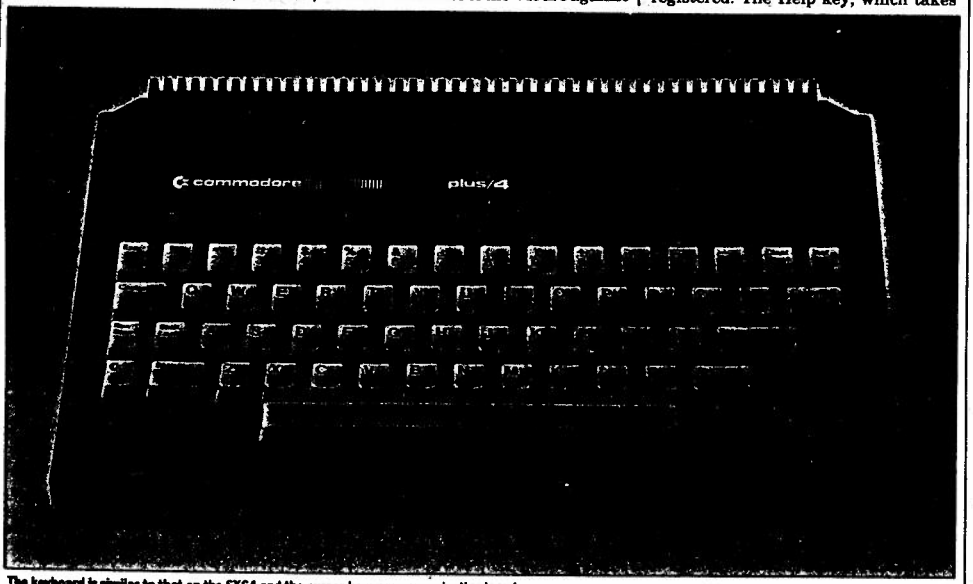
Persistent but unconfirmed rumours allege that a 128K version of the 64, capable of full 64 emulation, and with 128K addressable from Basic, with switching between 40 and 80 columns is on the way but Commodore maintains a dignified silence. If this machine really is planned then it will hit the Plus/4 market hard.

Features

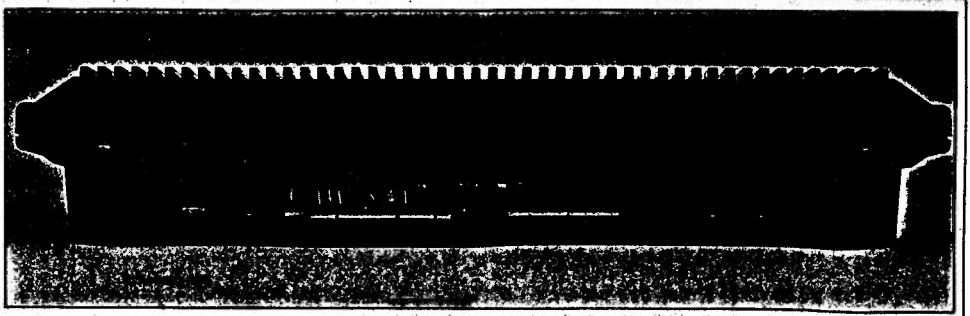
So what is the 'unique selling point' of the Plus/4? It is the integral software which gives the machine its character. A word processor, spreadsheet, filing system, and graphics package are available on power-up. Known as 3 Plus 1 (four packages), this selection offers windows, and genuine integration and a form of multi-tasking.

You can transfer figures from your spreadsheet directly to your word processor, in RAM, immediately, with the window (split screen) showing both documents at once!

The Plus/4's appearance is definitely in its favour. It resembles the lap portables in size and shape, and the keyboard slopes pleasantly down to a low profile nose. The function keys have a positive 'click-feel', which tells you very effectively when the keys have registered. The Help key, which takes



The keyboard is similar to that on the SX64 and the cursor keys are conveniently placed.



All connections are clearly marked on the rear of the machine.

the place of F8, is a great added convenience.

The keyboard is noisier than on the 64, but more gently sprung. It is very pleasant to use, and lacks only the shift-lock warning light to be as attractive as the portable 64's. The cursor control keys, four of them, are attractive and large. It is a little frustrating to have no numeric keypad, especially since the 364 alternative model, with such a pad, is to be marketed elsewhere.

The power connection has at last been removed to the back of the machine out of harm's way. Typically with Commodore, the plug is a new design. Also the cartridge or memory expansion slot is just slightly narrower.

The joystick ports are different again. The Atari-type D-connector has given way to a new Commodore design. However, Commodore's own joystick will be analog.

The connection of disk drive to the computer is through the hated slow serial port. The effect is as before — user frustration. We can only hope that the 'fast-disk drive' turns out to be really fast. Connection to a television or monitor is provided for.

The new reset button is on the side of the machine and reasonably out of the way of accidental resetting.

Commodore's new Basic 3.5 has taken account of all the criticisms previously aimed at Commodore machines.

Structured programming is available, and proper control of sound and graphics is provided by new, and self-explanatory keywords. A good machine code monitor is provided, including the display of the ASCII equivalents.

The trade-off for all these extras is that sound has been cut to two voices compared with three on the 64, while the main reason for the 'serious home user' tag would appear to be the absence of sprites. Presumably writing games is not 'serious'.

Software

On the Plus/4 hitting F1 followed by the return key gets you immediately operational, in the word processor.

Commodore has paid attention to the statistical law about 80 per cent of the value being in 20 per cent of the items. This implies that most people use only a fraction of the facilities to be found in these common packages.

Thus the field was wide open for packages which, while not providing all the bells and whistles of the fully-fledged software, would nevertheless offer most of what most people needed. Approaching all four packages in this way enabled them to fit into the 32K ROM.

Word processor

The word processor has a good range of commands. It works with the 40-column screen acting as a window onto a document which is a maximum of 77 characters wide. The major limitation is that the maximum length of a document is 99 lines — 22 lines appear on the screen at any one time. Unfortunately, you cannot quickly change to 40 characters width for rapid editing of the document, as you can in Vizawrite on the Commodore 64, for example. You can of course change the document width to suit you, and reset it for printing.

Word wrapping does not take place on the screen, although it does when you print the document.

For the first time Commodore has recognised that people may use non-Commodore printers with a Commodore computer. Accordingly you can send reverse field control codes from within your text, so as to take advantage of your printers' more esoteric capabilities, like elite, or italic compressed or double-width characters.

Also a special command, 'other', reverses the character set to standard ASCII, which should solve many interfacing problems. Some of the facilities in the word processor are very impressive.

Mail-merge can take place, and there's text movement and copying. Most of the instructions will be pretty easy to remember. All of these instructions are inserted into text in reverse video (obtained by hitting Control 9), following by the appropriate mnemonic code and Control 0.

Pressing Return takes you to a new paragraph. It also deletes the remainder of the line beyond he cursor. This is bad news for experienced users of Commodore's screen-editor in Basic, who are accustomed to using Return anywhere in a line, to enter it as a Basic line. They will, on occasion accidentally delete lines which they wish to retain. However, help is at hand! There is what in trans-Atlantic parlance would no doubt be called an 'Oops' buffer.

Putting the cursor at the point where the accidental deletion occurred, and hitting the Commodore logo key followed by @, will restore the missing line!

Block insertion, deletion and movement are available. Blocks of text can be up to 16 lines long. This is not a major constraint, bearing in mind that the maximum document length is 99 lines. For insertion, you set a pointer at the end of the text already typed in, using the SP (Set Pointers) command. Then you move the cursor above this point and type in your block of text for insertion. You now have the end of the block indicated, but not the beginning.

To define this, you position the cursor at the start of the block and use the Create Block command. You then move the cursor to the place where the block is to be inserted, and use the Insert Block command followed by Return. The insertion occurs above the current cursor line, but not at the current cursor position in the line.

Although easier to do than to describe, this is pretty clumsy compared with the Insert mode of operation common to other word processors, which open-up the text, and creates space for inserted text automatically and apparently effort- 40 ▶

Cursor style: now to Commodore.

Finally! Deleting the block which has been identified is accomplished by the DB command.

For moving a block, having identified it by Setting a Pointer, and Creating a Block, you use Insert Block (IB) to make your move.

The small maximum document length — 99 lines — means that linking files will be essential for work of any magnitude. The Linkfile command is put at the end of any document, to enable the next part to be loaded in, and so on.

The Merge command is a disappointment. Simple enough to use, it appends the incoming text at the foot of the document. Of course you can then move it about, but that is a rather clumsy way of doing things. If you give the command PR, your document will be saved to disk automatically, before printing takes place, using a special file, 'tw' — temporary workspace.

As soon as this has been done you can load any file you wish, and start work on it. You can set up a continuous printing operation, using a whole disk-full of linked files. A pause instruction at the end of the last file enables you to switch disks and carry on!

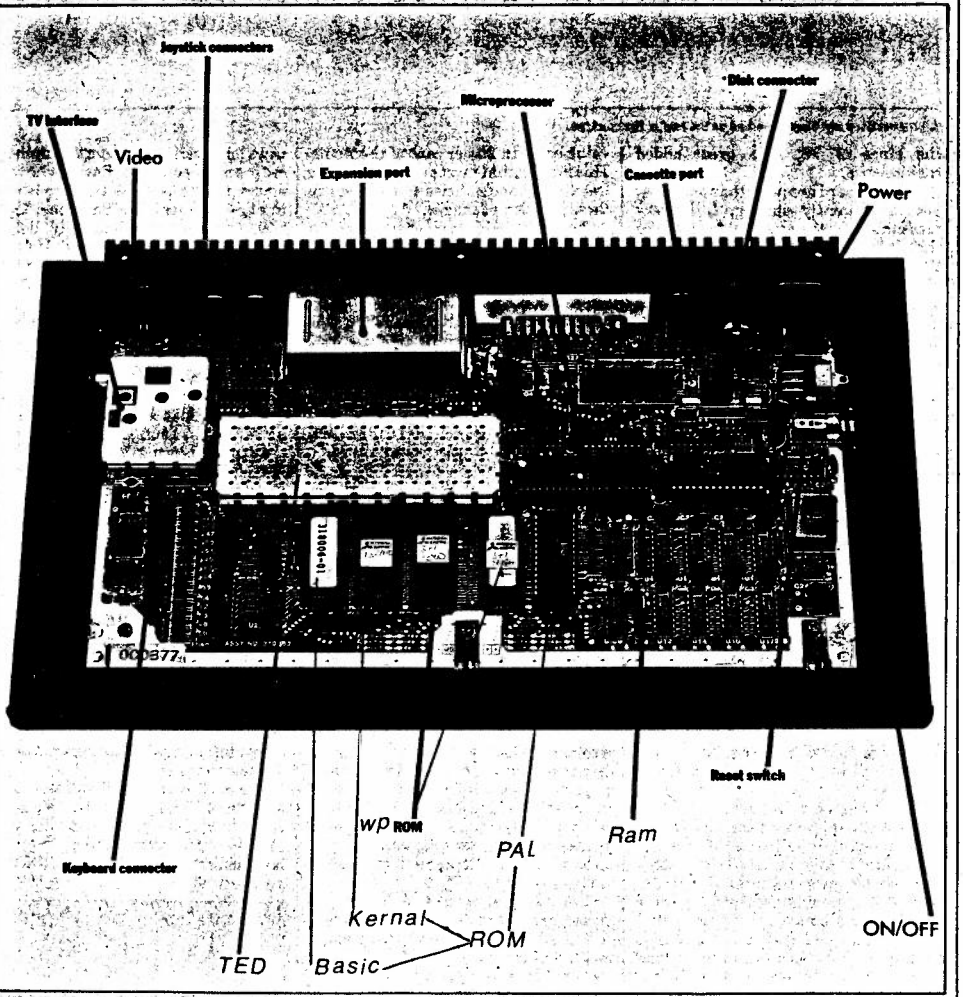
You may also be given an instruction that the printing should pause after printing of each page.

Free Cursor movement is a feature of this package. F1 Function key takes you to the left margin, F2 to column 41, Home to the top line of text, and Shifted "=" acts as a TAB key. There are no commands for emboldening, double strike, underline, or similar special printer activities. For these you will have to use the ASCII function, which will send out the appropriate control codes to your particular printer.

Spreadsheet

On the spreadsheet, the cell locations are described by two numbers rather than by the almost universal combination of letters and numbers for rows and columns. This is a pity, because it leaves the way open for you to forget which comes first — row or column, — when you are giving cell references. When you are copying formulae into cells, this is particularly dangerous. You cannot point to cells by means of cursor movements, when making up formulae, but on the other hand, you can use labels for data instead of cell-references, which is a very good feature.

Function keys F1 and F2 are used to move from cell to cell, leftwards, and



rightwards and you can edit any cell-contents, using the left and right arrow keys.

The command to get to the top lefthand corner of your sheet is achieved in a roundabout way. You hold down the CBM logo key, and press 'C' followed by typing 'home'. I would have been much happier with the use of a Function key for this very frequent activity. Similarly the GOTO command must be typed out in full too.

The spreadsheet is comparatively small — 50 rows by 17 columns. You can move about the sheet by using the cursor up and down keys. The spreadsheet is not intelligent in deciding whether you are typing in numbers, text or formulae; you must tell it.

For copying the contents of a cell, hold down the CBM key, hit c, then type 'copy', followed by the cell whose contents you want to copy, into the cell where the cursor is. The repeat command, done by holding down the CBM/ key and hitting Q, will enable you to copy the same data into a number of cells, such as with underlines.

Replicating formulae relatively is given the unusual name "fit". To copy a formula into a number of cells, you must use the repeat command, CBM Q).

Commodore offers windowing for the various elements in this set of packages. However, the window turns out to be a horizontally-split screen, with the ability to pass data from the spreadsheet or graphics package to the wordprocessor. You use the block map command to pass data from spreadsheet to wordprocessor, setting a rectangle to be transferred. The amount transferred is 11 characters per cell. Up to 36 characters per cell can be transferred by using the Map command. This will also enable you to transfer the formulae themselves into the wordprocessor, so that they can be printed out. This latter is a slower process, because you must proceed row by row.

The usual variety of formats is available for your number cell by cell. You cannot give a command to format globally throughout the sheet in integers for example.

The sum command works on tows and columns of data. Formulae are evaluated from left to right. This is common on spreadsheets, but some prefer the correct use of the hierarchy, as in Basic. But if you inset column or rows, the formulae require adjustments.

It is possible to copy rows or columns. This is not as helpful as we might hope: data only is copied, not formulae.

You can label a cell, and refer to it by label. This is a big advantage since the numeric system is capable of causing confusion. The use of IF True, with a number of operators, enables you to proceed conditionally. This feature will permit you to modify the order of calculation.

You can freeze and unfreeze values in cells, which is all to the good. You can also left-justify numbers in a cell.

Numeric constants must be preceded by a hash sign. This comes more naturally to Americans than to us.

The spreadsheet requires rather more work, and alertness to use than more sophisticated ones. However, it has many useful features. The graphics are merely low resolution bar graphs and point graphs of any row of data on your spreadsheet. However, the graphs will print on any Commodore printer.

Graphs can be transferred into the wordprocessor for labelling and printing. The graphs are scaled automatically, which avoids the most tedious aspect of graphing. But the point graphs are created by eliminating all but the highest point in each bar of the bar graph.

Although the graphics are primitive, they serve a useful purpose. The whole package, wordprocessor, spreadsheet graphics and file manager are in a single 32K block of memory! So it is absurd to expect too much in the way of facilities.

File Manager

The File Manager is also the sort of cut-down version which you would reasonably expect. Nonetheless, you can have records containing up to 17 fields, with up to 38 characters in each field. A separate disk must be maintained for your data, because the Filer does not use the Commodore relative file system.

Random access to a record is not by key. You must either know the record number of the record (bad news!) or use a searching process, whereby your string of characters is looked for in any field of any record. This is not the fastest of processes. The maximum number of records on a disk is 999.

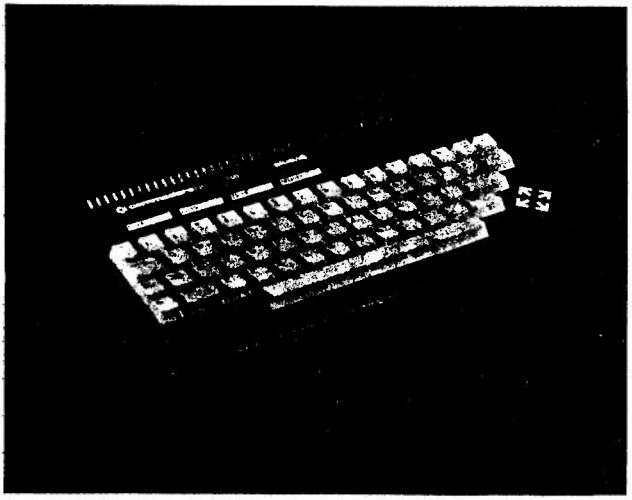
A faster Review command enables you to scan through the records more quickly, starting with any record you choose (by number). Reporting is done from within the wordprocessor, using special commands. An EOF command enables you to continue the report through the file, or the selected section of it which you have selected by your own criteria.

The Pick command carries out your selection of records, matching up to 38 characters. The Picked file can then be used for sorts or reports. You can use a delimiter of the Highrc Highrecord command. This enables you to specify the record at which the Sort, Pick, Search, Review or Reporting function is to stop.

The file manager will serve as a useful introduction to this type of operation.

Verdict

The Plus/4 is an interesting machine with a lot of good features. As with all designs, there are compromises. However, there is enough of everything to keep most purchasers very happy. Perhaps it is a little pricey. Time will tell.



SPECIFICATIONS

Game Review

Game reviewed: WAY OF THE EXPLODING FIST
Publisher : RICOOHET (MASTERTRONIC)
Price : £2.99
Reviewer : Matthew Newton-Lewis, West Sussex

The object of the game is to try to beat all the different standards, starting with NOVICE then 1ST DAN, 2ND Dan etc.

Way of the exploding fist was made in 1986 making it an old game. It is very well known of because of the sucess on the 8-bit computers but very rarely seen or heard of for the C16/+4.

Apart from the rather wierd way of loading this is a quite classy game. The graphics aren't bad but they aren't good either. There is only one background screen of an out of proportion Buda with a bit of a house and a wall.

There are over sixteen moves, some of which are really impressive. The deatail on the moves is amazing. With so many moves it makes the game get a bit out of control because you don't know what you're doing, or what you're meant to be doing the whole time. The moves include: low/high punch, low/high kick, roundhouse, backick, and some simply amazing blocks.

The reactions of the joystick are slow but you get used to them. There is a good two-player option where you play against each other trying to score more than your oponent. Unfortunately as it is a joystick only game you have to have two joysticks for the two-player option.

It's quite a good deal but if you don't like these types of games then I wouldn't adive buying it.

Ratings as a percentage:

GRAPHICS : 72
SOUND : 56
PLAYABILITY : 87
V.F.M. : 83
OVERALL : 80

Please send me Vol 2 Issue 4 July 1990 of 'C16/C116/+4 COMPUTING MONTHLY'.

I enclose a PO/CHEQUE for the value of £1.00.

NAME & ADDRESS:

SIGNED: _____

The above form should only be filled in by those who order the mag monthly, not by those who subscribe yearly. Monthly subscribers can photocopy the form so you don't chop up the mag, but please send it when ordering the issue required as it saves me a lot of time and you will get your mag much more quickly.

C16 AND PLUS4 COMPATIBLE DISK DRIVES

BY DAVID CAMPBELL

After reading Roy's editorial in the December 1989 issue, asking what the difference was between the 1541 and 1551 disk drives, I Decided to look into the various disk drives that are claimed to be Commodore compatible.

1) COMMODORE 1541

This disk drive is based on the PET's old 4040 and 2031 system, it started life as the 1540 drive for the VIC20. When the C64 was launched, the 1541 superceeded the 1540, having a new casing, some mechanical changes and slightly improved speed.

There are a few ROM Problems which appeared on the 4040 that still applies to the 1540 and the 1541. The save and replace facilities malfunctions when the directory fills an exact number of data blocks being the main Problem.

This fault has been carried through to the 1541D and 1570/71 drives.

2) COMMODORE 1551

This drive was intended as an alternative for the C16/PLUS4 systems and offers Parallel data transfer with vastly increased speeds. I understand that it is 25% faster than the 1541 even when fitted to the serial socket. I have only seen a 1551 once (and it was in a repair shop) so I have never seen one in use and the books and magazine articles I have read either don't mention the 1551 or only mention it in passing. Perhaps one of the members with a 1551 will write an article describing any bugs it may have.

3) COMMODORE 1570/1571

These drives were developed for use with the C128.

The 1570 drive contains three storage/retrieval systems and the 1571 has four.

Both drives have a fully compatible 1541 system but in 128 mode a faster system clicks into operation (with CP/M and MS-DOS abilities)

The extra mode in the 1571 arises because it can use both sides of the disk for storage.

The drive automatically selects 1541 mode when it is turned on and it is the first access by the computer whether normal mode will continue (i.e. for C64, C16 or +4) or change to fast transfer mode (for C128).

There are Problems with the 1571 ROM for example when 2 files are open at the same time, writing to side 2 is often slow and spurious DEVICE NOT PRESENT errors are generated.

4) COMMODORE 1581

This drive uses 3.5in disks instead of the normal 5.25in disks. The main advantage of this is that the 5.25in has only 664 blocks available, while the 3.5in disk has a capacity of 3160 blocks. In bytes this means 790K of available storage opposed to a mere 166K on the standard system.

5) EXCELLERATOR +PLUS FSD-2

This is an improved Excellerator Plus and it is claimed to be totally compatible with the 1541 but some slight incompatibilities have been found. The Excellerator is less than half the size of the 1541 but the power supply is separate and the casing is metal which offers an excellent shield against electrical and magnetic disturbances. The Excellerator is also fitted with two switches located on the under side of the drive which allows easy switching of the device number from 8 to 9, 10 or 11.

6) OCEANIC OC-118N

This has an awful lot in common with the Excellerator. It has an identical casing with identical sockets and switches.

Internally the mechanical construction is different. Electronically however, the circuit boards look similar with a few differences. The Oceanic has a higher code number than the +Plus but this does not mean that it is an improvment on the +Plus which it might be.

7) BLUE CHIP 5.25

This machine is twice the size of the Excellerator/Oceanic drives but smaller than the 1541. It is claimed to be more compatible than the 1541D.

The Blue Chip possesses a more reliable locking system on the disk slot. A latch locks in do the Excellerator and Oceanic drives but with the Excellerator/Oceanic it is possible that the key latch can be swung into position when the disk is still 0.25in from full insertion into the drive so it is possible to trap and damage a disk but with the Blue Chip the disk can only be 0.125 out and the mechanism fully pull the disk into place if the key latch is turned to lock. This drive like Commodores can also be opened in order to change device numbers.

One advantage that Commodore compatible drives have over others is the fact that they are 'intelligent'. This means that the DOS is held on board the drive rather than occupying memory within the computer itself. This allows the drive to be programmed and facilities such as spooling from disc to printer can be achieved without tying up the computer.