VOL 5 ISSUE 5

July 1989

MEMOTECHNIQUES

MEMOTECH OWNERS'

CLUB

MAGAZINE

13 COPSE ROAD,
TOWNHILL PARK,
SOUTHAMPTON,
SO2 2GY.

CONTENTS

1. EDITORIAL
2. LOGO TREE
3. DOG FIGHT
4. DOG FIGHT
5. DOG FIGHT
6. SOFTWARE FOR SALE
7. IS THIS CHESS ??
8. CP/M SORTING
9. YOUR LETTERS
- ADVERT
10. HARDWARE FOR SALE
11. BOOKS FOR SALE
12. PROGRAM LIBRARY
13. PROGRAM LIBRARY
14. NMI PART 3
15. REVIEWS
16. MERGING
17. CIRCUIT DIAGRAM
18. BANNER !!
19. BANNER
20. BANNER

COVERS

If you have an interesting cover for the magazine or an interesting picture for the cover then please send it to us. The cover can be of anything but it must be mostly white, large areas of black do not duplicate very well.

A FREE piece of software for each monthly mag.

This months winner is Mr N Cooper from Leyton London. He will receive a copy of Astromilon as requested. (Please let use know which piece of software you would like!).

Competition

This months competition rules are simple. It is nothing to do with computing, (unless you wish to make it so!).

Make as many words as you can from MEMOTECH OWNERS.

Subscriptions

THE MOC Magazine is available only by subscription:-

U.K.	£12.00
European	£15.00
BFPO NO.s	£12.00
Outside Europe	
Region A	£16.00
Region B	£17.50
Region C	£18.00

How To Submit Articles

We prefer all articles on disc or tape, but very small pieces are OK on paper. Please put your name and address on any disc's and tapes. A return address label would be appreciated if you have such things. If you are sending any "camera ready" artwork please ensure it is not folded and use black ink.

Advertising rates

Private ads cost £2.00

Commercial rates

Quarter Page £7.50 Half Page £15.00

Telephone contacts

Hazel has deemed that it is time to do the telephone hotline properly. So the hotline will be on Monday and Tuesday evenings between 6 and 8pm and when available at weekends.

Phil Eyres 0703 585106

(Ansa Machine when not available)

Paul Wood 0905 24260

EDITORIAL (July 1989)

Phil Hazel & Siobhan Eyres
13 Copse Road
Townhill Park
Southampton

GENERAL

Most people this month seem to have had an invitation to join the Software Source. This letter from Mr G.Heywood typifies the letters that we have received.

"I recently received an invitation to join The Software Source.

Whilst I am naturally interested in anything which may be of help, I was struck by the words 'a new user group designed to break down the barriers of older, more bogged down user groups which the MTX has been plagued with' and 'instead of producing a magazine full of any old rubbish' etc.

I hope these are not oblique references to the M.O.C. because if they are, I consider them quite unfounded and unfair.

Therefore I am relucant to join this organisation without first being advised from yourself that its instigator(s) have never been connected with you. I say this, bearing in mind your Editorial about the departure of a person named Alan Hamilton. Could he be the person concerned? The address given is Beith Ayrshire.

Please advise."

I'll try and give you all I know in as unbiased a way as I can. Alan has left the club, he did so not on the best of terms, with threats of sueing us for one thing and another. Because of this I did my best to close all links between Alan and the club as quickly as possible. Alan has now started up the Software Source, the literature about which I am sure we have all received. As the whether or not the comments he makes are about our club or the now defunct GENPAT are for you to decide.

I hope in time he can focus his energy into doing something really good for the MTX,he must not underestimate the effort required, it will be huge.

MICRO SHOW

I have left this bit from last month, just to keep jogging your memory!!

There is an Alternative Micro show in November, on the 11th to be precise, in Stafford. I will be going on the Saturday, so see you there. So pencil this date in your diary.

ARTICLES

This month we have an interesting Basic program called DOG FIGHT. You should find it fun if you enjoy zapping things.

We also have an interesting BANNER program which is really quite friendly to use.

HARDWARE

The hardware continues to be very popular, and again this month I have bought more than the club can afford. I now have two MTX NODE ROMS. If anyone is interested in having a play with them (In return for an article about them) then please let me know.

The Club Hotline is between 6 and 8 pm Monday and Tuesday evening. During the day and after 8.00pm a club answer phone takes over. I hope this is ok for everyone. The number to phone now is (0703) 585106, ask for Phil.

If anyone would like back issues they are available for the small remittance of 80p each. At present there are 4 back issues, 10 for volume 1, 10 for volume 2, 10 for volume 3, 10 for volume 4 and 4 for volume 5.

It should be noted that all articles are the copyright of the sender and M.O.C., anyone wishing to have articles published elsewhere should inform us first.

Phil Eyres

LOGO TREE

By G.D. Pratt

This is a recursive program using the excellent logo type graphics commands on the MTX 512. Based on ideas in Boris Allen's book "Introducing Logo".

```
11 REM
13 LET SKY=15: LET STEM=1: LET LEAF=6:
REM CHANGE FOR DIFFERENT COLOURS
15 VS 4: COLOUR 2,SKY: COLOUR 4,SKY:
COLOUR 3,STEM: CLS
20 ANGLE PI/2: LET TREERAD=0.55: LET
BRANCH=59: LET REDUCTION=0.84: LET
RELATIVE=0.35: LET TREEANGLE=0.6: LET
ORDER=9
30 PLOT 168,30
40 GOSUB 1000
50 CSR 12,22: PAPER SKY: INK STEM: PRINT
"LOGO TREE"
999 GOTO 999
1000 REM DRAW TREE
1010 IF ORDER=0 THEN RETURN : IF
ORDER<3 THEN COLOUR 3,LEAF ELSE
COLOUR 3,STEM: LET
BRANCH=BRANCH*REDUCTION: ARC
BRANCH,TREERAD: LET ORDER=ORDER-1
1020 GOSUB 1000
1030 LET ORDER=ORDER+1: IF ORDER<3
THEN COLOUR 3,LEAF ELSE COLOUR
3,STEM
1040 PHI PI: ARC BRANCH,-TREERAD: PHI PI:
ARC
BRANCH*RELATIVE,-TREERAD*TREEANGLE:
LET ORDER=ORDER-1
1050 GOSUB 1000
1060 LET ORDER=ORDER+1: IF ORDER<3
THEN COLOUR 3,LEAF ELSE COLOUR
3,STEM
1070 PHI PI: ARC
BRANCH*RELATIVE,TREERAD*TREEANGLE:
PHI PI: LET BRANCH=BRANCH/REDUCTION
1080 RETURN
```

PARAMETERS

TREERAD : SPREAD OF TREE

BRANCH : LENGTH OF BRANCH

REDUCTION: AT EACH STAGE

RELATIVE : LEFT/RIGHT SIDE

TREEANGLE: LEAN OF TREE

ORDER : NO. OF BRANCHES

SMALL ADVERTS

For Sale

MEMOTECH 500 + 64K expansion + Pascal ROM and ALL the good software (Sepulchri, Son of pete, Qogo etc about 40 tapes) Speculator, new manual, new modulator.

Offers above £75.00

Bill Mitchell 0865-711618

FOR SALE

MTX512 + set of 12 software library including 'Word-Pal' a cassette based word processor.

£45.00 inc of P&P.

Perfect working order. Phone Phil 0272 682300

To advertise your private items for sale, it costs just £2.00!! (Maximum 8 lines) Please send your adverts by post to: Phil Eyres 13 Copse Rd, Townhill Park, Southampton. SO2 2GY.

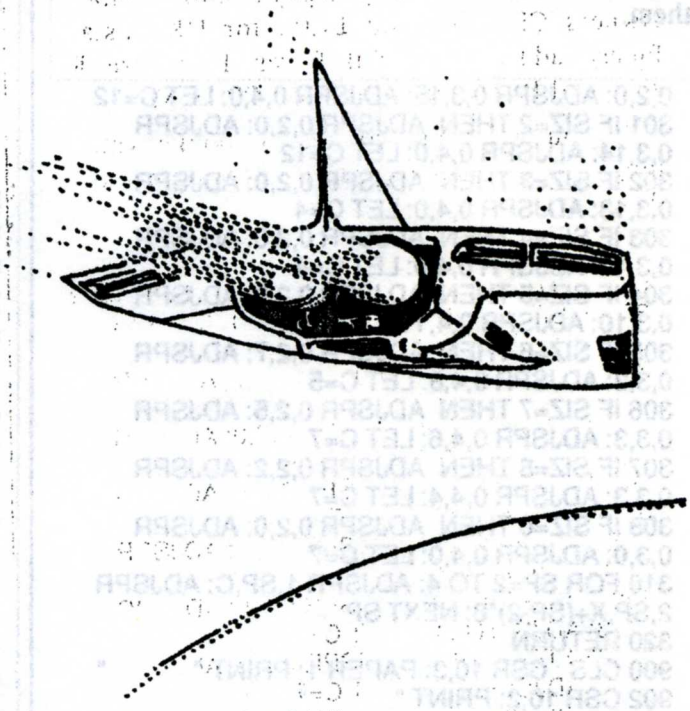
ADVERTS

If you would like to advertise something in a litte section like this. Just send your text (up to 8 lines) along with a £2.00 fee per issue, and we will make sure it gets in the next magazine.

DOG FIGHT

This is a really good little program. You have a star-trek style viewing screen in front of you. Your job is to home in and destroy as many of the astroids as you can in the time that you have available to you.

```
1 GOTO 5
2 SAVE "DOG FIGHT"
4 REM *** GOTO 5 FOR AUTO LOAD ***
5 GOSUB 5000: REM *** GENPATS ***
6 VS 4: CLS
7 REM *** SOUND OFF , SPRITES ON ***
8 FOR I=1 TO 3: SOUND I,0,0: NEXT I
9 CTLSPR 1,4: CTLSPR 2,12: CTLSPR 6,0:
GOSUB 200
10 REM *** TITLE SCREEN ***
11 COLOUR 4,7: PAPER 3: INK 1: CLS :
GOSUB 4000
15 PAPER 7: CLS
20 REM *** DEFINE SCREEN ***
21 GOSUB 900: REM *** SKY ***
22 GOSUB 1000: REM *** REST ***
23 LET T=250: REM *** FUEL LIMIT ***
24 GOSUB 100: REM *** GET KEY, SHOOT ***
25 GOSUB 300: REM *** ADJUST SAUCER ***
26 REM *** FUEL UPDATE, MOVE SHIP ***
27 LET T=T-1: PAPER 0: INK 1: CSR 17,16:
PRINT "    ": CSR 17,16: PRINT T
28 LET YS=YS+(KEY=11 AND YS>-4)-(KEY=10
AND YS<4): LET XS=XS+(KEY=25 AND
XS>-4)-(KEY=8 AND XS<4)
29 LET Y=Y+YS*2+YA*1.5: LET
X=X+XS*2+XA*1.5: IF SIZ=0 THEN LET SIZ=1
30 LET A=10: LET B=19: PAPER 3: CSR A+2,B:
IF Y<105 THEN LET SIZ=0: LET Y=105:
PRINT "v" ELSE PRINT " "
31 REM *** DIRECTION FINDER ***
32 CSR A,B-2: IF X<10 THEN LET SIZ=0:
LET X=10: PRINT "<" ELSE PRINT " "
33 CSR A+4,B-2: IF X>215 THEN LET
SIZ=0: LET X=215: PRINT ">" ELSE PRINT "
"
34 CSR A+2,B-4: IF Y>160 THEN LET
SIZ=0: LET Y=160: PRINT "^" ELSE PRINT "
"
35 FOR SP=2 TO 4: ADJSPR 3,SP,Y: NEXT
SP
36 CSR 11,16: PRINT YS*(YS<0): CSR
11,18: PRINT -YS*(YS>0): CSR 12,17: PRINT
XS*(XS<0): CSR 10,17: PRINT -XS*(XS>0)
37 IF MOD(T,5)=0 THEN LET Z=SIZ/2: LET
XA=Z*(RND<0.5)-Z*(RND<0.5): LET
YA=Z*(RND<0.5)-Z*(RND<0.5): LET
SIZ=SIZ-1*(SIZ<8): LET SC=SC+INT(SIZ/3)
39 REM *** SCORE, CHECK END ***
40 CSR 17,19: PAPER 0: INK 1: PRINT "    ":
CSR 17,19: PRINT SC
50 IF T<1 THEN FOR I=900 TO 1000:
SOUND 1,I,7: NEXT I: CSR 5,13: PAPER 1:
INK 11: PRINT "GAME OVER-PRESS
SPACE BAR": IF INKEY$<>"." THEN GOTO 50
60 IF T<1 THEN GOTO 6
90 SOUND 3,7,8: SOUND 2,T,0
99 GOTO 24
100 LET KEY=ASC(INKEY$)
105 IF KEY=26 THEN SOUND
1,1200,240,40,30,10,1 ELSE RETURN
106 LET SC=SC-5
110 LET F=1: PAPER 1: FOR I=4 TO 7: INK
11-4*(I=4)+(I=7): CSR F+9,I: PRINT "k": CSR
22-F,I: PRINT "l"
117 CSR F+9,I: PRINT " ": CSR 22-F,I: PRINT "
": LET F=F+2: NEXT I
118 IF X<112 OR X>126 OR Y<124 OR Y>136
THEN GOTO 140
119 FOR I=32 TO 37: SOUND 3,7,15: SOUND
2,I+RND*10,-1*(MOD(I,2)=0)
120 FOR SP=2 TO 4: ADJSPR
1,SP,MOD(I,12)-4*(I<>37)+1: NEXT SP
130 NEXT I
133 LET X=RND*205+10: LET SIZ=1
135 LET SC=SC+25*SIZ
140 SOUND 1,0,0: RETURN
200 LET SIZ=1: LET X=116: LET Y=146: LET
XS=0: LET YS=0: SPRITE 2,2,X,Y,0,0,7
201 SPRITE 3,3,X+8,Y,0,0,7
202 SPRITE 4,4,X+16,Y,0,0,7
204 SPRITE 1,1,126,131,0,0,9
210 LET H=0: LET SC=0
220 LET XA=SIZ*(RND<0.5)-SIZ*(RND<0.5):
LET YA=SIZ*(RND<0.5)-SIZ*(RND<0.5)
250 RETURN
300 LET SIZ=INT(SIZ): IF SIZ=1 THEN ADJSPR
```



```
0,2,0: ADJSPR 0,3,15: ADJSPR 0,4,0: LET C=12
301 IF SIZ=2 THEN  ADJSPR 0,2,0: ADJSPR
0,3,14: ADJSPR 0,4,0: LET C=12
302 IF SIZ=3 THEN  ADJSPR 0,2,0: ADJSPR
0,3,13: ADJSPR 0,4,0: LET C=4
303 IF SIZ=4 THEN  ADJSPR 0,2,0: ADJSPR
0,3,12: ADJSPR 0,4,0: LET C=4
304 IF SIZ=5 THEN  ADJSPR 0,2,9: ADJSPR
0,3,10: ADJSPR 0,4,11: LET C=4
305 IF SIZ=6 THEN  ADJSPR 0,2,7: ADJSPR
0,3,3: ADJSPR 0,4,8: LET C=5
306 IF SIZ=7 THEN  ADJSPR 0,2,5: ADJSPR
0,3,3: ADJSPR 0,4,6: LET C=7
307 IF SIZ=8 THEN  ADJSPR 0,2,2: ADJSPR
0,3,3: ADJSPR 0,4,4: LET C=7
308 IF SIZ=0 THEN  ADJSPR 0,2,0: ADJSPR
0,3,0: ADJSPR 0,4,0: LET C=7
310 FOR SP=2 TO 4: ADJSPR 1,SP,C: ADJSPR
2,SP,X+(SP-2)*8: NEXT SP
320 RETURN
900 CLS : CSR 10,3: PAPER 1: PRINT "        "
902 CSR 10,2: PRINT "        "
903 INK 4: PAPER 7: CSR 9,3: PRINT "i": CSR
22,3: PRINT "j": CSR 7,2: PRINT "imm": CSR
22,2: PRINT "mmj"
904 CSR 7,3: PAPER 1: INK 7: PRINT "i": CSR
24,3: PRINT "j"
906 CSR 3,3: PRINT "    ": CSR 25,3: PRINT "    "
909 FOR I=4 TO 11
910 CSR 1,I: PRINT "                          "
920 NEXT I
930 RETURN
1000 INK 5
1002 CSR 1,12: PAPER 4: PRINT
"eeeeeeeeeeeeeeeeeeeeeeeeeeeeeee"
1003 CSR 1,13: PRINT "jjjjjjjjjjjjjjjjjjjjjjjjjjjjjjj"
1004 FOR Q=14 TO 20
1005 CSR 1,Q: PRINT "    "
1006 CSR 23,Q: PRINT "    "
1007 NEXT
1012 CSR 4,21: PAPER 7: INK 4: PRINT
```

CPM MENU'S

A couple of months ago Graham Mitchell released a suit of CPM utility programs on diskette. He went about it in the correct professional manner. Paying for his advert in MOC. Sales were slow, zero in fact. In my opinion it was correct for the club to stand by his side and try and help. We have placed the advert since free of charge, reviewed the software and included it in our soon to be released mail shot.

Graham has just sent us a letter saying that by way of thanks he is dropping the price of the software from £15.00 plus vat to £12.00 plus vat.

On top of that he has also included another program on the disc called INFO.COM. This is what his literature has to say about it.

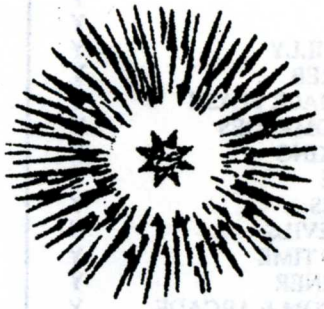
INFO.COM is a CP/M program written mainly to demonstrate programming in CP/M. It is similar in operation to DIR but gives each file a number so you can quickly see how many files you have on a disc and unlike dir it also displays system files.

You can use INFO with the normal combinations of * and ? wild cards. The program is provided as an ASCII file so that you can Assemble and Load it yourself. (Like with the SORT.ASM program elsewhere in the magazine)

Many thanks Graham for your efforts.

I feel that perhaps members should write in and tell us what you want if any in the way of programs. We would gratefully accept all comments in a constructive manner.

```
"hhhhhhhhhhmmmmmmhhhhhhhhhh"
1020 LET T=3: FOR I=16 TO 18
1030 CSR T,I: INK 2: PAPER 6: PRINT "j": CSR
31-T,I: PRINT "i": LET T=T+1: NEXT I
1040 CSR 1,16: INK 1: PRINT "  ": CSR 29,16:
PRINT "  ": CSR 1,17: PRINT "j": CSR 30,17:
PRINT "i"
1050 CSR 1,3: INK 7: PAPER 4: PRINT " i": CSR
29,3: PRINT "j ": CSR 1,2: PRINT "i": CSR 30,2:
PRINT "j"
1060 INK 15: PAPER 12: CSR 17,18: PRINT
"SCORE": CSR 17,15: PRINT " FUEL"
1100 INK 1: PAPER 7
1102 REM *** PANEL GRAPHICS ***
1106 LINE 134,50,178,50: LINE 134,51,178,51
1109 LINE 76,26,76,75: LINE 76,75,132,75: LINE
132,75,132,26: LINE 132,26,76,26
1110 LINE 73,24,73,78: LINE 73,78,182,78: LINE
182,78,182,24: LINE 182,24,73,24
1113 LINE 0,191,24,167: LINE 255,191,231,167
1114 LINE 40,184,56,168: LINE 215,184,199,168
1115 LINE 40,184,40,191: LINE 215,184,215,191
1120 CSR 2,18: PAPER 1: INK 5: PRINT "j":
CSR 29,18: PRINT "i": CSR 2,19: PRINT "j": CSR
29,19: PRINT "i": CSR 2,20: PRINT "j": CSR
29,20: PRINT "i"
1130 CSR 5,0: PAPER 7: PRINT
"hhhhhhhhhhhhhhhhhhhhhh"
1140 FOR J=1 TO 21 STEP 2
1145 INK 6
1160 LET DE=0: FOR J=167 TO 153 STEP -2
1170 LINE 8+DE,J+16,40+DE,J+16: LINE
247-DE,J+16,215-DE,J+16: LET DE=DE+2:
NEXT J
1200 FOR I=15 TO 19
1210 CSR 10,I: PAPER 3: PRINT "    ": NEXT I
1400 RETURN
4000 SOUND 1,PEEK(I)+100,10: SOUND
2,PEEK(I-MOD(I,3))+250,9
4001 PAPER (MOD(I,16)+1)
4002 CSR 10,9: PRINT "  DOG  FIGHT  "
4003 CSR 3,11: PRINT "BY T.J.S. GRAPHICS
BY A.G.M"
4005 INK 1: IF INKEY$="" THEN  LET I=I+1:
GOTO 4000
4006 PRINT "": SOUND 1,0,0: SOUND 2,0,0
4007 RETURN
5000 GENPAT 0,101,0,255,0,0,255,255,255,0
5010 GENPAT 0,104,0,0,90,255,189,231,195,195
5020 GENPAT 0,105,255,127,63,31,15,7,3,1
5030 GENPAT
0,106,255,254,252,248,240,224,192,128
5040 GENPAT 0,107,240,240,240,248,24,4,2,1
5050 GENPAT 0,108,15,15,15,31,24,32,64,128
5060 GENPAT
0,109,255,255,255,255,255,255,255,255
5070 GENPAT 3,0,0,0,0,0,0,0,0,0
5080 GENPAT 3,1,255,129,0,0,0,0,129,255
5090 GENPAT 3,2,3,15,63,255,255,63,15,3
5100 GENPAT
3,3,255,255,255,255,255,255,255,255
5110 GENPAT
3,4,224,240,252,255,255,252,240,224
5120 GENPAT 3,5,1,7,15,63,63,15,7,1
5130 GENPAT
3,6,128,224,240,252,252,240,224,128
5140 GENPAT 3,7,0,3,7,31,31,7,3,0
5150 GENPAT 3,8,0,192,224,248,248,224,192,0
5160 GENPAT 3,9,0,0,1,3,3,1,0,0
5170 GENPAT 3,10,0,126,255,255,255,255,126,0
5180 GENPAT 3,11,0,0,128,224,224,128,0,0
5190 GENPAT 3,12,0,0,60,255,255,60,0,0
5200 GENPAT 3,13,0,0,24,126,126,24,0,0
5210 GENPAT 3,14,0,0,0,60,60,0,0,0
5220 GENPAT 3,15,0,0,0,0,24,0,0,0
5230 GENPAT 3,16,0,0,0,0,24,0,0,0
5240 RETURN
60000 PRINT ASC(INKEY$): GOTO 60000
```

SOFTWARE PRICE LIST
JULY 1989

MOC Phil Eyres 13 COPSE ROAD, TOWNHILL PARK, SOUTHAMPTON.
Tel 0703 585106

Title	Stock
26*26 SPREADSHEET	Y
ADVENTURE QUEST	Y
3D TACHYON FIGHTER	Y
AGROVATOR	N
ALICE	Y
ASTROMILLON	Y
ASTROPAC	Y
ATTACK OF KER TOMS	Y
BACKGAMMON	Y
BASIC BUSINESS	Y
BLOBBO	Y
BOUNCING BILL	N
BRIDGE	Y
CAVES OF ORB	Y
CHAMBEROIDS	N
CHARMKATZ	Y
CHESS	Y
COLOSSAL ADVENTURE	Y
COMBAT	Y
CONT. RAIDERS	Y
CRYSTAL	Y
DENNIS AND THE CIRCUS	Y
DISASM	Y
DOODLEBUG	Y
DOWNSTREAM DANGER	Y
DRAUGHTS	Y
DR FRANKIE	Y
DRIVE THE CEE-5	Y
DUNGEON ADVENTURE	Y
EDASM	Y
EMERALD ISLE	Y
ESCAPE FROM ZARKOS	Y
EXTENDED BASIC	Y
F1 SIMULATOR	Y
FATHOMS DEEP	N
FIG FORTH £10.00	Y
FIG-FORTH MOC	Y
FIG FORTH SDX	Y
FIREHOUSE FREDDIE	N
FIRST LETTERS 1	Y
FLUMMOX	Y
GOLDMINE	Y
GRAPHICS	Y
HAWKWARS	Y
HELI-MATHS	Y
HIGHWAY ENCOUNTER	N
HUNCHY	Y
ICEBURG	Y
JET SET WILLY	Y
JOHNNY REB	Y
JUMPING JACK FLASH	Y
KILLER TOMATOES	Y
KARATE KING	Y
KILOPEDE	Y
KNUCKLES	Y
LITTLE DEVILS	N
LORDS OF TIME	Y
MANIC MINER	Y
MISSILE COM & ARCADE	Y
MATHS 1	Y
MAXIMA	Y
MEMOCHEQUE	Y
MEMOSKETCH	Y
MEMOSKETCH SDX	N
MINEFIELD	Y
MINER DICK	Y
MISSION ALPHATRON	Y
MISSION OMEGA	N
MTX UTILITIES	Y
MURDER AT THE MANOR	Y
MUSIC PAD	Y
NEMO	Y
OBLITERATION ZONE	Y
OBLOIDS	Y
PAINTBOX	Y
PAYROLL	Y
PHAID	Y
PHYSICS	Y
PONTOON & BLACKJACK	Y
POT HOLE PETE	Y
PUC-MAN	Y
PURCHASE LEDGER	Y
QOGO	Y
QOGO 2	Y
QUANTUM	Y
QUAZZIA	N
QUEST 1	Y
REVEAL	Y
REVERSI	Y
ROLLA BEARING	Y
RUTHLESS B.	Y
SALES LEDGER	N
SALTY SAM	Y
SEPULCRI SCELERATI	Y
SMG	Y
SMGII	Y
SNAPPO	Y
SNOWBALL	Y
T.SNOOKER	Y
SON OF PETE	Y
SOUL OF A ROBOT	Y
STAR COMMAND	Y
SUPA CODER	Y
SUPER BIKE	Y
SUPER MINEFIELD	Y
SURFACE SCANNER	N
TAPEWORM	Y
TARGET ZONE	Y
THE WALL	Y
TOADO	Y
THE DESIGNER	Y
THE KEY TO TIME	Y
THE ZOO	Y
TUMBLEDOWN TOWER	Y
TURBO	Y
USER BASIC	Y
USER BASIC SDX	Y
UTILITIES SDX	Y
VERNON AND THE VAMPIRES	Y
WORD & PICTURE	Y

IS THIS CHESS ??

Project for which a circuit needs to be designed with a parts list.

BACKGROUND

A.E. Hills has developed and fully tested a computer program for the display of chess games in progress, ie for audiences at chess tournaments. It is very fast being written in assembler but it has the drawback of relying on operator input of moves. There is a system of automatic input existing which has been used at prestige chess events, but only where heavy sponsorship is available, due presumably to the price asked. I have come up with a simple method of achieving automatic input, suggested by, of all things, a Musical Tea Mug. The two new ingredients which encourage me to press forward are, a plastic material which is practically opaque and an extremely sensitive light operated switch which can operate through it. The firm who make these have expressed willingness to let me have materials for trials and discuss price later. My knowledge of electronics is enough to specify what I wish to have but not enough to design the circuit.

WHAT IS REQUIRED OF THE CIRCUIT

The computer is the Memotech. This has a user port for Input and Output. Pin-out details are given. The port has been extended outside of the case by a 20 way ribbin cable and a header sits in a DIL socket. It is an easy exercise to make any combination of eight LEDs light up by sending a number between 0 and 255 to the port OUT but what I want to do is the opposite, ie on the instruction IN have the computer receive a byte value representing the active or inactive state of eight light sensitive devices corresponding to a rank on the chess board. (They will be active or inactive according to whether a chess piece is sitting over a short tube in which each one sits.) No encoding device is required, the computer does that.

What the computer manual says about the IO port is as follows; "Data may be latched in for reading with an active low pulse on the enable line, designated INSTB" and "This is an uncommitted TTL compatible ...available on an internal 20 pin DIL socket. The port is normally transparent but input data may be latched by taking INSTB to a logic low ...only TTL compatible signals may be used. The 5v current drain must not exceed 20mA"."

A signal for latching is envisaged as being given when the action of the player in pressing his clock (all serious games are played with a special design of two toggle on-off clocks) which in a quite simple manner would close the circuit. As there are 64 squares on a chess board, ie of the eight-bit ranks mentioned it is clear that eight bytes of information are required, from which my program will deduce what changes have occurred. Some form of octal counter will therefore be required to make the circuit take a reading of eight ranks of switches in turn.

The light sensitive switches come with a tiny battery which appears to be 1.5v. The drop across a switch appears to be in excess of point one of a volt. I wonder about having 64 tiny batteries any one of which may fail, whether the voltage is enough to switch on a 5v TTL signal, whether the switch is affected by having it on 5 volts, and given the limitation of current drain from the computer's supply, whether an independent and reliable means of providing 5 volts should be included. I do not envisage any wires having to travel far, as the

circuits will sit in a shallow box of the chess board, indistinguishable from any normal one, is the top, with the computer sitting on a ledge directly underneath.

There is a magazine article available which describes the I/O port as being a 74LS373 quad transparent latch as regards the eight inputs and a 74LS374 tri-state octal D-type flip-flop as regards output. I found this somewhat confusing as both these numbers are very similar-looking 20pin devices in the catalogue. The article gives more details of possible circuits for the expansion bus and for the printer port than for the I/O port but the printer port has only four input lines and the expansion bus seems to involve decoding which the I/O port can handle.

The action of the player in pressing the lever of his clock is the conclusion of a legal move, and is convenient to use as the moment for "taking a picture" by providing a latch pulse. However, his clock lever will stay down till his opponents comes down, and we only need the one set of eight readings. A very slight pause should be introduced so that the circuitry does not run ahead of the computer taking on board each successive byte value. We are talking about minute portions of time here, and no matter how fast the players move, as they do in a short of time finish, it would seem inlikely that they could ever beat the circuitry and a computer program in machine code.

There is a bewildering variety of devices in Maplins catalogue which might do all I want, but I do not know enough to choose and be sure.

Can you help??

If anyone is interested or has some ideas. Please sen them to:-

A.Hills 9 New Place Rd, Pulborough, West Sussex RH20 2JD. Tel 07982 3688

CP/M ROUTINE

By Dave Dulson

This is an interesting routine from some time ago when we were printing a lot of articles about sort routines. This is only a simple sort routine. as it stands it is only a demostration but it could obviously be used for real if it were made to point to some real data.

For those that are not into CPM assembler or if you wonder why you do not recognise the code, here are a few help lines.

1. This code is written in CPM 8080 code which uses different mnemonics from the Z80 which you may be used to.

2. The code should be written using an ASCII text editor. This could be the CPM editor ED.COM or more simply NEWWORD in non document mode.

3. This ASCII code should now be assembled using ASM.COM. Then it should be loaded using LOAD.COM. For more reference read chapter 9 of the CPM manual.

4. Even if you do not fully understand, have a go at making a running program.

```
;*********************************
;*                               *
;*    BUBBLE SORT PROGRAM        *
;*                               *
;*********************************
BDOS    EQU 5
ORG 100H
SORT    MVI E,0         ;clear exchange flag
        MVI D,LEN-1     ;length of table to sort
        LXI H,LIST      ;start of table
        LXI B,LIST+1    ;pointer to next value
NEXT    LDAX B          ;get first value
        CMP M           ;compare with previous
        JNC NOEXCH      ;leave if in order
        PUSH PSW        ;save second on list
        MOV A,M         ;get first number
        STAX B          ;move to next pos
        POP PSW         ;get second number
        MOV M,A         ;move to first pos
        INR E           ;make flag none zero
NOEXCH  INX H           ;move pointer
        INX B           ;up the list
        DCR D           ;check if at end of list
        JNZ NEXT        ;continue sorting if not
        MOV A,E         ;get exchange flag into A
        ORA A           ;set flags
        JNZ SORT        ;check for sort completed
        LXI D,DISP
        MVI C,9
        CALL BDOS
        JMP 0           ;RETURN TO CP/M
;list of numbers to sort , maximum 256 values
DISP    DB 27H
LIST    DB 5AH,59H,58H  ;sample list of values
        DB 53,51,35,65,48
        DB 66,49,67,85,86
        DB 54H,43H,27H,32H,56H
        DB 44H,52H,45H,51H,28H
        DB 46H,50H,47H,4FH
        DB 48H,4EH,49H,38H
        DB 4DH,4AH,4CH,4BH
LEN     EQU $-LIST      ;length of list to sort
        DS 36-LEN       ;save space for up to 256
        DB 24H
        DB 27H
        END 100H
```

Your Letters

Hints & Tips

If any member has a PANASONIC KX-P1081 (DMX80!!!) or similar printer and a black painted metal tear bar with hard to see column indents, just rub some TIPPEX correction fluid into the indents and wipe off the surplus a bit at a time with a damp cloth and you should have easy readable markings.

Paul Trainer Leeds.

Questions

Q. I believe that once upon a time Memotech had 'persuaded' a MTX512 to read Spectrum games tapes. Is this true? Also, did they do this for the Einstein and MSX machines?

A. You are indeed correct, Memotech did produce a hardware add-on Spectrum emulator ROM for the MTX. It included a cassette with software to load 20 Spectrum commercial games. There were promises of future tapes for more games, but sales were never high enough to encourage this to happen. I have tried one out (I think we reviewed it some years ago) and it worked ok.

Since then a better tape only solution has been marketed. It was called the Z-Loader. This in effect did the same as the emulator but only in software, and was quite a bit better. We have the tape in our commercial software library for a fiver. It might be worth it to have a play with if you know Spectrum at all.

It is true MSX is very similar so is Einstein, the problem is it takes a lot of effort to create these things and sales are usually very low. It puts people off a bit.

Q. Is it possible to use any of the memory expansion boards 'battery backed' so that the memory contents are not lost on power down?

How can the expansion be accessed? Can there be different page sizes?

Grant McKenzie.

A. This is not really an answer, just what I know. I have seen battery backups on silicon disc drives before. But not on memory expansion boards, although I see no reason why it could not be done.

The memory expansion boards all work on the principle of paging in and out 16K blocks of memory. The ROMs and the top 16K are not normally paged just the 32K in the middle.

Answers

As far as I've found the problems Mr Seekings is having of display corruption can't be stopped, even with the disc unit bolted to the main unit as suggested by MCL. It tends to be worst whilst everything is warming up. After 1/2hr or so things tend to stabilise. A small fan into the back of the disc unit and across the computer helps as well.

Andrew Fox. Leeds.

Titbits If anyone has an FDX system (or any other fan cooled system) make sure you clean out the sponge filter. If it gets clogged, you get reduced cooling and especially this time of year the machine will overheat.

If you MTX,SDX,FDX etc plays up. It could just be due to a bad contact. The numerous boards that fit on to the MTX are notorious for bad connections. It's worth trying to cure the problems by removing the boards and then remaking the connections.

Phil Eyres ---

Second Hand Hardware For Sale

Newword ROM Board

Fully Populated, ready to plug in and go. Including manual.

£25.00 + £1.00 P&P

RS232 Board

Fulling populated with chips including fly leads. £25.00 + £1.00 P&P

32K Memory Expansion

(Not working - PROM CHIP BLOWN) Useful for extra memory chips! £10.00 + £2.00 P&P

TAXAN KP-810 PRINTER

1 off ONLY

As New

140 Characters per second, DRAFT/NLQ, Tractor or Friction Feed, Manual, Graphics Dumping. Includes 4ft printer connecting cable.

£125.00 + £2.50 P&P

Shinwa Cp 80

Good working order. Includes 2 ink ribbons. Tractor Feed. Centronics.

£75.00 + £3.00 P&P

Memotech DMX 80

Good working order. Includes ink ribbon,Tractor Feed, Manual and connecting cable. Centronics.

£75.00 + £3.00 P&P

MTX500 Motherboard

This board is fully populated, but is missing the Video Processor board. It has Norway ROMS fitted and the major chips are socketed.

This is the last board. One of the others has been proved to be working as is now being used!! £15.00 + £2.00 P&P

MTX500 Motherboards

These boards have most of the IC's, but are missing the Video Processor board and a few support devices. They have Norway ROMS fitted and the major chips are socketed. They should be ideal for spares or for more memory chips! £10.00 + £2.00 P&P

MTX Dust Covers

Two secondhand dust covers for your MTX 500 or 512. A must for keeping that keyboard clean.

£3.00 + £0.50 P&P Each

Microvitec Colour 14" Monitor

A superb monitor for use with the MTX and SDX systems. Costs over £250.00 new

£100.00 + £3.50 P&P

SDX 80 Column Boards

This board takens you must of the way to a cpm system, all you need are the upgrade ROMs for your system.

You then plug in a monitor and your up and away. CPM, Newword & Supercalc provided.

These boards will require that you make a lead to connect the board to a monitor. We will provide the necessary diagram.

£50.00 + £2.00 P&P Each (We have two!!)

1 Meg SDX Controller

One SDX controller, that connects on the left hand bus connector. Ideal for use with one of the above 80 column cards above. The only extras you should need are a 1meg disc drive and a 5volt power source. £50.00 + £2.00 P&P

RS232 BOARD

For MTX500 complete with fly leads and instruction book.

£25.00 + £2.00 P&P

64K MEMORY EXPANSION BOARD

For MTX500 complete with instructions.

£35.00 + £2.00 P&P

MTX POWER SUPPLY

In good working order.

£18.00 + £2.50 P&P

RS232 TERMINAL

In working order. 12" paper white display with qwerty keyboard preset baud rates from 75 to 19200. At present set up as a remote terminal for MTX CPM system.

£25.00 O.N.O. Buyer Collects.

Please make cheques payable to Memotech Owners Club. Address all correspondance to 13 Copse Rd, Townhill Park, Southampton. SO2 2GY.

8" DRIVES

THE SEQUEL

May I first thank Stephen Atty and Johnny Steedman for sending in copies of information which helped with the interfacing of the 8" disc drive.

Well, a couple of Saturdays ago I got up in the morning with a burning enthusiasm to do something. I thought I had better not let this rare occurance go to waste. So, I opened up my FDX unit and the 8" disc drive. A close inspection showed that the disc controller card had an unused 50 connector which looked very similar to the one on the back of the 8" drive. The documentation says that the controller will support up to 4 drives so all should be ok. I obtained from Maplins a metre of cable and two 50 way edge connectors. Taking care to keep the pin 1 of each connector in the right place I made the cable and connected to the units. I fired everything up and at least it all seemed to work. There were four possible configurations for the drive as far as I could see. Having tried the first three with no success, the drive just 'clicking' to my request for life. I had one last chance, the best chance, as this was the highest disc capacity. It worked, a quick DIR and there was loads of files. The drive is in fact a 980K formatted capacity drive, which works about 2.5 times faster than the 500k 5.25" drives.

I wonder does anyone else have one of these beasts?

Phil Eyres.

GAP FILLER 1

The following program puts a copy of the Sprite Table in memory

```
DI
LD A,0
OUT(2),A
NOP
LD A,63        ;START OF TABLE IN VRAM
(3F00)
OUT(2),A
LD B,128       ;LENGTH OF TABLE
LD C,1         ;PORT NO
LD HL,#A000    ;COPY OF TABLE AT 40960
INBYTE: NOP
NOP
NOP
INI            ;IN AND STORE DEC B INC HL
JR NZ,INBYTE   ;IF B0 THEN GOT INBYTE
EI
RET
```

There are 32 Sprites each 4 bytes long one after the other in the table. Each sprite is set out like so:

byte (1) Distance from top of screen (2) Distance from left of screen (3) Pattern no (4) Colour

Sprite one is at PEEK (40960)-(40963)
two is at PEEK (40960)-(40967)

John Raybould

GAP FILLER 2

GAMES BOARD

This routine draws the games board used in chess and draughts.

```
100 GENPAT
 0,33,255,255,255,255,255,255,255,255: CRVS
2,1,8,4,16,16,32: VS 4: PAPER 3: CLS: VS 2:
PAPER 15: CLS: PAPER 15

160 VS 2:INK 1: FOR V=2 TO 14 STEP 4:
FOR H=0 TO 12 STEP 4: FOR J=0 TO 1: CSR
H,V+J: PRINT "!!";: CSR H+2,V+J-2: PRINT
"!!";: NEXT: NEXT: NEXT
```

VARIABLES USED

V Vertical position of cursor ....

H Horzontal position of cursor

J Display two rows of characters "!!"

PROGRAM LIBRARY

By Phil Eyres

We now have all of the tapes, discs and library documents available from the library. Paul Wood has had to drop doing the 3.5" discs because of other work commitments which does leave us with a bit of a problem in that area. Is anyone interested in taking up the 3.5" disc copying?? Only problem is really it needs to be someone with a 3.5" drive joined to their 5.25" system.

Firstly, two additions to the Library Documents.

LL14 Mouse Interface

This is the full set of documents written by Mike Frymyer for the building and programming of the mouse interface unit. This document has been printed in several parts in the magazine.

LL15 Forth Sorting Routines

This is a set of documents sent in by Peter Burns that describes and shows by program three types of sort. The Bubble, Shell and Shuttle sort.

It lists 13 Forth screens to type in. Really useful if you are new to Forth.

PROBLEMS

I have had the odd problem with my extended CP/M disc (cpm11), it does not have a replacement for the REName command, although this was mentioned in the January magazine. (A minor irritant, I will generally only want to rename text files and can use Newword); ERA.COM refuses to let me use it, telling me I am not qualified - - this is a nuisance as it obstructs moving files around between user areas. Those variations aside I would recommend the EZCPR system to anyone who has CP/M: your short plug in the February issue does not really do it justice.

I found the "In the Public Domain" article interesting, more info on what is on the utility discs would be helpful, also the space taken by files in each set when sending the discs. Given my suspision above of being a file short I would have found a document file listing the contents, with perhaps a few words on each, useful; if such existed on each disc they could be strung together and appear on every disk to encourage people to buy more.

Harry Seekings, Kew.

The copying fees for disc software are £2.75 on disc for 20 programs (5.25" only). Please add £1 if we are to supply the disc.

Software on disc's 1 to 6 are available on cassette, two programs per cassette, for £1.50, we supply the cassette.

When ordering disc's please state exactly the system you have.

All cheques/postal orders payable to Memotech Owners Club please.

DO NOT SEND FURTHER PUBLIC DOMAIN LIBRARY CORRESPONDANCE TO ALAN. ADDRESS IT TO PHIL EYRES.

Software Top 10

These are the 10 best selling items for June 1989

1. Advanced Graphic Designer

2. Chess

3. Memosketch

4. Basic Tutorial

5. Murder At Manor

6. MTX Utilities

7. Snappo

8. Knuckles

9. Jumping Jack Flash

10. Qogo 2

Hardware & Software

We are on the look out for any secondhand hardware and/or software at reasonable prices. Especially Disc systems and printers. (15% Commision for selling hardware for members). Contact Phil Eyres on 0703 585106.

Disc 1

01. Hex/Dec/Bin
02. CGEN
03 D-Draw
04. Whist
05. Mem-Save
06. MTX-Draw
07. LOGO-Draw
08. Simplex Tableau
09. Breakeven
10. Statistics
11. Unsolved Problem
12. Radio Routines
13. Light Cycles
14. Hex/Dec/Bin
15. Character Ed.
16. Quasimodo
17. Planner
18. Hanoi
19. Noble
20. Hi-Lo

Disc 2

21. Composer
22. Anova
23. Cashflow
24. Reversi
25. Full Time
26. Panel 3
27. Word Pro
28. SwMice
29. TNT TIM
30. Sw3D Funct1
31. Sw3D Funct2
32. SwSpEd
33. SwMathe
34. OXO
35. Solitaire
36. Cross Numbers
37. Avoid Seven
38. Numerology
39. Chemin
40. Dice

Disc 3

41. Reversi Vers 2
42. ISOT
43. DBase
44. Diary
45. Terminal
46. Skittles
47. Card-Ind
48. 2 * H & W
49. Hangman
50. Account
51. Mastermind
52. Connect 4
53. Jrny Into Dngr
54. Connect 4 V2
55. Patience
56. Life
57. Enigma
58. FKEY
59. SkyDiver
60. Dice

Disc 4

61. MPG
62. Spooler
63. Labels
64. Ski Version 2
65. PNT/BJCK
66. Biorythms
67. Perpetual Calender
68. Flitter
69. Stock Control
70. Fonts
71. Dune
72. Headliner
73. Morse Tutor
74. Building Societies
75. Measurements
76. Clock
77. Clock 80
78. 3D Maze
79. Graphics Calc
80. FastGraf

Disc 5

81. Orbiter
82. Card Index
83. Appointments
84. Phonebill
85. Calender
86. Bouncy Ball
87. Hiscroll
88. FastWorm
89. Morse Sound
90. Sound Editor
91. Cass. Inlays
92. Printerdraw
93. StarTrek80
94. Graph Calc 2
95. Blaster
96. Demolition
97. Fonts V1.2
98. Digi Clock 1
99. Digi Clock 2
100. Print Noddy

DISC 6

101. CHRGEN
102. Space Patrol

Cassette Only

CA01 Renum III
CA02 Merge
CA03 Money Manager
CA04 FKEY
CA05 DBASE III
CA06 Filetech
CA07 Flight
CA08 RAMDisc
CA09 TextEd
CA10 Deci_Clock
CA11 Elements
CA12 MkBook
CA13 Optics
CA14 Dbase IV
CA15 Filetech II
CA16 Forth Extensions
CA17 Inprinter
CA18 Renum IV
CA19 New for Old
CA20 Dumpliner
CA21 3D Maze 512
CA22 Fast Graf 512
CA23 Renumber V

CP/M Software
(CP/M disc systems ONLY)

CPM01 Z80 Assembler
CPM02 EBASIC programs
CPM03 Mail Label
CPM04 Turbo programs
CPM05 Comms disc
CPM06 Small C Compiler
CPM07 Utilities Disc 1
CPM08 Prolog/Valgol
CPM09 Utilities Disc 2
CPM10 Utilities Disc 3
CPM11 Extended CP/M
CPM12 Forth
CPM13 Adventure
CPM14 8080/Z80 Translator
CPM15 ZBASIC
CPM16 Car Maintenance
CPM17 Multitasker
CPM18 Utilities Disc 4
CPM19 Stock System
CPM20 dBase progs & F.Simulator
CPM21 Utilities 5

Library Documents

Available at 50p per document, this includes postage and packing.
LL01 System Variables
LL02 VDP Chip Explained
LL03 NewWord ROM Review
LL04 RST10 Explained
LL05 Undocumented NewWord
LL06 CP/M Programming Course
LL07 Instructions For CA16
LL08 Pascal course
LL09 CP/M Introduction
LL10 Flitter
LL11 Enhanced Supercalc
LL12 Inprinter Instructions
LL13 Hitch Hackers Guide
LL14 Mouse Interface
LL15 FORTH Sorting Routines

Compilation disc

Mixed bag of programs of all sorts - £1 (excluding disc)

NON-MASKABLE INTERRUPTS

By Mike Frymyer

In the last section a description of the event sequence of the NMI was listed and there was a mention of a couple of uses for it. In this section I'll look at the first application, that of a backup supply for the computer...

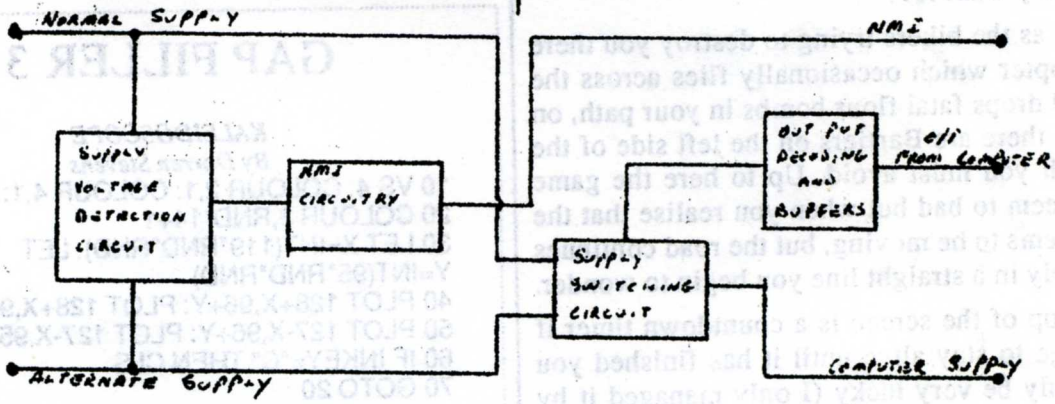
Isn't it utterly and absolutely mind blowingly aggrovating when your in the middle of an important piece of work and just before you hit the save button...

CRASH!!!

Now, I'm not advocating that everybody race out and buy a battery bank to run their computers, just setting up this section.

The block diagram shows the major elements in incorporating a backup supply in the computer system.

The detector monitors the supply and when it falls below a pre- determined amount, (higher than the main supply regulators), a signal is sent to the interrupt logic element.



Block Diagram of an Interupt Controlled Alternate Supply.

This section sends the NMI signal to the computer.

Now, it's up to the computer to protect itself from the loss of supply. (With the interrupt service routine).

A possible service routine might look like this:-

```
#0066 JP SERVICE
SERVICE: LD A,CODE
         OUT (PORT),A
         RST 10
         CALL SAVE
         CALL EXIT
         RETN
CODE:    DB #AA
PORT:    DB #07
```

CODE will be the right data for the hardware to recognise and act upon.

PORT is the number of the I/O port the peripheral is pugged into.

The message in this case is displayed by the RST 10 code and would be different under CPM.

Obviously the save and exit routines would be different under each MTX OS.

More Specifics to Follow....

REVIEWS

SUPERBIKE
Reviewed By
Andy Owen
Cost £5.00 from MOC

Released by Syntaxsoft a few years ago it is now available from MOC.

Loading takes a few minutes, most of which are taken up drawing the magnificent loading screen which promises much of the program. Unfortunately the game does not quite live up to its promise. There are no instructions so upon loading it is straight into the game. This consists of riding down an endless and seemingly straight road, controlling your bike using the cursor keys or right joystick.

Unfortunately it seems that you are not the only one travelling on this road to nowhere, Kamikaze Bikers rush at you and more often than not (to start with anyway) crash into you destroying your bike and laying you flat out on the ground. Soon you are up and ready to start all over again, the question is, do you really want to??

As well as the bikers trying to destroy you there is a helicopter which occasionally flies across the screen and drops fatal flour bombs in your path, on top of this there are Barriers on the left side of the road which you must avoid. Up to here the game does not seem to bad but when you realise that the scenery seems to be moving, but the road continues interminably in a straight line you begin to wonder.

At the top of the screen is a countdown timer if you manage to stay alive until it has finished you will not only be very lucky (I only managed it by going so slowly that no biker had a chance to hit me) but you explode and are told 'No time left' then you are asked 'Another Go?'.

As you look closer at the game you realise how dull all this really is: travelling down a straight road etc.. etc... and never seeming to come to an end. And you start to look for redeeming features, maybe the graphics are good; well the background graphics are the best bit, and they are only average, the Bikers being monochrome silhouettes moving rather jerkily towards you, the barriers white with black arrows on and also moving jerkily and the helicopter is as expected moving relatively smoothly; the road is pitch black with lines in the centre, which are sometimes the only evidence of movement, aas there are no roadside graphics. Your bike is exactly the same as the others but two tone for ease of identification, and as it only has to move from side to side, it moves smoothly. Then you listen - maybe the sound is decent, well it is, but it is representitive rather than realistic.

SUMMARY

I hate to put this program down so, as it is obvious that quite a lot of time has gone into it. Unfortunately the author seems to have stopped short of writing a really good game, as it could have been with a little more variety and smoothness. As it stands I would not recommend it unless you are either a motor bike game fanatic or you love boredom.

VFM 50%

GRAPHICS 60%

SOUND 60%

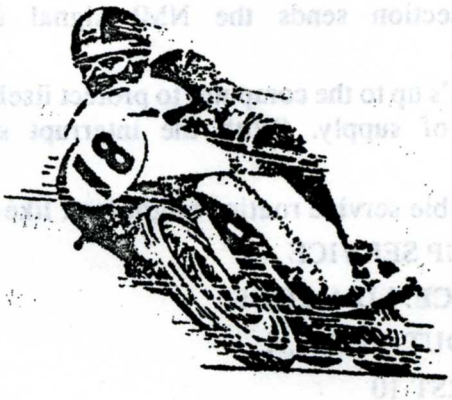
ADDICTIVENESS 49%

OVERALL 51%

GAP FILLER 3

KALEIDOSCOPE
By Darren Stevens

```
10 VS 4: COLOUR 2,1: COLOUR 4,1: CLS
20 COLOUR 3,RND*14+1
30 LET X=INT(119*RND*RND): LET
Y=INT(95*RND*RND)
40 PLOT 128+X,96+Y: PLOT 128+X,95-Y
50 PLOT 127-X,96+Y: PLOT 127-X,95-Y
60 IF INKEY="C" THEN CLS
70 GOTO 20
```



MERGING

The recent articles by Phil Eyres have shown that any sort routine written in BASIC will be slow and some will be very slow. We should, therefore, try to cut the number of records sorted to the minimum or, if we must sort a large file, reduce the number of records we actually move in the process.

Fortunately, one of the most common sorting tasks met with in practice is updating a file held on disk or tape where we have a large number of records already in sequence to which we want to add a small number of records in their proper places. This could be done by reading the existing file, adding the new records, sorting the lot and writing them all back to the file, but it can be speeded up by using the following technique:

1) Read the existing data file into an array.

2) Input the new data into a separate array and sort this new data only.

3) Compare the first records on each array and write the lower of them to the output file, then compare the next record on that array with the lowest record on the other array and so on until all the records have been written to the output file in sequence.

This results in a completely sorted file at the expense of sorting only the new records. The file could also be processed, e.g. printed, while it is being written out.

Suppose we have a need to add a few records to a file but we need to access them in sequence rather than write them staight out to a file. This can be tackled in a different way:

Read the records into an array but add to the end of each record a field containing the number of the next field in the array e.g. the first record has "2" added, the second has "3", etc. We can now go through the array in the sequence in which we read the file in by taking the number we have written on to each record and using it as the subscript of the next record, e.g. the record in ARRAY(1) has had "2" added to it and ARRAY(2) is the next record in sequence. This may not seem to give us an advantage but, supposing we add a new record to the file, putting it in the first empty space which happens to be ARRAY(400) al- though a quick check shows us that it should really go between records 50 and 51. All we need to do is alter the address on record 50 from "51" to "400" and the address on record 400 to "51" and we can read the file in sequence - and write it in se- quence to an output file if we wish - without having to shuffle the whole file up and down. If several records are to be added to the file they should obviously be sorted to reduce the amount of searching needed to find their correct sequences.

There are additional benefits to this technique: we can for example hold a file on disk in one sequence but have addresses on the end of each record so that it can be read in a second (or n different) sequence(s) without further sorting: this could be used in a file of club members where we want to produce regular prints in both surname and membership number sequences. If the file is held as a random access file on disk the addresses can also be used as record keys to allow the file to be read in se- quence without having to be sorted each time a new record is added.

Sometimes we need to sort the whole of a file into a new sequence. This can only be done by reading the whole file into memory and sorting it. If, however, the file is too big to be held in memory at one time we must resort to the following tech- nique:

Suppose we are going to sort a file of 1000 records which are so large that only 200 can be held in memory at one time: we read the first 200 records, sort them into the new sequence then write them to a file which we may call STRING1. We then read the next 200, sort them and write them to STRING2 and continue like this until the whole of the input file has been sorted into strings (in this case 5 strings). the next step is to read the first 40 records from each string into memory and start merging them into the output file until all the forty records from one string have been written away. The next forty records from this string can then be read into the same array and the process continued, topping up each array from its string until all the strings have been merged into the output file.

Even if a file held on disk has random access, normally the only way to find a particular record is to guess at a record number and then read through the file from this point checking each record to see if it is the one you want. With a large file this can be very tedious, but you may find it speeds things up if you write a record at the front of the file

containing an index with the record key (or the first few characters of the record key) and the record number of each, say, 50th record. If this record is read in when the program is loaded it will enable you to make a good guess where to start searching, using the next record keys to steer you through the file. n.b. make sure you start before the record you want. Unfortunately it only works with random files which are always accessed on the same key, e.g. a file which is always accessed by surname.

BANNER HEADING PROGRAM

This little assembler program will scroll chunky sized characters across the screen in MTX colour you like. Just read the BASIC lines at the end of the program for details.

The code is well documented so I will leave it alone.

```
 3 GOTO 10
 4 VS 4: CLS
 5 CODE
 4018       LD HL,0
 401B       LD (LENGTH),HL
 401E       LD (NOVRS),HL
 4021       LD (NOSTR),HL
 4024       LD HL,(#FA7B);VARIABLES..
 4027 LOPI:  LD A,(HL);VARIABLE NAME
 4028       CP #FF;AT THE END
 402A       RET Z;NO M$!
 402B       PUSH HL
 402C       CP #7F;END CHR?
 402E       JR C,LGNAME
 4030 LOPII: LD HL,(NOVRS)
 4033       INC HL;INC TOTAL
 4034       LD (NOVRS),HL
 4037       CP #C0;=NUMERIC?
 4039       JR NC,NVAR;THEN JUMP.
 403B       LD HL,(NOSTR)
 403E       INC HL;MUST BE A STRING
 403F       LD (NOSTR),HL
 4042       CP #8D;IS IT M$?
 4044       JR Z,MESSG;YES-GOTO MAIN PRG
 4046 NVAR:  POP HL;NUMERIC VARS
 4047       INC HL;NEXT ONE
 4048       JR LOPI;LOOP BACK
 404A LGNAME: POP HL;COME HERE IF
DEALING
 404B       INC HL;WITH A NAME OF
 404C       PUSH HL;MORE THAN 1 CHR
 404D       LD A,(HL)
 404E       CP #7F;END YET?
 4050       JR C,LGNAME
 4052       AND #F1;DONT WANT MA$ ETC.
 4054       JR LOPII;LOOP BACK
 4056 MESSG:  POP HL;FOUND M$
 4057       LD HL,(#FA7F);CALC STACK
 405A       LD DE,(NOVRS)
 405E       LD B,5;EACH VAR IS 5 BYTES LONG
 4060 MSLP1:  OR A;CLEAR CARRY
 4061       SBC HL,DE;SUBTRACT
 4063       DJNZ MSLP1
 4065       INC HL;MOVE FORWARD
 4066       INC HL;TO FIND THE
 4067       INC HL;STRINGS LENGTH
 4068       RST 8;INTO DE
 4069       LD (LENGTH),DE
 406D       LD BC,(NOSTR)
 4071       LD HL,(#FAA7);BASIC PROGRAM
END
 4074 MSLP2:  RST 8;-WHERE STRINGS ARE
 4075       DEC BC;DEC COUNTER
 4076       LD A,B
 4077       OR C
 4078       JR Z,MAINET;THIS IS IT.
 407A       ADD HL,DE;MOVE TO NEXT
 407B       JP MSLP2;AND GO BACK
 407E MAINET: LD DE,(LENGTH)
 4082       PUSH IX;IX POINTS TO COLOUR
 4084       LD IX,#FA8C
 4088       LD A,(IX+00)
 408B       AND #0F;MASK CODE
 408D       CP 2;BLACK IS NOT ALLOWED
 408F       JR NC,CVAL
 4091       LD A,#0F;WHITE
 4093 CVAL:  LD B,A;STORE
 4094       OR A;CLR CARRY
 4095       RLCA;MOVE CODE INTO
 4096       RLCA;HIGHEST 4 BITS
 4097       RLCA;OF A
 4098       RLCA
 4099       OR B;FULL COLOUR CODE
 409A       LD (IX+00),A;STORE
 409D       PUSH HL
 409E       PUSH DE
 409F       LD HL,SDAT;INFO FOR VDP
 40A2       LD C,2;OPERATING MODE
 40A4       LD B,3;3 REGISTERS
 40A6 SLP:    RST 8;LD DE,(HL)
 40A7       OUT (C),D;VALUE..
 40A9       OUT (C),E;TO REGISTER
 40AB       DJNZ SLP;BACK AGAIN
 40AD       LD HL,15360;NAME GEN ADDR
 40B0       CALL VWAD;SEND IT
```

```
40B3      LD BC,767;SCREEN LENGTH
40B6 SLP1:  LD A,#FF;DUMMY CHR
40B8      OUT (1),A;TO VDP
40BA      DEC BC;DEC POINTER
40BB      LD A,B;=0?
40BC      OR C
40BD      JR NZ,SLP1;BACK AGAIN
40BF      LD HL,DTABLE;TEMP SCREEN
40C2      LD DE,0;COUNTER
40C5      LD BC,#100;256 BYTES
40C8 SLP2:  PUSH DE;SAVE POINTERS
40C9      PUSH HL;FOR A WHILE
40CA      LD A,#1F;MASK FOR COUNTER
40CC      AND E;ADJUST E TO
40CD      LD L,A;GIVE 0-31
40CE      RR D
40D0      RR E;AND 32-63
40D2      RR D
40D4      RR E;BOTH 4 TIMES
40D6      LD A,#E0;ANOTHER MASK
40D8      AND E;GET BITS
40D9      OR L;GEN FINAL VAL
40DA      POP HL;ADDRESS
40DB      LD (HL),A;LOAD IT
40DC      INC HL;NEXT ADDR
40DD      POP DE;RESTORE PTR
40DE      DEC BC;DEC COUNTER
40DF      INC DE
40E0      LD A,B;FINISED YET?
40E1      OR C
40E2      JR NZ,SLP2;NO-GO BACK
40E4      POP DE;RESTORE
40E5      POP HL;POINTERS
40E6      DI;LEAVE ME ALONE FOR A
WHILE.....
40E7 MAINLP: LD A,D;DONE ALL THE
40E8      OR E;CHRS IN STRING
40E9      JP Z,EXIT;YET?
40EC      LD A,(HL);GET CHR..
40ED      INC HL;AND UPDATE
40EE      DEC DE;POINTERS
40EF      PUSH HL;STORE THEM
40F0      PUSH DE;FOR NOW
40F1      LD H,3;BUILD SCN ADDR
40F3      AND #7F;UDGS ARE 0-31
40F5      LD B,3;COUNTER @PROGRAM =
40F7      OR A
40F8 MAIN1:  RLA;BIT INTO CARRY
40F9      RL H;INTO H
40FB      DJNZ MAIN1;NEXT PASS
40FD      LD L,A;HL = VDP ADDR
40FE      CALL VRAD;FOR CHR IN
4101      LD B,8;RAM, 8 BYTES.
4103      LD DE,MDAT;TEMP STORAGE
4106      LD A,#F8;MASK FOR 6 BITS-IF
USING 8 BIT CHR SET CHANGE TO #FE
4108      LD (DE),A;TRANSFER
4109      INC DE
410A MAIN2:  CALL VRD
410D      LD (DE),A
410E      INC DE
410F      DJNZ MAIN2
4111 MVLP:  CALL DSCRL;SCROLL SCREEN
MAP
4114      LD C,A;FLAG
4115      CALL WSCN;WAIT FOR BLANKING
FRAME
4118      LD A,C;RECALL FLAG
4119      CALL DISPSC
411C      CALL BDISP;DISPLAY NEXT BIT
411F      JR C,MVLP;GO BACK IF CHR
UNFINISHED
4121      POP DE;RECALL POINTERS
4122      POP HL
4123      JP MAINLP;NEXT CHR
4126 BDISP:  LD A,(SCPOS);POSITION
4129      RLCA;ADJUST IT TO
412A      RLCA;GIVE SCREEN
412B      RLCA;ADDRESS
412C      AND #F8;AND THEN
412E      LD L,A;SEND IT TO
412F      LD H,0;THE VDP
4131      CALL VWAD
4134      LD DE,MDAT;THE CHR
4137      LD A,(DE);MASK
4138      LD C,A;IN C
4139      LD B,4;4 BYTES NOW
413B BLP1:  INC DE;INC POINTER
413C      LD A,(DE);THE BYTE
413D      RLCA;BIT IS IN C
413E      LD (DE),A;BYTE GOES BACK
413F      SBC A,A;0-#00 1-#FF
4140      AND (IX+00);COLOUR BYTE
4143      CALL VRT;TO SCREEN
4146      CALL VRT;TWICE
4149      DJNZ BLP1;NEXT BYTE
414B      LD B,4;LAST 4 BYTES
414D      INC H;GO LOWER DOWN
414E      CALL VWAD;THE SCREEN
4151 BLP2:  INC DE;LOOP AS ABOVE
4152      LD A,(DE)
4153      RLCA
4154      LD (DE),A
4155      SBC A,A
4156      AND (IX+00)
4159      CALL VRT
415C      CALL VRT
415F      DJNZ BLP2
4161      LD A,C;THE FLAG
4162      RLCA;C=0 IF FINISHED CHR
4163      LD (MDAT),A
4166      RET
4167 DISPSC: AND #1F;MASK A
4169      LD (SCPOS),A;AND STORE
416C      LD HL,15616;SCREEN ADDR
416F      CALL VWAD
4172      LD HL,DTABLE;TEMP SCREEN
STORAGE
4175      LD BC,#0100;256 BYTES
4178 DSLP:  LD A,(HL);FETCH BYTE
4179      OUT (1),A;SEND
417B      INC HL;NEXT
417C      DEC BC;DEC POINTER
417D      LD A,B;=0?
417E      OR C
417F      JR NZ,DSLP;GO BACK IF NOT
```

```
4181        RET;FINISHED
4182 DSCRL:  LD HL,DTABLE
4185        LD DE,DTABLE
4188        LD A,(HL);FIRST BYTE
4189        CALL DMVL;MOVE DATA
418C        LD HL,DTABLE1
418F        LD DE,DTABLE1
8192        LD A,(HL);FIRST BYTE
4193 DMVL:  INC HL;SET UP B.MOVE
4194        LD BC,#007F;127 BYTES
4197        LDIR;GO
4199        LD (DE),A;FIRST BYTE-LAST
POSITION
419A        RET;FINISHED
419B WSCN:  IN A,(2);VDP STS.
419D        RLCA;SCREEN BLANK?
419E        JR NC,WSCN;BACK IF NOT
41A0        RET;SCREEN BLANK NOW
41A1 EXIT:  EI;OK I'M FINISHED.
41A2        POP IX
41A4        RET;BACK TO BASIC
41A5 VWAD:  PUSH AF;STORE AF
41A6        LD A,L;LOW BITS
41A7        OUT (02),A;TO VDP
41A9        LD A,H;HIGH BITS
41AA        OR #40;SET BIT 6
41AC        OUT (02),A;TO VDP
41AE        POP AF;RESTORE AF
41AF        RET;DATA SENT
41B0 VRAD:  PUSH AF;STORE AF
41B1        LD A,L;LOW BITS
41B2        OUT (2),A;TO VDP
41B4        LD A,H;HIGH BITS
41B5        AND #3F;BITS 6,7 =0
41B7        OUT (2),A;TO VDP
41B9        POP AF;RESTORE AF
41BA        RET;DATA SENT
41BB VRD:   IN A,(1);READ BYTE
41BD        RET
41BE VRT:   OUT (01),A;WRITE BYTE
41C0        RET
41C1 DTABLE: DS 128;TEMP SCREEN
4241 DTABLE1:DS 128;MAP STORED HERE
42C1 SDAT:  DB #80,#00;REG 0=#00
42C3        DB #81,#E8;REG 1=#E8
42C5        DB #84,#00;REG 4=#00
42C7 SCPOS: DB 0;SCREEN COLUMN TO
WRITE TO NEXT
42C8 MDAT:  DW 0,0,0,0,0
42D2 LENGTH: DW 0;STRING LENGTH
42D4 NOVRS: DW 0;TOTAL NO. VARS
42D6 NOSTR: DW 0;NO OF STRINGS
42D8        RET
```

```
8 RETURN
9 REM ** ADD 10 OR 11 SPACES TO SCROLL
MESSAGE OFF SCREEN **
10 LET M$="STORE MESSAGE IN M$ AND
USE 'GOSUB 4' TO EXECUTE THE PROGRAM.
"
20 GOSUB 4
30 POKE 64140,9
40 LET M$=" POKE 64140 WITH THE COLOUR
CODE FOR DIFFERNT COLOURED LETTERS..
THIS IS COLOUR 9     "
50 GOSUB 4
```