

# Computing *with the* AMSTRAD

The independent magazine for Amstrad computer users  
Registered by Australia Post — Publication No. VBP7811

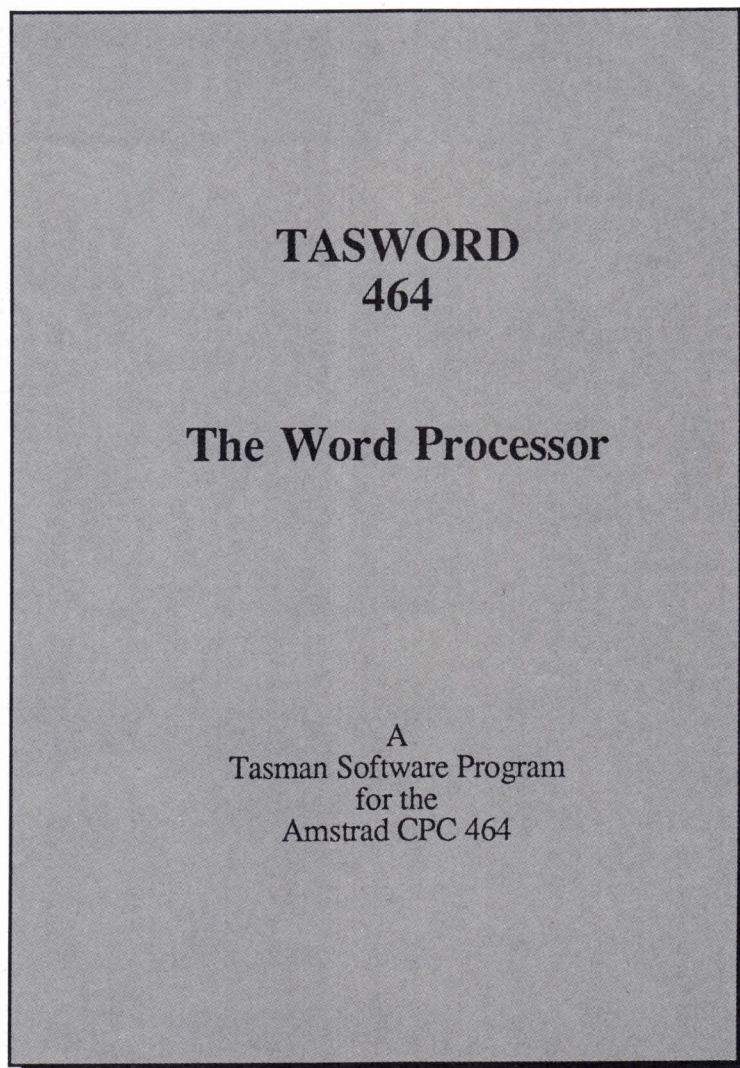
## Unveiled: Amstrad's Super PC



Two full length games for you to key in and play Pilot language — a full listing, Airo Review, Scrolling and Sprite utilities plus a Character Generator as well as ten more software reviews and all the regular columns.

**TASWORD 464 - THE WORD PROCESSOR**

**WE HAVE A LIMITED NUMBER LEFT AND ARE SELLING THESE AT  
ONLY \$36.95 ON TAPE AND \$48.95 ON DISK. MANUAL IS  
COMPLETE WITH ADDENDUM TO ENABLE TRANSFER TO DISK.  
HURRY! THESE WON'T LAST.**



**S**trategy  
**S**oftware

**BOX 5000, GLEN WAVERLEY, VICTORIA 3150**

**(03) 560 4324**

# ALL NEW TITLES AT SPECIAL PRICES

NEW AMSTRAD TITLES  
AVAILABLE ONLY FROM  
STRATEGY SOFTWARE

**1. GENESIS ADVENTURE  
CREATOR**  
DISK ONLY \$39.95

Don't pay more! Genesis offers a complete system for writing Text/Graphic adventures with Music and Sound Effects. Features include: Text Compression, synonyms for commands and objects, sentence analyser, multiple graphic windows, variable screen modes, extensive graphic commands and storage of text and graphics on disk for large, disk-based adventures. Ideal for the novice wishing to create his/her own adventures. Used by commercial software houses in the U.K.

**2. GRASP PLUS**  
DISK ONLY \$29.95

All the features of Grasp, plus hi-res (mode 2) screen plots, improved labelling, various printer options, faster screen dumps, exploded pie charts and all the advantages of disk operation.

**3. MUSICO**  
TAPE ONLY \$17.95

An easy to use system for creating 3 part music and sound effects. Results may be used in your own programs.

**4. DRUMKIT**  
TAPE ONLY \$16.95

Percussion Instrument Simulator with eleven standard percussion sounds plus three user-defined sounds. Uses the Amstrad sound chip, no extra hardware required.

**5. CHAOS FACTOR**  
TAPE \$15.95  
DISK \$27.95

A graphic adventure game, featuring Nurd Fungus and Narsty & Crutch, the amazing cops who can solve any crime simply by saying 'Hi' a lot! Don't miss it!

**6. EASY MUSIC**  
DISK ONLY \$34.95

A combination of MUSICO and DRUMKIT (see above) together with the advantage of disk operation.

**7. AMSTRAD POT-POURRI  
VOLUME 1**  
TAPE \$ 9.95  
DISK \$ 19.95

25 great games for your Amstrad. Fantastic value for money at less than 40¢ per program on tape. Volume 1 has a full-length text adventure which alone is worth the money!

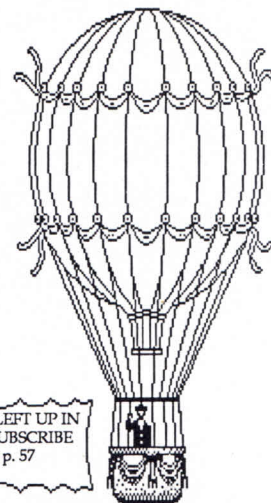
**8. AMSTRAD POT-POURRI  
VOLUME 2**

TAPE \$ 9.95  
DISK \$19.95

25 more previously un-released software on the same basis as Volume 1. Volume 2 includes a great machine-code Space Invaders.

## EXTRA SPECIAL!!!

Buy both Volumes 1 & 2 and receive, absolutely free, a tape copier and tape to disk utility. Please note that these programs cannot be guaranteed to work with all tapes.



# CONTENTS

6

## MY SAY

The editor says his bit.

30

## AIRO

Amstrad's new PC arrives in town and we give it the once over.

8

## DA BELLS

Esmeralda has been imprisoned again, this time in the Boston Stump. Guide the intrepid Hunchback to remove her to safety, avoiding all the obstacles in your way.

34

## CYRUS II CHESS

An in depth look at Cyrus II chess, the chess game everyone's talking about. It's available on the CPC, PCW and PC range.

14

## READY REFERENCE

Puzzled by MODE, INK and PEN? Our ready reference guide will help make life easier.

35

## TAKING STOCK

Jo Stork takes an overview of Stock Control in general and compares it with Stock recording. Do you know the difference?

15

## U.K. NEWS

All the latest news and gossip hot off the FAX machine from the people who know.

37

## SOFTWARE SURVEY

A bumper crop of new releases get the treatment from our intrepid reviewers.

20

## ADVENTURES

Wizard Gandalf presents a cheat program, reviews Lords of Midnight and has lots more hints for adventurers.

41

## PILOT

Another language for your Amstrad. Pilot is easy to understand and ideal for computer-aided learning.

24

## CHARACTER GENERATOR

Create and manipulate your own personal character set with this superb utility. Once you've used it you'll wonder how you ever managed without it.

46

## SIMPLE SPRITES

By making use of Amstrad Basic, logical operators and inbuilt interrupts, you can create sprite-like effects to enhance your programs.

48

## ANALYSIS

Join Trevor Roberts as he casts his eagle eye over some short, simple programs. You're bound to find some useful ideas here.

49

## POSTBAG

The part of the magazine you write yourselves. Just a small selection from the many interesting and informative letters you've been sending us.

52

## POUNCE ON A MOUSE!

Those furry blighters are hard to catch at the best of times. This one will cause you untold problems - it can even gnaw through the wall that you're building to trap it!

55

## SCROLLER

Add that professional touch to your text with this slick sideways scrolling routine.

56

## SUBSCRIPTION OFFER

For the few who haven't yet subscribed we prove we'll go to any lengths to get your money!

57

## ORDER FORM

Software, books, mags and other, this is where you can place your order without leaving your chair.

58

## LETTER LITTER

Collect the rubbish and learn your way around the keyboard in this entertaining educational game for all the family.

63

## NEXT MONTH

We tell you some, but not all, of the things we have lined up for you next month.

Computing With The Amstrad is a monthly publication published under license by Strategy Software from Database Publications. All text, computer listings and other content are copyright material.



Page 6 eh? - somebody must love me. Thought you'd heard the last of me didn't you? Seriously, a couple of things occurred to me during the last few weeks that I thought were worth passing on.

Last month I made mention of the Sales Tax imposed in the 1986 Budget. Since then I've had chance to look at the act itself - and what a mess!

Firstly, there are two levels of tax. As with other manufactured products a tax is levied at the wholesale stage of the sales process with tax payable at 20% of the wholesale price. If, for instance, the wholesale price of a piece of software was \$75 and the retail \$100, the \$75 now attracts the 20% sales tax (\$15) and the retail price effectively increases by 15% to \$115. Unlike other manufactured goods, and unique in Australian Sales Tax law (and probably the world), this tax does not apply where the manufacturer sells direct to the end user. Instead the tax is levied on the full retail price - in ALL other cases where the manufacturer sells direct the tax is levied on a theoretical wholesale price on a similar basis to that outlined above. It would seem that we're head for a Retail Sales Tax by the back door. What does it mean to you? Other than higher prices probably not very much because Sales Tax is not quoted as a separate item in this country - if you look at the U.K. system of V.A.T. you will see ads in magazines that quote a retail price 'plus VAT', at least then you know how much tax you're paying (small comfort!).

HERE'S THE RUB! - If you obtain a program and read the code into your own computer for your own use then you are exempted from the tax - if you make a back-up copy (illegal or otherwise) you are a manufacturer and have to register with the the Taxation Dept. and pay Sales Tax on the goods you manufacture! Pirates look out! It is obvious to anyone with a modicum of commonsense that the act is not intended to hit the end user with this tax. But people who consistently make copies of software and give them away or sell them should watch out.

On the subject of piracy it may interest readers to know of a situation which we were involved in earlier this year. A tape was returned to us as being faulty and the user informed us that he was given the cassette by a friend - could we please fix it? We soon contacted this person and shortly afterwards a solicitors letter was on its way to 'the pirate'. Within a week we were offered, and accepted, \$1000 damages with costs. At the same time, we obtained a written agreement from the party concerned wherein he guaranteed not to be naughty any more and giving us the names and addresses of all the people he had provided with copies of our tapes. I was pleased with the results, having expected nothing more than the agreement mentioned above, and asked our solicitors why the matter was resolved so quickly and why the other party was so generous with his damages offer. We had after all considered the copyright act to be a bit 'toothless'. I was astounded when I learned that it wasn't so much the breach of copyright that had caused the result but the fact that the other party could have faced a charge of theft for each tape he had supplied and that he would probably have received a \$50 fine for each occurrence. If that wasn't enough, the really big stick was the Trade Practices Act which carries fines of \$10000 and up for each breach. It would seem that by copying the software containing our name, he had illegally used our name and was therefore in breach of the TPA.

So what's all this mean to you? As I mentioned earlier, if you copy software, you're liable for Sales Tax. Ever tried to run away from the Tax Dept? The long and short of it is - DON'T DO IT! We know our software from this magazine is appearing in user group libraries around the country. This software is NOT public domain - it is copyright material and permission is not granted for it's use other than by the original purchaser for his/her own use. We would strongly advise those involved to cease this practise and get the software out of your library. One of the main justifications given by 'pirates' is that software is too expensive, we feel that if

this were true (and we don't think it is) then we must surely be exempt from piracy on the basis that we are selling software on a high quality cassette including postage for only \$3. How cheap does it have to get?

Enough of the lecture, let's look at what Amstrad are up to now they've released the Airo which you can read about elsewhere in this issue. I have a great deal of respect for the intelligence and marketing skills of Alan Sugar and got to thinking what had happened on the home computer front while all the publicity in recent months has been aimed at the new PC. Could it be that AMS gave the CPC design team a few months off to relax after the 6128 was released? - hardly likely. What's been happening at Spectrum now that Amstrad have taken over? Did Alan Sugar buy Spectrum just for fun and then sell off all the stock at ridiculous prices just to be Mr. Nice Guy to the people who bought them? - again, hardly likely. So what are they up to? If you analyze the structure of the home computer market it falls readily into groups at the sub \$500 level, \$500-1000 and then takes us into the business market at around \$1500 plus. To paraphrase Dirty Harry - 'Ask yourself one question - now that Amstrad have hit the big time, which area of the market isn't being addressed by Amstrad?' Ask yourself another question - which area of the market was Spectrum into in a big way? Now we can start to speculate. Will a new home computer arrive in the sub \$500 bracket which will run both Amstrad and Spectrum software? Will there be a Spectrum with a built-in cassette deck or 3" disk system? Will there be an Amstrad with no cassette or disk built-in? My bet is on a non-Amstrad compatible Spectrum with the option of built-in tape or disk but no monitor. My reasoning is that the U.K. domestic computer market has lived on a diet of Spectrum too long to see it disappear. At the same time, I can't believe Amstrad would wish to spoil the success of their current range by introducing another model which competes with them. I wonder what they're really up to?

Kevin Poynton

# SPECIAL!

In a limited pre-Christmas offer, the Giltronic Accounting package includes:-

PCW-8512, Accounting 1 software, 2 \* CF2-D discs, 2 \* CF2-DD discs, an assortment of pre-printed computer paper, Locoscript Tutor audio tapes and, for your entertainment, Cyrus II chess.

Total value in excess of \$2650

\*\*\*\*\*

**SAVE OVER \$400**

\*\*\*\*\*

Our price (inc. insured freight).....\$2250

\*\*\*\*\*

**Accounting One for PCW-8256 owners - \$1099**

Package consists:-

Accounting 1 software, 1 \* Giltronic 5.25" 1 megabyte disc drive, Interface and full fitting instructions. (Minor alterations are required to monitor case to allow cable access)

\*\*\*\*\*

PCW-8256 RAM upgrades available mid-November price to be announced - call for further details.

\*\*\*\*\*

**Why use expensive 3" discs???????**

**Fit a Giltronics 5.25" Disc Drive to your Amstrad!**

Drives suitable for:

CPC-464/6128 from \$319  
PCW-8256/8512 from \$349

1 Meg 5.25" "B" Drive for PCW-8256 (including interface etc.....\$535)  
Installation can be arranged (in Melb.) for \$36

\*\*\*\*\*

**Full range of Software and Hardware for Amstrad computers.**

**CALL  
(03) 580 9839  
NOW!**

# Giltronic

AUSTRALIA

# AMSTRAD

## MAIL ORDER

SOFTWARE & HARDWARE

## (03) 580 9839

### AMSTRAD PC1512



**IBM  
COMPATIBLE**  
(from \$1495)



CPC 464

**M**R HUMPY has the hump. The beautiful Esmerelda has been captured by the French when they invaded Lincolnshire, and she has been taken to the dark and damp Boston-de-Stump.

You must guide Mr Humpy over the roof of the Stump and rescue Esmerelda from her cell.

However this is no easy matter as you will have to avoid lumps of rock, jump over chasms using only a slippery rope, cross moving drawbridges and fight off Frenchmen.

Also on each screen is a barrel of TNT which is about to explode. If it does you lose a life.

To move onto the next screen you must ring the bell. When you ring the bell on the 15th screen you get a super bonus and Mr Humpy is returned to Screen 1 — where the fuse on the TNT is burning faster.



## KEYS

- Z — Right
- X — Left
- Space — Jump

By ARAMELLO CHAPMAN

## ROUTINES

- 140 Main loop. Go to the various hazard moving routines. Mr Humpy moving routines.
- 230 Bonus count, decreases bonus by (1/bonus) and prints fuse.
- 300 Makes TNT explode.
- 350 Moves Humpy. Reads keyboard. If jump key pressed to jump routines.
- 560 Makes Humpy jump, tests to see if he is on rope screen.
- 800 Updates position of fire balls.
- 970 Updates position of guard poles.
- 1070 Moves bridge on screen 9.
- 1210 Moves rope on screens 1 and 13.
- 1300 Moves drawbridge on screen 10.
- 1380 Moves all hazards on last screen.
- 1520 } Set up screens layout and all variables of
- 2120 } hazards used.
- 2130 Variable dump.
- 2270 Makes Mr Humpy ring bell and gives bonus.
- 2370 Graphics characters.
- 2670 Death of Mr Humpy.

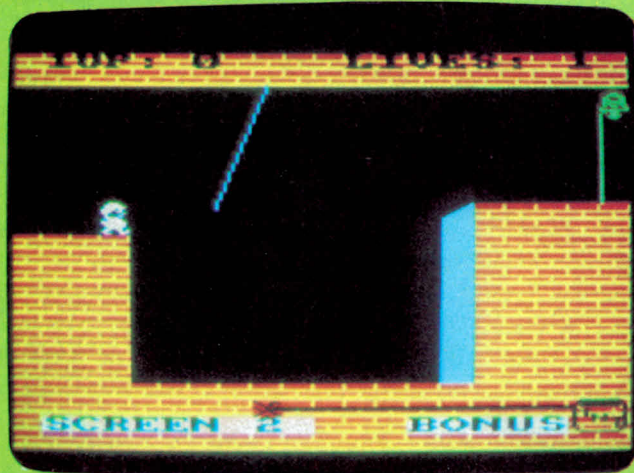
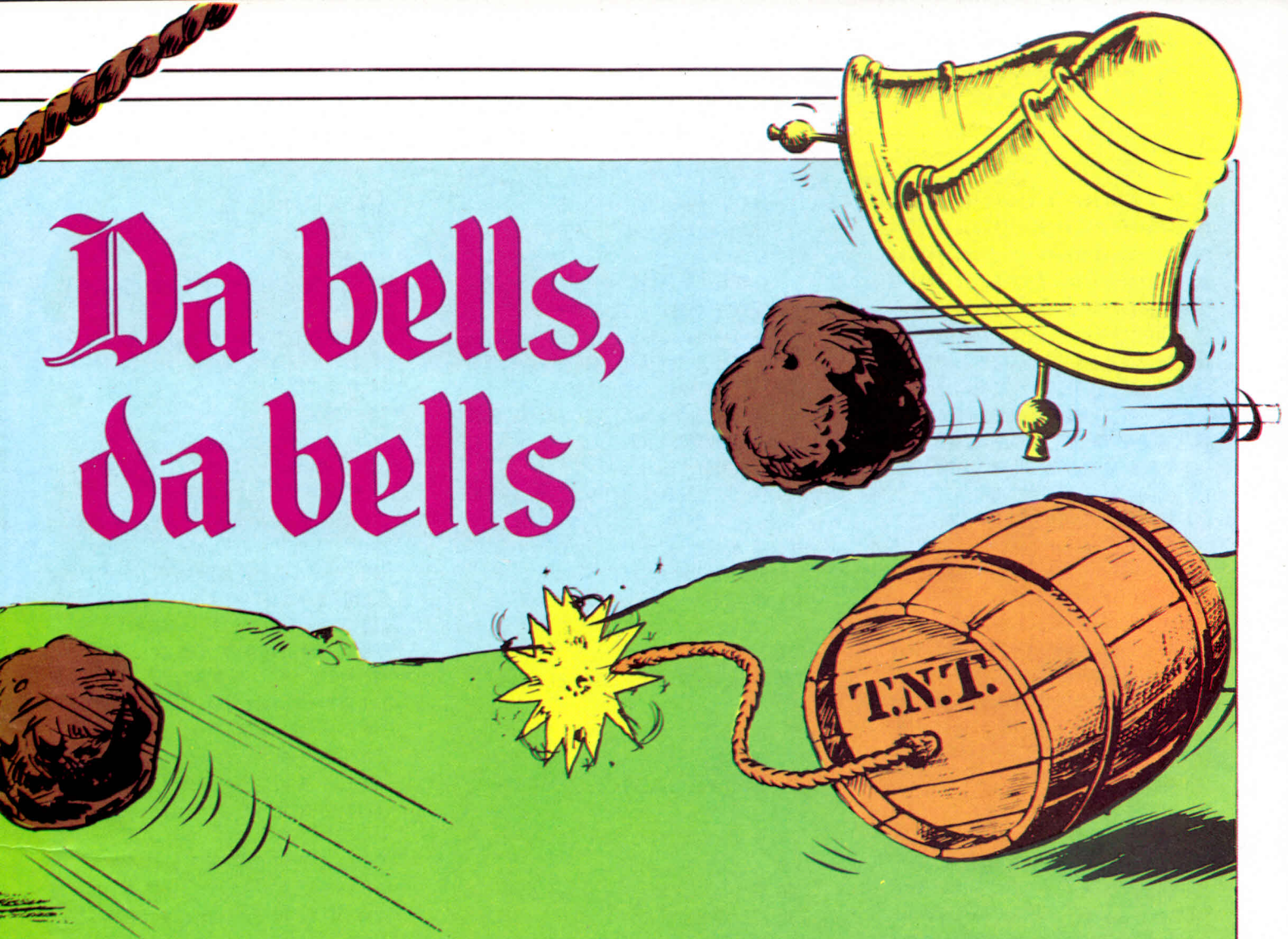
- 2880 Prints high score table and allows entry of name if score high enough.
- 3280 Instructions.
- 3720 Plays tune and gives super bonus when all 15 screens completed.

## VARIABLES

- X, Y Mr Humpy's position.
- ropeh If Mr Humpy is on a rope = 1, if not = 0.
- rope Position of swinging rope.
- roped Rope direction.
- pole Position of pole/portcullis.
- poled Pole direction.
- ball Position of rock.
- balld Direction of rock.
- ia\$ Name of high scorer.
- hs\$ High score.
- bonus Bonus left.
- bonusd Amount by which bonus is decreasing.
- screen Screen number.
- bridge Bridge position.
- bridir Bridge direction.
- score Score.
- lives Lives left.



# Da bells, da bells



```

10 REM*****
20 REM*****DA BELLS*****
30 REM*****
40 REM*****By A.Chapman*****
50 REM*****
60 REM(C)Computing with the Amstrad
70 DIM na$(10):DIM hs(10)
80 FOR f=1 TO 8:LET na$(f)="Mr Humpy
*:LET hs(f)=2000-(f*200):NEXT f
90 GOSUB 2370:REM SET UP U.D.G
100 GOSUB 3280:REM INSTRUCTIONS
110 GOSUB 2130:REM SET UP VARIABLES
120 GOSUB 1520:REM SET UP SCREEN
130 PRINT CHR$(22)+CHR$(1):PEN 0:LOCA
TE 18,1:PRINT lives:PRINT CHR$(22)+CH
R$(0):PEN 1
140 REM*****
150 REM*****MAIN LOOP*****
160 REM*****
170 ON screen GOSUB 800,1210,960,890,
800,970,960,970,1070,1300,890,1070,12
10,800,1380
180 GOSUB 350
190 GOSUB 230
200 IF x=19 THEN GOTO 2270
210 IF bonus<=1 THEN GOTO 300
220 GOTO 170
230 REM*****
240 REM*****BONUS COUNT*****
250 REM*****
260 LOCATE 19-INT(bonus),22:PAPER 1:P
EN 3:PRINT CHR$(225);:PAPER 3:PEN 0:P
RINT CHR$(242)
270 LET bonus=bonus-(1/bonusd)
280 PAPER 0:PEN 1
290 RETURN
300 REM*****
310 REM*****EXPLOSION*****
320 REM*****
330 FOR f=1 TO 5:SOUND 1,0,10,7,0,0,3
: INK 0,3:BORDER 3:SOUND 1,0,25,5,0,0
,10:INK 0,0:BORDER 0:NEXT f
340 FOR f=1 TO 100:NEXT f:GOTO 2690
350 REM*****
360 REM*****MOVE HUMPY*****
370 REM*****
380 IF INKEY(71)=1 AND INKEY(47)=1 AN
D INKEY(63)=1 THEN GOTO 420
390 IF INKEY(47)=0 THEN GOSUB 560:GOT
O 420
400 IF INKEY(63)=0 AND x<19 THEN GOSU
B 480:LET x=x+1:SOUND 1,200,1:m=1
410 IF INKEY(71)=0 AND x>1 THEN GOSUB
520:LET x=x-1:SOUND 1,200,1:LET m=0
420 IF TEST(32*(x-1)+8,16*(25-(y))+8)
=7 OR TEST(32*(x-1)+16,16*(25-(y))-4)
<>3 OR TEST(32*(x-2)+8,16*(25-(y))+8)
=7 THEN GOTO 2670
430 IF m=1 AND m2=1 THEN PEN 4:LOCATE

```

```

x,y:PRINT CHR$(227):LOCATE x,y-1:PRI
NT CHR$(226):PEN 1:GOTO 450
435 IF m=1 AND m2=2 THEN PEN 4:LOCATE
x,y:PRINT CHR$(228):LOCATE x,y-1:PRI
NT CHR$(226):PEN 1:GOTO 450
440 IF m=0 AND m2=3 THEN PEN 4:LOCATE
x,y:PRINT CHR$(247):LOCATE x,y-1:PRI
NT CHR$(249):PEN 1:GOTO 450
445 IF m=0 AND m2=4 THEN PEN 4:LOCATE
x,y:PRINT CHR$(248):LOCATE x,y-1:PRI
NT CHR$(249):PEN 1
450 IF TEST(32*(x)+8,16*(25-(y))+8)=7
THEN GOTO 2670
460 IF screen=12 AND (x=4 OR x=18) OR
screen=15 AND x=16 THEN GOTO 2670
470 RETURN
480 LOCATE x,y:PRINT " :LOCATE x,y-1:
PRINT "
490 IF m2=1 OR m2=3 THEN LET m2=2:RET
URN
500 IF m2=2 OR m2=4 THEN LET m2=1
510 RETURN
520 LOCATE x,y:PRINT " :LOCATE x,y-1:
PRINT "
530 IF m2=1 OR m2=3 THEN LET m2=4:RET
URN
540 IF m2=2 OR m2=4 THEN LET m2=3
550 RETURN
560 REM*****
570 REM*****HUMPY JUMP*****
580 REM*****
590 IF INKEY(47)=0 AND INKEY(71)<>0 A
ND INKEY(63)<>0 THEN m1=1
600 FOR a=-0.75 TO 0.75 STEP 0.5:LOCA
TE x,y:PRINT " :LOCATE x,y-1:PRINT "
":LET y=y+3*a
610 IF m=1 AND m1=0 AND x<19 THEN LET
x=x+1
620 IF m=1 THEN PEN 4:LOCATE x,y:PRIN
T CHR$(228):LOCATE x,y-1:PRINT CHR$(2
26):PEN 1
630 IF m=0 AND m1=0 AND x>1 THEN LET
x=x-1
640 IF m=0 THEN PEN 4:LOCATE x,y:PRIN
T CHR$(248):LOCATE x,y-1:PRINT CHR$(2
49)
650 ON screen GOSUB 800,1210,960,890,
800,970,960,970,1070,1300,890,1070,12
10,800,1380
660 IF (screen=13 OR screen=2) AND ro
peh=0 AND x-1=INT(rope/8) OR (screen=
13 OR screen=2) AND ropeh=0 AND x=INT
(rope/8) THEN GOTO 740
670 IF TEST(32*(x-1)+16,16*(25-(y))-4
)=12 OR TEST(32*(x-2)+16,16*(25-(y))+
8)=7 OR TEST(32*(x-1)+16,16*(25-(y-1
))+8)=12 THEN GOTO 2670
680 IF TEST(32*(x-1)+16,16*(25-(y))+8)
=7 OR TEST(32*(x-1)+16,16*(25-(y))+2

```

```

4)=7 THEN GOTO 2670
690 NEXT a
700 LET m1=0
710 LOCATE x,y:PRINT " :LOCATE x,y-1:
PRINT "
720 IF TEST(32*(x-1)+16,16*(25-(y))-4)
<>3 OR TEST(32*(x)+16,16*(25-(y))+8)
=7 THEN GOTO 2670
730 RETURN
740 LET ropeh=1:LOCATE x,y:PRINT " :L
OCATE x,y-1:PRINT " :LET x=INT(rope/8)
:LET y=9
750 LOCATE x,y:PRINT " :LOCATE x,y-1:
PRINT " :LET x=INT(rope/32)+5
760 PEN 4:LOCATE x,y-1:PRINT CHR$(226)
:LOCATE x,y:PRINT CHR$(229):PEN 1
770 GOSUB 1210
780 IF INKEY(47)=0 THEN GOTO 560
790 GOTO 750
800 REM*****
810 REM*****MOVE FIRE BALLS*****
820 REM*****
830 LOCATE ball,11:PRINT " :LOCATE 20
-ball,9:PRINT "
840 LET ball=ball-1
850 IF ball=1 THEN LET ball=19
860 PEN 7:LOCATE ball,11:PRINT CHR$(2
30):LOCATE 20-ball,9:PRINT CHR$(230)
870 PEN 1
880 FOR f=1 TO 20:NEXT f:RETURN
890 LOCATE ball,9:PRINT "
900 IF screen=8 AND (ball=6 OR ball=1
1 OR ball=16) THEN PEN 12:LOCATE ball
,9:PRINT CHR$(232):PEN 1
910 LET ball=ball-1
920 IF ball=1 THEN LET ball=19
930 PEN 7:LOCATE ball,9:PRINT CHR$(23
0)
940 IF screen<>8 THEN FOR f=1 TO 20:N
EXT f
950 RETURN
960 FOR f=1 TO 70:NEXT f:RETURN
970 REM*****
980 REM*****MOVE GUARDS POLES*****
990 REM*****
1000 FOR f=6 TO 16 STEP 5:PEN 12:LOCA
TE f,pole:PRINT CHR$(232):LOCATE f,po
le-1:PRINT " :NEXT f
1010 LET pole=pole+poled
1020 IF pole=5 THEN LET poled=1
1030 IF pole=11 THEN LET poled=-1
1040 PEN 1
1050 IF screen=8 THEN GOSUB 890
1060 RETURN
1070 REM*****
1080 REM*****MOVE BRIDGE*****
1090 REM*****
1100 PAPER 1 :PEN 3:LOCATE bridge,12:
PRINT CHR$(225);:PAPER 0:PRINT " :LOC

```

```

ATE bridge=4,12:PRINT" ";:PAPER 1:PRI
NT CHR$(225):PAPER 0:PEN 1
1110 LET bridge=bridge+bridir
1120 IF bridge=9 THEN LET bridir=1
1130 IF bridge=13 THEN LET bridir=-1
1140 IF screen=9 THEN GOTO 1190
1150 LOCATE 4,pole:PEN 12:PRINT CHR$(
232):LOCATE 17,pole:PRINT CHR$(232):P
EN 1:LOCATE 4,pole-1:PRINT" ":LOCATE
17,pole-1:PRINT" "
1160 LET pole=pole+poled
1170 IF pole=7 THEN LET poled=1
1180 IF pole=11 THEN LET poled=-1
1190 PAPER 0:PEN 1
1200 RETURN
1210 REM*****
1220 REM*****MOVE ROPE*****
1230 REM*****
1240 PLOT 270,368:DRAWR ROPE-128,-120
,0
1250 LET ROPE=ROPE+ROPED
1260 IF ROPE>225 THEN LET ROPED=-8
1270 IF ROPE<40 THEN LET ROPED=8
1280 PLOT 270,368:DRAWR ROPE-128,-120
,10
1290 RETURN
1300 REM*****
1310 REM*****MOVE DRAWBRIDGE*****
1320 REM*****
1330 PAPER 1:PEN 3:LOCATE bridge,12:P
RINT CHR$(225):LOCATE 19-bridge,12:PR
INT CHR$(225):PAPER 0:LOCATE bridge+1
,12:PRINT" ":LOCATE 18-bridge,12:PRIN
T" "
1340 LET bridge=bridge+bridir:IF brid
ge=8 THEN LET bridir=-bridir
1350 IF bridge=5 THEN LET bridir=1
1360 PEN 1
1370 RETURN
1380 REM*****
1390 REM*****LAST SCREEN*****
1400 REM*****
1410 LOCATE 5,pole:PEN 12:PRINT CHR$(
232):LOCATE 10,pole-3:PRINT CHR$(149)
;CHR$(149);CHR$(149):LOCATE 10,pole-2
:PRINT CHR$(231);CHR$(231);CHR$(231):
LOCATE 5,pole-1:PRINT" ":LOCATE 10,po
le-1:PRINT" "
1420 LOCATE 16,pole-1:PRINT CHR$(232)
:LOCATE 16,pole-2:PRINT" "
1430 LET pole=pole+poled
1440 IF pole=6 THEN LET poled=1
1450 IF pole=12 THEN LET poled=-1
1460 LOCATE ball,11:PRINT" "
1470 LET ball=ball+balld
1480 IF ball=15 THEN LET balld=-1
1490 IF ball=7 THEN LET balld=1
1500 LOCATE ball,11:PEN 7:PRINT CHR$(
230)
1510 PEN 1:RETURN
1520 REM*****
1530 REM*****SCREEN SET UP*****
1540 REM*****
1550 MODE 0:IF screen=16 THEN screen=
1
1560 ON screen GOSUB 1580,1950,1720,1
720,1720,1770,2080,1770,1900,1860,208
0,1900,1950,2080,2010
1570 RETURN
1580 FOR f=12 TO 24
1590 PAPER 1:PEN 3
1600 LOCATE 1,f:PRINT STRING$(20,CHR$(
225))
1610 NEXT f
1620 FOR f=1 TO 2:LOCATE 1,f:PRINT ST
RING$(20,CHR$(225)):NEXT f
1630 PAPER 0:PEN 1
1640 LET ball=19
1650 LOCATE 20,3:PEN 13:PRINT CHR$(23
5):LOCATE 20,4:PRINT CHR$(236):FOR f=
5 TO 10:LOCATE 20,f:PRINT CHR$(237):N
EXT f:PEN 1
1660 PRINT CHR$(22)+CHR$(1)
1670 LOCATE 19,22:PAPER 1:PEN 0:PRINT
CHR$(238);CHR$(239):LOCATE 19,23:PRI
NT CHR$(240);CHR$(241):LOCATE 1,22:PR
INT STRING$(18,CHR$(154)):PAPER 0:PEN
1
1680 LOCATE 2,1:PEN 0:PRINT"1UP:";sco
re:LOCATE 12,1:PRINT"LIVES:";PEN 1
1690 PRINT CHR$(22)+CHR$(0)
1700 LOCATE 2,23:PAPER 4:PEN 0:PRINT"
SCREEN";screen:LOCATE 14,23:PRINT"BON
US":PAPER 0:PEN 1
1710 RETURN
1720 GOSUB 1580
1730 FOR f=5 TO 15 STEP 5
1740 LOCATE f,12:PRINT" ":LOCATE f,1
3:PRINT" ":LOCATE f,14:PRINT" ":NEX
T f
1750 FOR f=7 TO 17 STEP 5:PEN 2:LOCAT
E f,12:PRINT CHR$(214):LOCATE f,13:PR
INT CHR$(143):LOCATE f,14:PRINT CHR$(
143):PEN 1:NEXT f
1760 RETURN
1770 GOSUB 1720
1780 FOR f=6 TO 16 STEP 5:PEN 10:LOCA
TE f,13:PRINT CHR$(233):LOCATE f,14:P
RINT CHR$(234):NEXT f
1790 PEN 1
1800 pole=12:poled=-1
1810 RETURN
1820 GOSUB 1580
1830 FOR f=12 TO 20:LOCATE 5,f:PRINT
STRING$(11," "):NEXT f
1840 FOR f=12 TO 20:LOCATE 15,f:PEN 2
:PRINT CHR$(143):NEXT f:LOCATE 15,12:
PRINT CHR$(214):PEN 1
1850 RETURN
1860 GOSUB 1820
1870 PAPER 1:PEN 3:LOCATE 15,12:PRINT
CHR$(225):PEN 1:PAPER 0
1880 LET bridge=5:LET bridir=1
1890 RETURN
1900 GOSUB 1820
1910 LET bridge=13:LET bridir=-1
1920 LET ball=19
1930 LET pole=11:LET poled=-1
1940 RETURN
1950 GOSUB 1820
1960 LET rope=128:LET roped=8:LET rop
eh=0
1970 LOCATE 16,11:PAPER 1:PEN 3:PRINT
STRING$(5,CHR$(225)):LOCATE 16,10:PR
INT STRING$(5,CHR$(225)):PAPER 0:PEN
1
1980 LOCATE 15,11:PEN 2:PRINT CHR$(14
3):LOCATE 15,10:PRINT CHR$(214)
1990 LOCATE 15,12:PRINT CHR$(143):PEN
1
2000 RETURN
2010 GOSUB 1580
2020 LOCATE 4,12:PRINT" ":LOCATE 4,1
3:PRINT" ":LOCATE 4,14:PRINT" ":PEN
2:LOCATE 6,12:PRINT CHR$(214):FOR f=
13 TO 14:LOCATE 6,f:PRINT CHR$(143):N
EXT f:PEN 1
2030 LOCATE 17,5:PAPER 1:PEN 3:PRINT
STRING$(3,CHR$(225)):LOCATE 18,6:PRIN
T CHR$(225);CHR$(225):LOCATE 9,3:PRIN
T STRING$(5,CHR$(225)):LOCATE 9,4:PRI
NT STRING$(5,CHR$(225)):PAPER 0
2040 LOCATE 18,3:PEN 11:PRINT CHR$(25
4):LOCATE 18,4:PRINT CHR$(255):PEN 1
2050 LET ball=12:LET balld=1:LET pole
=12:LET poled=-1
2060 PEN 10:LOCATE 5,13:PRINT CHR$(23
3):LOCATE 5,14:PRINT CHR$(234):PEN 1
2070 FOR f=4 TO 10:PEN 12:LOCATE 10,f
:PRINT CHR$(149);CHR$(149);CHR$(149):
NEXT f:PEN 1:RETURN
2080 GOSUB 1820
2090 LOCATE 8,12:PAPER 1:PEN 3:PRINT
CHR$(225):LOCATE 12,12:PRINT CHR$(225
)
2100 PRINT CHR$(22)+CHR$(1):LOCATE 8,
3:PEN 13:PRINT CHR$(235);" ";CHR$(
235):LOCATE 8,4:PRINT CHR$(236);"
";CHR$(236):LOCATE 8,5:PRINT CHR$(237
);" ";CHR$(237):LOCATE 8,6:PRINT C
HR$(237);" ";CHR$(237)
2110 PRINT CHR$(22)+CHR$(0):PAPER 0:P
EN 1
2120 RETURN
2130 REM*****
2140 REM*****VARIABLE DUMP*****
2150 REM*****

```

```

2170 LET y=1:LET x=1
2180 screen=VAL(chose$)
2190 PEN 1
2200 LET ropeh=0
2210 LET bonus=10:LET bonusd=5
2220 LET lives=3:LET m=1
2230 LET score=0:LET m2=1
2240 ENV 6,15,-1,1:ENT 1,30,10,1:ENV
5,15,-1,10
2250 ENV 1,10,-1,2:ENV 2,15,1,3:ENV 3
,1,0,2,15,-1,20
2260 RETURN
2270 REM*****
2280 REM*****REACH END OF SCREEN****
2290 REM*****
2300 FOR f=1 TO 4:PEN 4:LOCATE x,y-1:
PRINT CHR$(226):LOCATE x,y:PRINT CHR$(
229):LOCATE x,y-2:PRINT " ":SOUND 1,1
40,12,0,1:SOUND 1,120,12,0,1:LOCATE x
,y:PRINT " ":LOCATE x,y-1:PRINT CHR$(2
29):LOCATE x,y-2:PRINT CHR$(226)
2310 SOUND 1,120,12,0,1:SOUND 1,140,1
2,0,1:NEXT f
2320 LET score=score+10*INT(bonus)
2330 LET screen=screen+1
2340 LET x=1:LET y=11:LET bonus=10
2350 IF screen=16 THEN GOSUB 3720
2360 GOTO 120
2370 REM*****
2380 REM*****U.D.G*****
2390 REM*****
2400 SYMBOL AFTER 224
2410 SYMBOL 225,254,254,254,0,239,239
,0
2420 SYMBOL 226,56,124,126,244,236,19
4,68,120
2430 SYMBOL 227,60,254,254,62,60,24,2
4,30
2440 SYMBOL 228,60,254,254,126,60,109
,199,102
2450 SYMBOL 229,250,255,126,56,56,29,
15,7
2460 SYMBOL 254,0,60,46,70,70,39,56,5
5
2470 SYMBOL 255,250,60,60,126,126,126
,255,36
2480 SYMBOL 230,24,62,126,255,255,126
,126,40
2490 SYMBOL 231,24,24,24,24,24,126,60
,24
2500 SYMBOL 232,24,24,60,90,24,60,90,
24
2510 SYMBOL 233,60,66,153,129,165,231
,126,60
2520 SYMBOL 234,24,126,231,165,36,36,
36,231
2530 SYMBOL 235,0,0,24,60,66,102,90,1
02
2540 SYMBOL 236,90,129,255,88,152,188

```

```

,152,128
2550 SYMBOL 237,128,128,128,128,128,1
28,128,128
2560 SYMBOL 238,31,63,63,223,192,92,7
2,74
2570 SYMBOL 239,240,252,252,250,2,2,2
,66
2580 SYMBOL 240,75,66,66,64,64,64,32,
31
2590 SYMBOL 241,66,222,74,10,10,10,4,
248
2600 SYMBOL 242,132,74,40,151,95,40,6
0,146
2610 SYMBOL 243,0,16,45,103,114,125,2
0,0
2620 SYMBOL 244,0,2,226,254,254,224,0
,0
2630 SYMBOL 247,60,127,127,126,60,182
,227,102
2640 SYMBOL 248,60,127,127,127,60,24,
24,120
2650 SYMBOL 249,28,62,126,47,55,67,34
,30
2655 SYMBOL 250,198,165,198,165,6,40,
40,16
2660 RETURN
2670 REM*****
2680 REM*****
2690 REM*****
2700 REM*****DEATH SEQUENCE*****
2710 REM*****
2720 LET x=x+1
2730 LOCATE x-1,y-1:PRINT " ":LOCATE
x-1,y:PEN 15:PRINT CHR$(243);CHR$(244
)
2740 PEN 1
2750 IF bonus<=1 THEN FOR f=1 TO 500:
NEXT f:GOTO 2810
2760 RESTORE 2760
2770 FOR f=1 TO 10
2780 READ n,d,n1,d1,n2,d2
2790 SOUND 1,n,d,7:SOUND 2,n1,d1,6:50
UND 4,n2,d2,4
2800 NEXT f
2810 LET lives=lives-1
2820 INK 2,20
2830 IF lives=0 THEN GOTO 2880
2840 LET bonus=10:LET x=1:LET y=11
2850 GOTO 120
2860 DATA 1276,100,319,100,159,100,11
36,20,284,20,142,20,1073,50,268,50,13
4,50,1276,20,319,20,159,20,0,100,0,10
0,0,100
2870 DATA 1276,100,319,100,159,100,11
36,20,284,20,142,20,1073,50,268,50,13
4,50,1276,70,319,70,156,70,902,100,22
5,100,113,100
2880 REM*****
2890 REM*****HIGH SCORE*****

```

```

2900 REM*****
2910 MODE 1
2920 LOCATE 15,5:PEN 3:PRINT"HIGH SCO
RE"
2930 PEN 2
2940 LOCATE 4,6:PRINT CHR$(150);:PRIN
T STRING$(30,CHR$(154));:PRINT CHR$(1
56)
2950 FOR f=7 TO 17:LOCATE 4,f:PRINT C
HR$(149):LOCATE 35,f:PRINT CHR$(149):
NEXT f
2960 LOCATE 4,18:PRINT CHR$(147);:PRI
NT STRING$(30,CHR$(154));:PRINT CHR$(
153)
2970 FOR f=1 TO 8
2980 IF score>hs(f) THEN GOTO 3090
2990 NEXT f
3000 FOR f=1 TO 8:PEN 1:LOCATE 8,f+8:
PRINT na$(f):LOCATE 10,f+8:PEN 3:PRIN
T".....";hs(f):NEXT f
3010 LOCATE 1,19:PRINT STRING$(120,"
")
3020 IF INKEY("<") THEN GOTO 3020
3030 PEN 2:LOCATE 1,22:PRINT STRING$(
40,CHR$(154)):LOCATE 1,24:PRINT STRIN
G$(40,CHR$(154)):PEN 1:LOCATE 6,23:IN
PUT"SELECT START SCREEN (1-15)";chose
$
3040 IF VAL(chose$)>15 OR VAL(chose$)
<1 THEN LOCATE 1,23:PRINT STRING$(40,
" "):GOTO 3030
3050 LOCATE 1,23:PRINT STRING$(40," "
)
3060 LOCATE 11,23:PEN 3:PRINT"PRESS";
:PEN 1:PRINT"< SPACE >";:PEN 3:PRINT"
TO PLAY.":PEN 1
3070 IF INKEY(47)<>0 THEN GOTO 3070
3080 GOTO 110
3090 LET a$="ABCDEFGHIJKLMNPOQRSTUVWXYZ
.#%&()!{}?*" +CHR$(250)
3100 LET c=19:LOCATE 1,22:PEN 1:PRINT
a$
3110 LOCATE 2,19:PEN 3:PRINT"USE CURS
OR KEY'S LEFT,RIGHT AND COPY":LOCATE
3,20:PRINT"TO SELECT LETTERS.(MAXIMUM
OF 10.)":PEN 2:LOCATE 1,21:PRINT STR
ING$(40,CHR$(154))
3120 LET x$=""
3130 FOR z=1 TO 10
3140 LOCATE c,23:PEN 2:PRINT " "
3150 IF INKEY(1)=0 AND c<40 THEN LET
c=c+1
3160 IF INKEY(8)=0 AND c>1 THEN LET c
=c-1
3165 IF INKEY(9)=0 AND c=40 THEN LOCA
TE 7,f+8:PRINT " ":LET x$=""
:GOTO 3120
3170 IF INKEY(9)=0 THEN LET x$=x$+MID
$(a$,c,1):LOCATE 7+z,f+8:PEN 1:PRINT

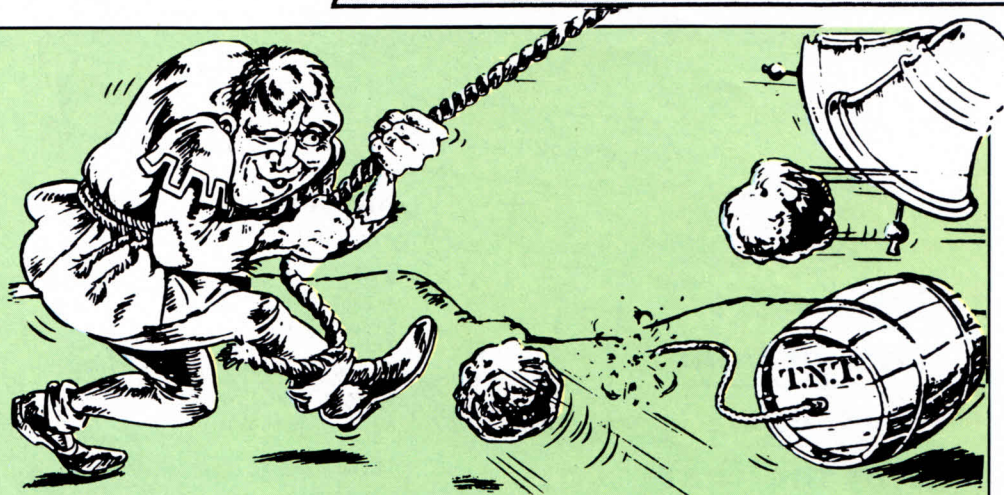
```

# Game of the Month

```

MID$(a$,c,1):FOR a=1 TO 200:NEXT a:NE
XT z:GOTO 3210
3180 LOCATE c,23:PRINT"*"
3190 FOR a=1 TO 50:NEXT a
3200 GOTO 3140
3210 LET hs(0)=score:LET na$(0)=x$
3220 LET f=0
3230 FOR z=1 TO 7
3240 IF hs(z)<hs(z+1) THEN LET t=hs(z
+1):LET hs(z+1)=hs(z):LET hs(z)=t:LET
a$=na$(z+1):LET na$(z+1)=na$(z):LET
na$(z)=a$:LET f=1
3250 NEXT z
3260 IF f=1 THEN GOTO 3220
3270 GOTO 3000
3280 REM*****
3290 REM*****INSTRUCTIONS*****
3300 REM*****
3310 MODE 1:INK 0,0:BORDER 0
3320 PAPER 1:PEN 3:PRINT STRING$(40,C
HR$(225)):FOR f=1 TO 7:LOCATE 1,f:PRI
NT CHR$(225):LOCATE 40,f:PRINT CHR$(2
25):NEXT f:PAPER 0
3330 LOCATE 17,3:PEN 2:PRINT STRING$(
0,CHR$(131)):LOCATE 17,4:PEN 3:PRINT"
DA BELLS":PEN 1:LOCATE 17,5:PRINT STR
ING$(0,CHR$(140))
3340 PAPER 1:PEN 3:PRINT:PRINT STRING
$(0,CHR$(225)):PAPER 0
3350 PEN 2
3360 PRINT" Mr Humpy is very sad.The
love of his life Esmerelda has been
captured and put in the horrible Bost
on-de-Stump."
3370 PRINT" It is your task to help
him run along the heavily guarded wal
l of the Stump and free Esmerelda f
rom her cell."
3380 PRINT" There are however some o
bstacles which are out to stop you. Th
ese include manic Frenchmen, magenta-c
oloured rocks and a barrel of TNT whi
ch is about to explode."
3390 PAPER 1:PEN 3:PRINT STRING$(40,C
HR$(225)):PAPER 0
3400 LOCATE 1,22:PEN 2:PRINT STRING$(
40,CHR$(131)):LOCATE 1,24:PRINT STRIN
G$(40,CHR$(140)):LOCATE 7,23:PEN 3:PR
INT"Press <SPACE BAR> to continue"
3410 IF INKEY(47)<>0 THEN GOTO 3410
3420 CLS
3430 PAPER 1:PEN 3:PRINT STRING$(40,C
HR$(225)):FOR f=1 TO 7:LOCATE 1,f:PRI
NT CHR$(225):LOCATE 40,f:PRINT CHR$(2
25):NEXT f:PAPER 0
3440 LOCATE 17,3:PEN 2:PRINT STRING$(
0,CHR$(131)):LOCATE 17,4:PEN 3:PRINT"
DA BELLS":PEN 1:LOCATE 17,5:PRINT STR
ING$(0,CHR$(140))

```



```

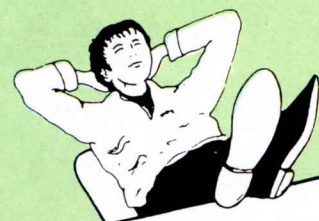
3450 PRINT:PEN 3:PAPER 1:PRINT STRING
$(0,CHR$(225)):PEN 2:PAPER 0
3460 PRINT" To complete each screen
you must ring the bell. If you can ri
ng it before the TNT explodes you wil
l be given a bonus. However if the TN
T explodes one of your lives will be
lost."
3470 PRINT" Once you have finished a
ll 15 screens a Super Bonus is given
and you have to start again. But thi
s time the fuse on the TNT will be b
urning a lot faster."
3480 PAPER 1:PEN 3:PRINT:PRINT STRING
$(0,CHR$(225)):PEN 1:PAPER 0
3490 LOCATE 1,22:PEN 2:PRINT STRING$(
40,CHR$(131)):LOCATE 1,24:PRINT STRIN
G$(40,CHR$(140))
3500 LOCATE 7,23:PEN 3:PRINT"Press <S
PACE BAR> to continue"
3510 IF INKEY$="" THEN GOTO 3510
3520 IF INKEY(47)<>0 THEN GOTO 3520
3530 IF INKEY$<>"" THEN GOTO 3530
3540 CLS
3550 PAPER 1:PEN 3:PRINT STRING$(40,C
HR$(225)):FOR f=1 TO 7:LOCATE 1,f:PRI
NT CHR$(225):LOCATE 40,f:PRINT CHR$(2
25):NEXT f:PAPER 0
3560 LOCATE 17,3:PEN 2:PRINT STRING$(
0,CHR$(131)):LOCATE 17,4:PEN 3:PRINT"
DA BELLS":PEN 1:LOCATE 17,5:PRINT STR
ING$(0,CHR$(140))
3570 PAPER 1:PEN 3:PRINT:PRINT STRING
$(0,CHR$(225)):PAPER 0
3580 PEN 3
3590 LOCATE 17,10:PRINT"THE KEYS":PEN
2:LOCATE 17,11:PRINT STRING$(0,CHR$(
131))
3600 PEN 3:LOCATE 14,12:PRINT"x....no
ve left"
3610 PEN 1:LOCATE 14,14:PRINT"z....no
ve right"
3620 PEN 2:LOCATE 14,16:PRINT"SPACE..
.to jump"
3650 PEN 1:LOCATE 9,18:PRINT"Z and SP
ACE to jump left"

```

```

3660 PEN 3:LOCATE 9,20:PRINT"X and SP
ACE to jump right"
3670 PEN 2:LOCATE 1,22:PRINT STRING$(
40,CHR$(154)):LOCATE 1,24:PRINT STRIN
G$(40,CHR$(154)):PEN 3:LOCATE 8,23:IN
PUT"SELECT START SCREEN (1-15)":chose
$
3680 IF VAL(chose$)<1 OR VAL(chose$)>
15 THEN LOCATE 1,23:PRINT STRING$(40,
" "):GOTO 3670
3690 LOCATE 1,23:PRINT STRING$(40," "
):LOCATE 11,23:PEN 2:PRINT"Press <SPA
CE> to Play."
3700 IF INKEY$<>"" THEN GOTO 3700
3710 RETURN
3720 REM*****
3730 REM*****CONGRATULATIONS*****
3740 REM*****
3750 LET bon=INT(RND*40)*10
3760 LOCATE 6,2:PEN 14:PRINT bon+500;
"BONUS "
3770 RESTORE 3030
3780 FOR F=1 TO 24:READ N:SOUND 4,N,2
0,15,1:NEXT F
3790 FOR F=1 TO 5000:NEXT F
3800 LET score=score+500+bon
3810 PEN 1:LET bonusd=bonusd-1:IF bon
usd=0 THEN LET bonusd=1
3820 RETURN
3830 DATA 60,53,47,45,60,0,45,47,45,4
0,53,0,53,47,45,36,40,40,45,45,47,53,
47,60,9999,9999

```



Give your fingers a rest . . .  
All the listings from this month's  
issue are available on cassette.

# Ready Reference: Graphics

**Get the facts at your fingertips with the first of our ready reference charts**

pen/paper	ink	colour
0	1	Bright Blue
1	24	Bright Yellow
2	20	Bright Cyan
3	6	Bright Red
4	26	Bright White
5	0	Black
6	2	Bright blue
7	8	Bright Magenta
8	10	Cyan
9	12	Yellow
10	14	Pastel Blue
11	16	Pink
12	18	Bright Green
13	22	Pastel Green
14	1,24	Flashing Blue / Bright Yellow
15	16,11	Flashing Pink / Sky Blue

Mode 0 defaults

Ink	Colour	Ink	Colour
0	Black	14	Pastel Blue
1	Blue	15	Orange
2	Bright Blue	16	Pink
3	Red	17	Pastel Magenta
4	Magenta	18	Bright Green
5	Mauve	19	Sea Green
6	Bright Red	20	Bright Cyan
7	Purple	21	Lime Green
8	Bright Magenta	22	Pastel Green
9	Green	23	Pastel Cyan
10	Cyan	24	Bright Yellow
11	Sky Blue	25	Pastel Yellow
12	Yellow	26	Bright White
13	White		

Ink colours

pen/paper	ink	colour
0	1	Bright Blue
1	24	Bright Yellow
2	20	Bright Cyan
3	6	Bright Red

Mode 1 defaults

pen/paper	ink	colour
0	1	Bright Blue
1	24	Bright Yellow

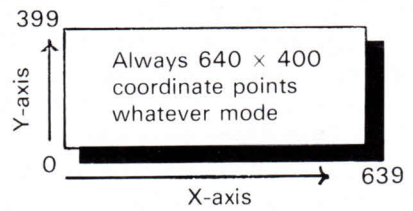
Mode 2 defaults

Mode	Number of characters	Number of colours
0	20	16
1	40	4
2	80	2

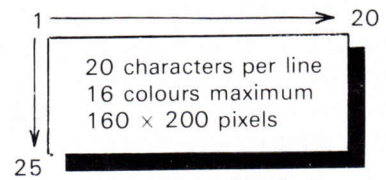
Mode characteristics

Mode	graphics coordinates		pixels	
	x axis	y axis	x axis	y axis
0	640	400	160	200
1	640	400	320	200
2	640	400	640	200

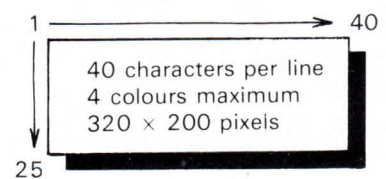
Graphics coordinates and pixels



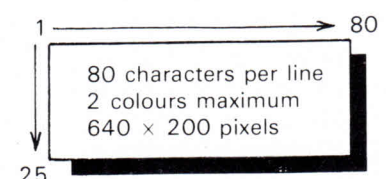
Graphics screen




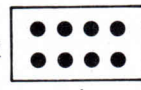




Text screen Mode 0



Text screen Mode 1



Text screen Mode 2

Mode	Character	Pixel
0	16  32	2  4
1	16  16	2  2
2	16  8	2  1

Characters and pixels measured in screen coordinates

# *Amstrad PC leak came via Taiwan*

An investigation by Computing with the Amstrad has revealed the source of the breach of security which surrounded the launch of the Amstrad PC.

This leak led to comprehensive details of the then top secret machine being published in leading trade magazine Microscope.

And as this took place six weeks before the official unveiling, it is known to have incensed company executives, in particular chairman Alan Sugar.

While Amstrad itself mounted its own enquiry into how the highly confidential material had been obtained, Computing with the Amstrad started a separate investigation.

Naturally Microscope chose to keep the names of its informants to itself.

The key to the mystery turned out to be one of the three people named as the team behind the front page article.

Bruce Everiss, a controversial industry character, is more associated with featuring in news stories rather than writing them.

The investigation revealed that it was Everiss who had

passed on the machine's specifications and a photocopy of an artist's impression to Microscope.

Where had Everiss got this highly confidential information? It transpired that it had come from a company itself very much involved in the clone manufacturing scene - Citadel Products.

And as it so happens, the firm for which Everiss works as a consultant. Victor Cedar, Citadel's boss, readily admitted that he was the UK source.

"We knew the specifications three months ago, he said. "But it was only when someone sent us a picture of the machine that we told anybody".

Cedar confided that the information had come to him from the Far East - not from Korea where the new machine is being made, but from Taiwan. It appears to have been originally gleaned by a components representative calling on a number of computer manufacturing plants in the region.

He picked up the specifications and the picture in Korea, then passed it on to interested parties in Taiwan.

The Taiwanese are known to

be closely monitoring what Amstrad is doing in Korea in case it eventually threatens their booming clone market.

"It's very difficult to keep secrets our there", says Victor Cedar. "And although we were apparently the first to hear the specifications over here, they had been

common knowledge in the Far East for quite some time".

As a parting shot Victor Cedar offered the latest tidbit from his contacts in the Far East. "I've heard that Tatung are going to bring out a clone of the PCW 8256 for £349" he said.

## **PRE-LAUNCH SCOOP**

This is what Microscope had to say about the Amstrad PC six weeks before its launch:

The machine, to be known as the PC1512, is powered by an 8MHz Intel 8086 CPU and will be sold with GEM and GEMPaint as well as MS-DOS 3.2. A mouse, monitor, serial and parallel ports (but no printer) will be included.

Four models will be announced, each with 512k of RAM, expandable to 640k, and a 16 colour IBM standard graphics adaptor (giving grey scale displays at 640x200 resolution on monochrome machines).

Model 1 priced at £399, is a single 5 1/4 inch 360k disc mono machine. Model 2, at £499 has an extra disc drive: models 3 and 4 at £549 and £649 are as models 1 and 2 but with a colour monitor. All machines have three

available full-size IBM bus slots, accessible through the top CPU unit.

Standard IBM cards, including network interfaces and hard cards will run.

The PC1512 has a small footprint, reminiscent of the Olivetti M19: 156 sq. in. compared with 256 for a standard IBM PC.

Amstrad chairman Alan Sugar has agreed to license 50,000 copies of MS-DOS a month, suggesting a sales target of 600,000 PC1512s by September 1987.

This is almost five times the number of IBM micros sold in the UK last year.

The machine will be endorsed by Lotus as fully 1-2-3 compatible in a £12.5 million ad campaign due to start in the first week of September'.

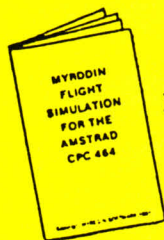


**3D LANDMARKS  
YOU CAN FLY AROUND**

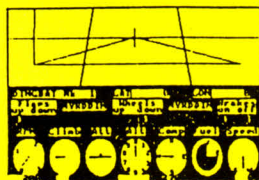
**SUPERB REAL  
TIME SIMULATION**

# MYRDDIN FLIGHT SIMULATION

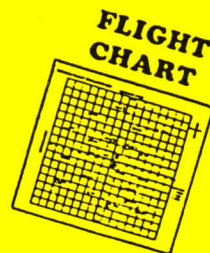
## AMSTRAD CPC 464



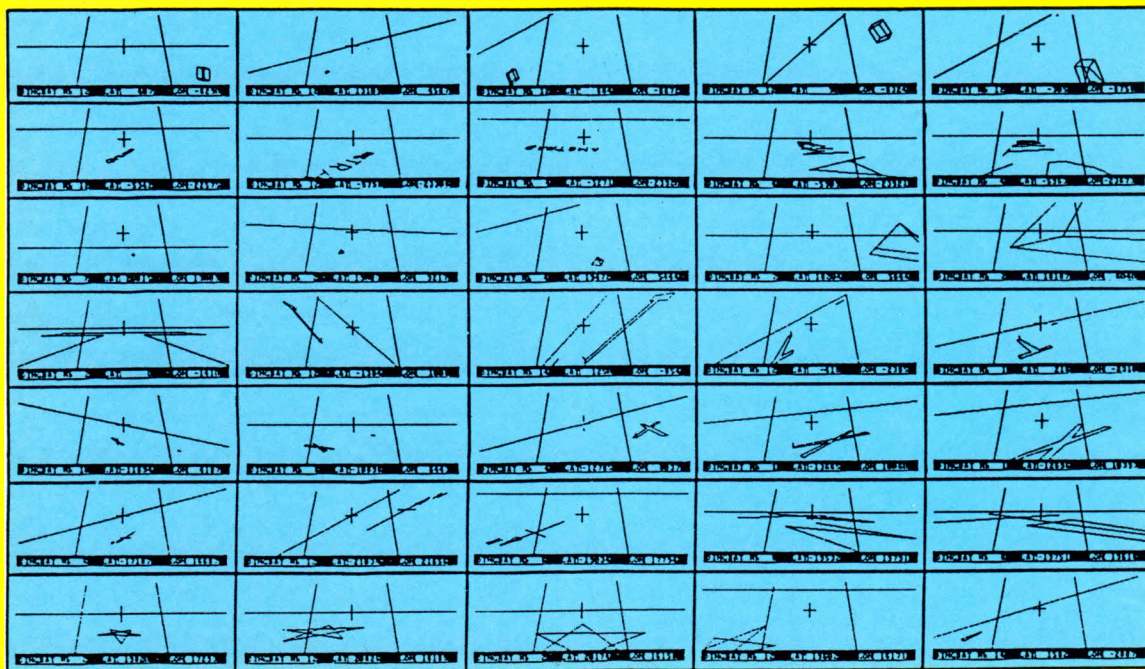
**MANUAL**



**FULL SCREEN  
DISPLAY**



Here are some screens from a typical flight showing the view from the cockpit (top half of screen) produced as printouts of the actual simulator.



A real time simulation with 3D graphics uses a massive 64000 x 64000 longitude & latitude flying area, making each flight completely different. Developed under pilot instruction to give realistic flight effect. The view through the cockpit gives moving 3D graphics.

Comprehensive instrument panel with moving needle meters & digital displays. 15 aircraft types with varying control sensitivities & speeds of between 100 - 500 knots.

3 runways available for refuelling, take off & landing. Ground and landmark orientation correct with all flying attitudes (rolls etc).

The 3D graphics are still accurate when you fly upside down.

3D landmarks you can fly around.

Comes complete with manual & fully detailed chart of landmarks & airfields.

Joystick or keyboard operation.

**STRATEGY SOFTWARE  
BOX 5000  
GLEN WAVERLEY  
VICTORIA 3150  
(03) 560 4324**

**TAPE \$17.95**

**DISK \$29.95**



## It's all go for the big Show

A record number of new products are set to be launched at the fifth Amstrad Computer Show, thanks in part to the new PC.

In all more than 200 innovations - close to double the previous all time high - will be unveiled at the three day event which opens at the Novotel, London on October 3.

And a survey has revealed that 30 per cent of these will be for the latest Amstrad machine.

The remainder of the new releases are reported to be almost equally divided between the PCW and CPC ranges.

Yet the feature attraction at this show - the exhibition floor space virtually being sold out months ago - seems likely to be the IBM compatible model.

Citadel Products has been quick off the mark - understandably based on the main news story - with a Winchester disc drive for the machine which comes complete with a plug in card for around £399.

One company alone, Digital Research, will be showing nine packages targeted at the new PC.

These include four well established titles - GEM Draw, GEM Write, GEM Graph and GEM Word/Chart - all priced at £99.95.

In addition, the software giant will have five new products available for the PC. These are GEM Diary - a page per day diary complete with notepad, 11 alarms and card index file costing £39.99; GEM Fonts and Drivers Pack and GEM Draw Business Library, both priced at £39.95; GEM Font Editor at £99.95; and GEM Programmers ToolKit at £175.95.

Caxton Software is also aiming at the IBM clone with a total of six packages.

Cardbox, Scratchpad, Brainstorm and Touch 'N Go have all been available before for other machines but Condor Junior and Timekeeper are first time launches.

Condor Junior which is an enhanced version of Condor 1, a relational database and report system suitable for novices, costs £99.99. Timekeeper, a version of which will also be for the PCW, is a time management system costing £49.99. On the PCW front Compact Software will be unveiling its extended business range including Financial Reporter priced at £49.99.

First Calc, the latest module in Minerva's Learner Series for the PCWs will be released at the show. An

introduction to spreadsheets, its price has yet to be fixed.

Minerva will also be launching instant Access and Random Access. The first is a program writing utility priced at £29.95, the second a database costing £39.95, both for the PCW range.

CPC users will not be taking a back seat either. New from CDS is Colussus Bridge 4.0 which comes with a copy of the tutor book "Begin Bridge" by expert G.C.H. Fox. Features include hand input facility, cheat options and variable speed of play. The price is £11.95 on tape, £14.95 on disc.

Advantage is set to launch its Professional C Compiler for the CPC models. This offers floating point arithmetic to 16 points and costs £35.50.

The company is also unveiling Nemesis, a four-in-one games package for both the CPC and PCW ranges, which is available on disc priced at £13.95.

"With more new products than ever before, this show will once again provide the mirror image for Amstrad's continuing success story", says Derek Meakin, head of organisers Database Exhibitions.

## LISTINGS GO ON-LINE

All program listings in Computing with the Amstrad are now available for free downloading on MicroLink, the UK's fastest growing electronic mail service. They join hundreds of programs already available on Britain's national on-line database.

## Branching out

Two subsidiary labels have been created by Ariolasoft as an outlet for its increased Amstrad product range.

Primarily known for marketing Stateside packages, Ariolasoft is now focusing on its independently sourced software.

Arcade/action games will be available on Reactor, and strategy/adventure games on 39 Steps. Programs under licence and in the business range will continue to appear under Ariolasoft.

Set for autumn release on the CPC range are Deactivators on Reactor and They Stole a Million on 39 Steps.

The first promises to be full of explosive action, while the second involves planning a perfect raid which is then carried out in arcade sequence.

Both cost £14.95 on disc and £9.95 on cassette.

Have you subscribed yet?

See our special offer on Page 56

## TRANSATLANTIC LINK FOR MICROS OPENED

MicroLink has achieved a world first by creating a permanent electronic link across the Atlantic. It will allow owners of Amstrads and other micros to log on in a matter of seconds to one of the most dynamic databases in the USA.

And the cost will be approximately half that of a conventional transatlantic telephone call.

The breakthrough has come about as a result of a deal struck between MicroLink and Mnemetics based in Sparkill, New York.

With the help of its US partner, MicroLink has set up a complex electronic gateway which will enable the exchange of messages between micros in the US and the UK.

The new service was officially opened when Derek Meakin, head of MicroLink, sent the first message to his opposite number at Mnemetics, Gregg Squires.

Seated at the keyboard of his office in Stockport, he simply typed the letters MNE. Without any delay a welcome message from New York appeared on his screen. Then he typed:

"Dear Gregg - This is a historic moment, the opening of the first-ever transatlantic gateway to enable micro users in the UK to talk to their fellow enthusiasts in the USA.

"On behalf on many thousands of MicroLink subscribers in the UK, together with many others who have joined us from countries all round the world. I send greetings to you and everyone at Mnemetics.

"From now on we're all going to have lots to talk about".

The UK-USA computer link

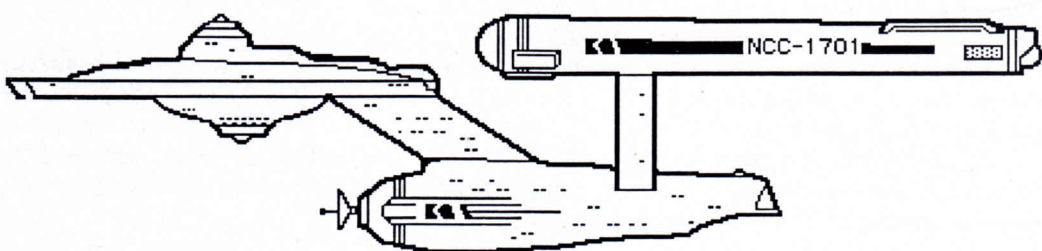
opens up a host of new services for subscribers to MicroLink.

They will be able to choose from playing real-time games to participating in the activities of more than 40 special interest groups.

And with Associated Press providing a 24 hour global news service, there will be no need to wait for the radio news.

"We are just delighted that we at MicroLink are able to offer this first", says Derek Meakin. "It's going to transform the way we communicate with our cousins over in the States".

Now Gregg Squires hopes to open a reverse link - to allow micro users in the USA to share all the facilities of the MicroLink.



### BEYOND BOLDLY GOES

An exclusive licensing contract has been won by Beyond Software to boldly go where no manufacturer has been before.

It has been commissioned by Paramount to produce the first official arcade/strategy game of cult TV series Star Trek to coincide with its 20th anniversary.

Mike Singleton, author of the multi-award winning Lords of Midnight, heads the team designing and creating the game.

Star Trek - the game - will be on a huge scale, with 1,000 star systems. Other

features include 3D vector and solid graphics, and 256 subplots.

Players take on the roles of the seven leading protagonists, including Captain Kirk, Dr. Spock and Scotty.

The main task involves foiling the relentless Klingons' latest bid for universal dominance.

They have created a psymitter to control the minds of all those in the Kleinsphere, so this area must be immunised to avert the threat of a complete take over.

### NOTE RE: U.K. PRICES

Australian and New Zealand readers would be aware that local currencies have been subject to severe fluctuations of late against both £Stg. and the U.S.\$.

Rather than attempt to convert all prices shown in the U.K. News section of the magazine, we would advise readers to use a multiplication factor of between 3 and 5 to calculate local prices.

Generally speaking, the lower the U.K. price the higher the conversion factor. This is because some importation costs are the same for items with vastly differing prices. Also, don't forget to add 15% where prices are shown excluding the U.K. consumption tax, VAT.

## Shattered Mrs T

In Split Personalities from Domark Mrs. Thatcher has come unstuck and you have to pick up the pieces.

This latest release for the CPC range is a jigsaw game featuring a variety of famous people, among them Amstrad's Alan Sugar.

Points are scored by assembling a full picture within a frame. One distraction is a flying Denis Thatcher and others are in the form of bombs.

All 10 screens feature a hi-res and colourful representation of a public eye figure. Available on cassette for £8.95 and disc £14.95

## Virgin re-run

The Virgin Atlantic Challenger is repeating its record breaking transatlantic run, but this time the quest for the Blue Riband is on the CPC range.

Playing Captain Richard Branson, you must keep as high a speed as possible without veering off course and falling foul of the obstacles that lie in your way.

An added distraction is the need to make crucial business decisions while at the helm.

Virgin Atlantic Challenger Game is available on cassette from Virgin Games for £8.95.

# Amstrad will dominate UK market - US report

A leading US market research company has forecast that Amstrad is set to dominate the UK home computer market.

The International Resource Development Corporation claims that from now on the pacesetters in the UK micro industry will establish the de facto standards.

In a recently completed report on what's happening on the UK home computer scene, IRD concluded that no one could stand in the way of "mighty Amstrad". Any newcomers to the market face

an uphill struggle according to the Americans.

However one of the US research team suggests that Amstrad's dominance will offer at least one beneficial result.

"It will reduce the risk of marketing add-on and software products since these products will not have to be produced in multiple formats to fit several different kinds of home computer", says Lawrence D. Gasman.

The report goes on to predict that thanks mainly to Amstrad some £35 million

worth of peripherals will be sold to home computer users in the UK this year.

But this figure is likely to remain static for the next five years.

Software sales on the other hand are likely to grow from £40 million in 1986, climbing steadily to a peak of £50 million by 1990.

"Computers are no longer a novelty in the UK", writes Gasman, "but for the time being serious uses are confined in the main to word processing".

## Alien threat

Mission Omega is now available for the CPCs 464 and 6128 from Mind Games. The player has only one hour to prevent a radioactive alien annihilating civilisation.

A selection of robots, each with unique characteristics, must be programmed to complete a reactor shutdown.

They can be manoeuvred through a preplanned computer program or via the real-time manual control commands.

Price: £9.95 on cassette, £12.95 on disc.

## IN BRIEF

The follow-up to Ocean's Hunchback is Hunchback, the Adventure, a three part graphic and text adventure for the CPC range.

The twin tape pack costs £8.95 and disc £14.95.

A version of the classic arcade game Space Invaders has been released for the PCW range from Supernova.

Invaders, featuring hi-res graphics and sound, costs £12.95.

\*\*\*

\*\*\*

PCW 8256 owners can now play Graham Gooch's Test Cricket from ASL. Compatible with DK'Tronics and Cascade joysticks, it costs £19.95.

\*\*\*

Look for more of the very latest news from the U.K. next month, brought to you within days of publication in the U.K.



# Calling all cheats!

**A** PROGRAM that will dump the contents of a binary file to the screen has been sent in by Michael Wong. This is an excellent way of cheating in adventures as it allows you to view all the words and responses in the program.

I hereby nominate Mike for the Hall of Fame. If any other readers have any short programs that they think would help fellow adventurers send them in and if they are printed I will send them a free adventure.

Don't forget to enclose a list of the adventures you already own so that I know what not to send you. Send me your list, Mike, so that I can get a tape in the post to you.

The response to the competition for names for the Sorcery high score table has been excellent. The winner, with 26 suggestions, was John Saker.

Also worth an honourable mention are Richard Levy and Geoff Dawson. Geoff came up with Clive, which was the only one that John didn't get.

The names to try are: Strangeloop, Virgin, Sorcery or Sorcery+, Amstrad, Arnold, Roland, Hugh, Brynn, Elaine,

## Adventuring with Gandalf

Nick, Angela, Jeremy, Patricia, Steve, Simon, Ian, Andy, Dave, Charles and Clive.

In case you are wondering where the rest of the names are, entering no name at all will produce an amusing response, as will several words that I dare not print.

I'd like to thank several people for sending in maps and solutions to

various adventures: P. Robson and John McCann for Dun Durach, Linda Wright for Smuggler's Cove (good luck in the new house Linda), and Di Reynolds for Gems of Stradus and Return to Eden.

Di wants to know whether Classic Adventure and Colossal Adventure are the same. They are both based on Colossal Caves, the original adventure, but have quite a few differences, the main one being that the Level 9 version has an end-game whereas Melbourne House's does not.

There are also quite a few differences in the play of the game. To get past the snake for instance, only requires the bird in Colossal whereas the flute and bird are needed in Classic.

```

10 REM Mike Wong's cheat
20 REM (c) Computing with the Amstrad
30 REM Needs start location and length of program
40 MODE 1:INK 0,0:INK 1,20:INK 2,16:INK 3,26:BORDER 0
45 margin% = VAL(margin$)
50 PEN 1:PRINT "Enter location ";:PEN 2:INPUT a
60 PEN 1:PRINT "Enter length ";:PEN 2:INPUT b
70 MEMORY a-1
80 PEN 3:LOAD "filename"
90 CLS:MODE 1:PEN 1:PRINT "Vocabulary - (To restart type GOTO 80)":PEN 3:PRINT CHR$(7):FOR z=a TO a+b
100 IF PEEK(z)>31 AND PEEK(z)<129 THEN PEN 3:PRINT CHR$(PEEK(z));:GOTO 120
110 IF PEEK(z)<32 THEN PEN 2:PRINT "*"
120 NEXT:PRINT CHR$(7):PRINT:PRINT:PRINT:PRINT:PRINT "Again?":PEN 2:INPUT q$
130 q$=lower(LEFT$(q$,1))
140 IF q$<>"y" THEN CLS:STOP
150 GOTO 80

```

Adventure cheating program

**LORDS of Midnight** from Beyond Software is more than just an adventure or war game, it is an epic.

The land of Midnight has been almost overrun by the evil Doomdark. Through his control of the ice crown he has covered the land in permanent winter and now it is the solstice, the height of winter and Doomdark is preparing his armies for the final battle with the Free.

You play Luxor, Lord of the Free, and with you are three companions,

Corleth, Rothron and Morkin, your newly-discovered son.

As you are the last of the moon princes you have the moon ring, a magical object that allows you to see through the eyes of your allies and control their actions.

You have two different ways of gaining a victory over Doomdark. You can meet and defeat him on the battlefield and then capture his citadel, Ushgarak, or you can guide Morkin to the Towers of Doom and

steal and destroy the ice crown, the source of Doomdark's power.

To play the game to the full means that you must send Morkin on the quest as well as attempt to defeat Doomdark. Either can be done separately, and until you get used to controlling the characters you would be well advised to do just one or the other.

To enlist lords and their armies to your cause you or one of your allies must visit the keep or citadel where they live. Once a lord has been recruited he comes under the control of Luxor.

As far as I have been able to discover there are 25 such lords, which makes keeping track of who you are controlling at any given time a task in itself.

For Doomdark to win he must kill Morkin and take Xajorkith, the citadel of the Free. As long as Morkin is alive, the game will continue. This means that Xajorkith can either be defended or abandoned in favour of a better strategy.

I have found two ways of winning by battle. Gather as many lords and their armies as possible and take them to Xajorkith. Doomdark's armies will arrive eventually and, since they don't wait for reinforcements but attack immediately, you can defeat every army as it arrives.

The other way is to take your armies to Ushgarak. If you choose an indirect route you should find that the majority of Doomdark's armies are too far away to help him. In either case, Morkin should be hidden away in a keep.

I have yet to take Ushgarak before Xajorkith has fallen, and would be interested to know if this can be done.

It is possible to concentrate solely on Morkin's quest and leave Luxor and his armies to fend for themselves.

Since this is the adventure part of the game I won't give too many hints. It is possible, despite the relevant part of the documentation saying otherwise, to get an army to help Morkin. If the army is in front of him it isn't with him.

Don't go too far to the North with this army though, or you will attract too much attention.

The game is controlled by single key input. Characters and their actions are selected directly from the keyboard or via a menu. Each

## BUG HUNTER

I'VE instituted this section to report any bugs that are found in adventures.

Di Reynolds reports that there is one in **Classic Adventure**. The game has a feature that scrolls the screen whenever your inventory is too large or the objects in a location are too numerous to fit on one screen.

Usually pressing the

**Spacebar will scroll the screen down. Di says that whenever she visits a location containing a lot of objects the paged mode comes on but on pressing the Spacebar the game prints a list of all the objects in the game and then crashes, causing her to reload the game from the start.**

**You have been warned, so bear it in mind.**

## SOS

**SORCERY+** is causing problems for Mike Singleton and John Saker. Mike wants to know how to surround the necromancer completely, as he can only find three hearts and a disguised unidentifiable object. And they both want to know how to get practice mode in Chapter 2. Can anyone help?

Iain Gardener is baffled by **Devil's Crown**. He wants to know how to get started and what to do.

So do I!

I received a copy of this superb game for review but gave it back as I considered it to be more of an arcade game. Several readers have written in asking for help with it since and now I wish I'd kept it. Can anyone help Iain?

Richard Levy wants to know how to free the sorcerer in the wine cellar in **Sorcery**. Is this the one you need the scroll for?

## FEEDBACK

**ANDREW Busby** has written in to say that, although he may lose some lives, **H. Woods** should climb the wall to avoid the bird in **Jet Set Willy**. Surely there must be a way to do this without losing lives?

**Linda Wright** says that **Dave Carr's** problem in **Smuggler's**

**Cove** can be solved if you **DIG WITH SPADE**.

**Di Reynolds** has answered **Mark Scofield's** problem with **Knightlore**. To give objects to the wizard he should jump on to the cauldron and then drop the object.

**I hope these hints will help.**



character has a range of options at each location.

All the lords and their armies perform according to their strength, which varies from utterly invigorated to exhausted. In addition, because of the ice fear that Doomdark spreads across the land, you may find that some of your armies are too scared to fight.

Each location a character visits is displayed graphically and a view may be obtained for each direction. Since there are about 4,000 locations this means that there are a massive 32,000 possible pictures in the game.

Characters and armies that are to be found in the location that would be visited if a move is made are also displayed, yet the graphics are drawn instantly.

Each character has a limited number of moves he can make before nightfall, when he can move no longer. The number of moves varies with the type of terrain and is decided by the computer.

When night falls it is Doomdark's

turn to go into action.

The complexities of the game are such that the booklet accompanying it contains 28 pages long, half devoted to explaining the background to the game and giving valuable insights into the way the characters behave.

One slight niggle is that after saving your position on tape the program tries to verify it.

Apart from not prompting you to rewind the tape, if for any reason it can't verify the saved game you will have to reload and start from the beginning.

This can be very awkward, especially when, like me, you lose the results of over three months successful campaigning.

I was unable to get a disc copy for review so am unable to say whether the problem is confined to the tape version.

I simply cannot praise this game too highly. The more I play the more I find to enjoy in it. It is the best game I have ever played!

## HINTS DEPT

**IN *Jewels of Babylon* John McCann can't find the rod to open the vertical stone slab behind the waterfall, and Barry Murphy can't open the door in the dusty corridor.** The rod is on the other side of the door and you will need to cross the pit to get the key to it.

**One problem in *Fantasia Diamond* seems to come up every month – how to get out of the dingy cellar. Joanne Bradburn, Lynne Fairley and E.C. Harvey are all stuck there.** Drink the wine and then rub the glass.

**Colin Todd is having trouble negotiating *Marl's Gateway in Forest at World's End*.** Try going N, E, N, N from Wildmoor Woods.

**Sarah Jane Lambert is stuck in the fog in *Mordon's Quest*.** From the hall go S, S, W, N and when the torch is lit, N again. Incidentally, thank you for the map of *Gems of Stradus*.

***Colossal Adventure* is posing some problems for**

**Matthew Phillips.** The pirate's chest is in the "All the same maze". From the west end of the Hall of Mists go S, E, S, S, E, N, E to the orange column then E, NW to the chest. Feed the sandwiches to the bear.

**Matthew Johnsterne is stuck in *Red Moon and Return to Eden*.** The fan is past the fountain abyss. Drop the chalk to neutralise the pool of acid. Drop the sweet pea where the bird can find it and enter the resulting edifice. Use what you find and the leviathan will cease to bother you.

**V. Lowe and Di Reynolds are in difficulties with *Classic Adventure*.** The trident is a treasure and you can also use it to get the pearl from the clam. Say plover in the plugh room. Get the jewellery south of the snake location. Don't forget the rug after you have killed the dragon. There are two mazes, one with the vending machine (a red herring!) and one with a valuable treasure chest.

## WHAT NOW?...

THIS part of the column was prompted by a plea for help from Neil Robertson, who wants to know what he is supposed to do when he gets the "What now?" prompt in Mayday, and M.R. Pugh who is in similar difficulties with Forest at World's End.

Since there must be many Amstrad owners equally baffled I shall spend some time over the next few months getting the complete beginner going and hopefully explaining to non-adventurers what all the fuss is about.

If you have never played an adventure before, let me warn you now that they are very addictive and likely to result in you losing a lot of sleep.

An analogy I find useful is to compare them with detective stories. The author creates a story that has the hero starting with little or no idea how to arrive at a solution and ends with him unmasking the culprit.

Along the way he will find red herrings designed to throw him off the scent and valuable clues that help him to a solution. Objects in the locations he visits can sometimes help him to progress further.

A simple example would be that of finding a key that opens a safe. The safe may hold a will that provides a clue as to who the culprit is.

An adventure follows much the same pattern, except that you can't turn to the last pages to find out how the book ends. Instead you play the detective and seek the clues, using them to solve the game.

Since most adventure writers try to make the games as hard as possible you will find you need to adopt a lateral thinking attitude.

In the example above, you may have had to dig in the garden with a shovel to find the key. The shovel may have been in a boarded-up cellar that you needed to use an axe to get into. The axe might have been left up in a tree and you may have needed a ladder to get it, and so on.

Moreover, the scenario may be set in the far future and your purpose may be to repair a spaceship in order to deliver a cargo, as in Mayday, rather than solve a crime. Here you not only have a different set of problems but a vastly different technology.

Alternatively, it might be set in the distant past or a fantasy land where objects have special properties, such as a wand that, when waved, magically creates a bridge across a ravine.

Whatever the setting, they all have one thing in common - you have to think yourself into the role the writer has put you in and into the scenario he has created.

As in the real world, you have to move around and examine your surroundings. Locations in an adventure are usually called rooms, whether they are the rooms of a house or quicksands on a desolate moor, and you move around as if you were doing so in real life.

Since you are interfacing with a computer, you tell it which direction you want to move in. The micro will then move for you and tell you what you can see.

Obviously, you can't move everywhere in real life, and the same applies with adventures, so you are usually told which directions it is possible to move in.

Movements are usually made by typing in commands like GO NORTH. Most adventures allow you to abbreviate this to NORTH or even N.

Assuming it is possible to move in the direction you have chosen, you will find yourself in a new room and the screen will be updated to give you what information is available about it.

After giving several such commands you may well find that you have forgotten how to get back to where you started. A map is the answer, but you have to make your own.

Whenever I load in a new adventure the first thing I do is to explore every location open to me, making a map as I go along.

For each location I visit I draw a box and mark in it anything I have found. I also write in a description that identifies it from the other rooms, and draw lines to indicate the directions in which I am told I can travel.

This is very useful, as when I am deep into the game, I can check back to make sure that there aren't any rooms left unvisited.

Eventually I have a map of all the locations open to me and have met

several problems that need to be solved.

At this point I usually look at the objects I have come across and try to relate them to the problems I have to solve.

Next month I shall look at some of the objects commonly found in adventures and the kind of use they can be put to.

In the panel below I have listed the most common commands and an example of their use. Where abbreviations are normal, I have put them in brackets.

**GET object:** Used to pick things up - GET KEY.

**DROP object:** Drop an object already picked up - DROP KEY.

**EXAMINE object (EXAM):** Used to determine if there is anything special about a room or object. Quite often the object will need to have been picked up first, as in the following dialogue:

**EXAMINE BED** - 'You see a pillow'.

**EXAMINE PILLOW** - 'You haven't got it'.

**GET PILLOW**

**EXAMINE PILLOW** - 'You see a piece of paper sticking out of the pillow that says: Get Computing with the Amstrad regularly'.

**INVENTORY (INVE, INV, I):** Used to display a list of the objects you are carrying.

**LOOK:** Quite often the room description scrolls off of the screen while you are doing something with any objects you have found. LOOK is usually used to redescribe the room, though sometimes it can act as an alternative to examine.

**HELP:** What it says! Most adventures will allow you to ask for help, though the response might be even more cryptic than the problem.

**SCORE:** Useful for finding out how well or badly you are doing.

**N,W,S,E,U,D:** Abbreviations for the compass points and up and down. A few games also allow you to go southeast and so on. In these cases you can use SE, SW, NE, NW.

# All the gen on character generation

**C**HARACTER Generator is a useful utility that allows you to re-design any individual character to your own specifications. In fact you can create your own custom made character set.

Facilities are included for rotating, inverting and vertical or horizontal mirroring your characters. The data for your design can be output to the

screen or printer, and saved or loaded from cassette or disc.

The program displays an 8 x 8 grid containing a flashing cursor in its top left hand corner. You design your characters by filling in the relevant cells of this grid using a joystick or the cursor keys to move the flashing cursor.

The current position can be set using the fire button or Copy key. Pressing once more will clear that cell.

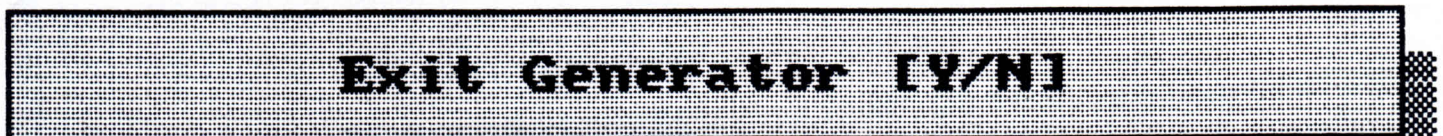
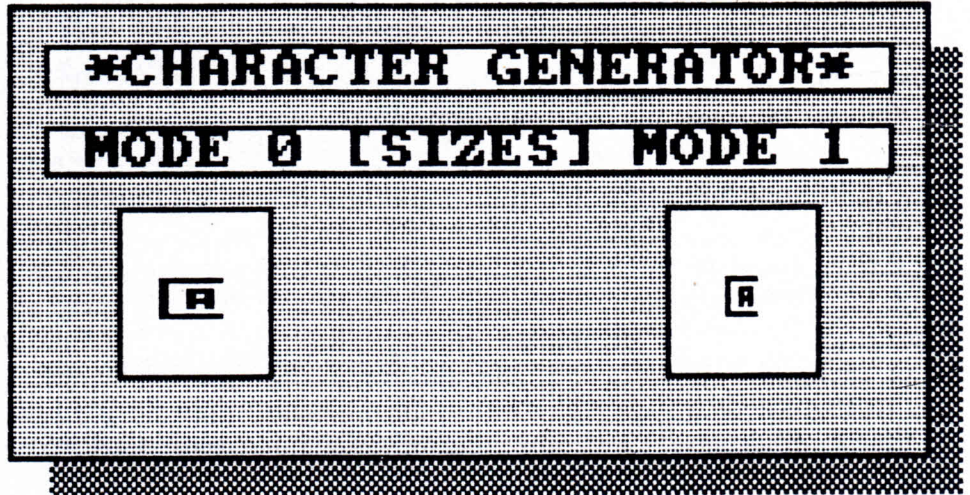
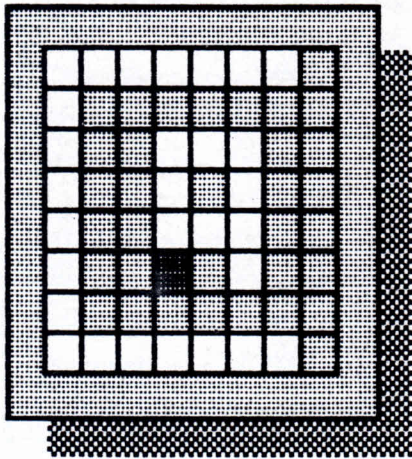
As you build up your character it will be duplicated in the two small boxes to the right of the grid. These show you how the character appears "life-size" in Mode 0 and Mode 1.

When you have finished your design, using the [S]tore option (by pressing 'S') will safely store the design in memory. Your design will now replace the default character of the same number. You can also save the character set on tape or disc.

Make sure you've stored your character in memory before trying to save it.

When saving a character set the

**RICHARD CROSKELL shows how to create your own custom built character set**





generator stores it as a file on cassette. However this file contains more data than just the new character set. If you want to create an appropriate file for inclusion in your own program you must follow this three step procedure:

- Reset the micro using Esc/Ctrl/Shift.
- Load the program in which you wish to use the character set and make its first line:

```
10 CLS: SYMBOL AFTER 32: LOAD"! ",
42240
```

then save the amended program and don't rewind the tape - we'll call this the master tape. You're going to put the data on the tape *after* this program.

- Now insert the tape that contains

the file of characters saved from the generator and run a program consisting of the lines:

```
10 CLS:SYMBOL AFTER 32: LOAD"! ",
42240
20 SAVE "filename",b,42240,1792
```

This will load the file into memory. When you get the prompt to press record and play, put back the un-rewound master tape in the cassette before proceeding. This will then save the character file where you want it.

The options listed below give a lot of flexibility when designing characters and are displayed continuously in the centre of the screen for quick reference. To select any option simply press the corresponding key.

- [M]irror - Allows the contents of

the design grid to be mirrored about its horizontal or vertical axis. Select [V] or [H] when prompted.

- [R]otate - Rotates the contents of the grid by 90 degrees.

- [I]nverse - Inverses the grid contents, i.e. any cell that was set will be clear and vice versa. In more simple terms it produces a negative of the original contents.

- [C]lear - Clears the design from all three boxes. Don't forget to store the contents if you want to save the design, otherwise the data will be lost when this option is used.

- [V]iew - Allows you to retrieve any of the characters (codes 32 - 255) from memory and place its form on the design grid for examination or modification.

- [D]ata - Produces the eight character bytes in decimal form, to be directed to the [S]creen or [P]rinter when prompted. Further prompts, [C]haracter, [N]ext and [F]inish allow you to select the order in which you wish to view them.

## VARIABLES

- gr% Grid square condition (1=set 0=Clear).
- ro% Condition after rotation (1=set 0=clear).
- re% Check for redefined character (1=Yes 0=No).
- by% Eight bytes that form character matrix.
- gx% Horizontal position of cursor.
- gy% Vertical position of cursor.
- mx1% Mode 0 character horizontal plot start position.
- mx2% Mode 1 character horizontal plot start position.
- my1% Mode 0 character vertical plot start position.
- my2% Mode 1 character vertical plot start position.
- f Flag for pen in design grid.
- p Flag for pen in true size grid.

## PROGRAM STRUCTURE

- 10-25 Title and credits.
- 30-50 Disable Break, select mode, colours, and set up variables.
- 55-90 Display flashing cursor, wait for key press.
- 95-150 Move flashing cursor in appropriate direction.
- 155-165 Cursor state (Set/clear).
- 170-200 Rotate option.
- 205-225 Invert option.
- 230-285 View option.
- 290-305 Clear option.
- 310-425 Cassette/Disc Save and Load.
- 430-480 Store option.
- 485-500 Exit program?
- 505-550 Mirror option.
- 555-690 Data option.
- 695-705 Get character matrix into array by%.
- 710-865 Poke machine code, define windows, graphics and set up screen.
- 870-875 Machine code data for character transfer routine.

```
10 REM *****
15 REM * * CHARACTER GENERATOR * *
20 REM * BY R.J.CROSKELL *
22 REM *(c)Computing with the Amstrad*
25 REM *****
30 MODE 1:INK 0,10:INK 1,0:INK 2,20:INK
K 3,15,1:PAPER 0:BORDER 10
35 CALL &BB40:IF PEEK(38500)=255 THEN
45 ELSE POKE 38500,255
40 SYMBOL AFTER 32:MEMORY 37999:SPEED
INK 15,15
45 DIM gr%(8,8),ro%(8,8),re%(224),by%(
8):gx%=1:gy%=1:mem%=&9470
50 f=0:p=0:mx1%=286:mx2%=534:my1%=278:
my2%=278:KEY 140,"":GOSUB 710
55 PEN #5,3:LOCATE #5,gx%,gy%:PRINT #5
,CHR$(255);
60 k$=UPPER$(INKEY%):IF k$="" THEN 60
65 IF k$=CHR$(240) OR JOY(0)=1 THEN 95
70 IF k$=CHR$(241) OR JOY(0)=2 THEN 11
0
75 IF k$=CHR$(242) OR JOY(0)=4 THEN 12
5
80 IF k$=CHR$(243) OR JOY(0)=8 THEN 14
0
```

```

85 IF k$=CHR$(224) OR JOY(0)=16 THEN 1
55
90 ON INSTR("RIVCTSEMD",k$) GOTO 170,2
05,230,290,310,430,485,505,555:GOTO 60
95 IF gy%=1 THEN PRINT CHR$(7);:GOTO 6
0 ELSE LOCATE #5,gx%,gy%
100 IF gr%(gx%,gy%)=1 THEN PEN #5,0 EL
SE PEN #5,2
105 PRINT #5,CHR$(255);:gy%=gy%-1:GOTO
55
110 IF gy%=8 THEN PRINT CHR$(7);:GOTO
60 ELSE LOCATE #5,gx%,gy%
115 IF gr%(gx%,gy%)=1 THEN PEN #5,0 EL
SE PEN #5,2
120 PRINT #5,CHR$(255);:gy%=gy%+1:GOTO
55
125 IF gx%=1 THEN PRINT CHR$(7);:GOTO
60 ELSE LOCATE #5,gx%,gy%
130 IF gr%(gx%,gy%)=1 THEN PEN #5,0 EL
SE PEN #5,2
135 PRINT #5,CHR$(255);:gx%=gx%-1:GOTO
55
140 IF gx%=8 THEN PRINT CHR$(7);:GOTO
60 ELSE LOCATE #5,gx%,gy%
145 IF gr%(gx%,gy%)=1 THEN PEN #5,0 EL
SE PEN #5,2
150 PRINT #5,CHR$(255);:gx%=gx%+1:GOTO
55
155 IF gr%(gx%,gy%)=1 THEN gr%(gx%,gy%
)=0:p=0:ELSE gr%(gx%,gy%)=1:p=1
160 PLOT mx1%+(gx%*4)-2,my1%-(gy%*2),p
:PLOT mx1%+(gx%*4),my1%-(gy%*2)
165 PLOT mx2%+(gx%*2),my2%-(gy%*2):GOT
0 60
170 FOR t%=1 TO 8:FOR g%=8 TO 1 STEP -
1:c%=ABS(g%-9)
175 ro%(t%,c%)=gr%(g%,t%):NEXT:NEXT
180 FOR t%=1 TO 8:FOR g%=1 TO 8:gr%(t%
,g%)=ro%(t%,g%)
185 IF gr%(t%,g%)=1 THEN PEN #5,0:p=1
ELSE PEN #5,2:p=0
190 LOCATE #5,t%,g%:PRINT #5,CHR$(255)
;:PLOT mx1%+(t%*4)-2,my1%-(g%*2),p
195 PLOT mx1%+(t%*4),my1%-(g%*2):PLOT
mx2%+(t%*2),my2%-(g%*2):NEXT:NEXT
200 gx%=1:gy%=1:GOTO 55
205 FOR t%=1 TO 8:FOR g%=1 TO 8:LOCATE
#5,t%,g%
210 IF gr%(t%,g%)=1 THEN gr%(t%,g%)=0:
f=2:p=0:ELSE gr%(t%,g%)=1:f=0:p=1
215 PEN #5,f:PRINT #5,CHR$(255);:PLOT
mx1%+(t%*4)-2,my1%-(g%*2),p
220 PLOT mx1%+(t%*4),my1%-(g%*2):PLOT
mx2%+(t%*2),my2%-(g%*2)
225 NEXT:NEXT:gx%=1:gy%=1:GOTO 55

```

```

230 CLS #2:LOCATE #2,2,2:INPUT #2,"Vie
w which character [32-255]-",char$
235 IF VAL(char$)<32 OR VAL(char$)>255
THEN 230 ELSE CLS #2
240 v%=VAL(char$):v%=(v%-32)*8:FOR t%=
0 TO 7:by%(t%+1)=PEEK(39000+v%+t%):NEX
T
245 FOR t%=1 TO 8:r%=1:FOR g%=7 TO 0 S
TEP -1
250 IF by%(t%)=2^g% THEN 260
255 gr%(r%,t%)=0:GOTO 265
260 by%(t%)=by%(t%)-2^g%:gr%(r%,t%)=1
265 r%=r%+1:NEXT:NEXT:FOR t%=1 TO 8:FO
R g%=1 TO 8
270 IF gr%(t%,g%)=1 THEN PEN #5,0:p=1:
ELSE PEN #5,2:p=0
275 LOCATE #5,t%,g%:PRINT #5,CHR$(255)
;:PLOT mx1%+(t%*4)-2,my1%-(g%*2),p
280 PLOT mx1%+(t%*4),my1%-(g%*2):PLOT
mx2%+(t%*2),my2%-(g%*2):NEXT:NEXT
285 gx%=1:gy%=1:LOCATE #2,9,2:PRINT #2
,"** Information area **":GOTO 55
290 LOCATE #5,1,1:PEN #5,2:PRINT #5,ST
RING$(64,255);
295 FOR t%=6 TO 9:LOCATE #4,4,t%:PRINT
#4,STRING$(4,32);
300 LOCATE #4,19,t%:PRINT #4,STRING$(4
,32);:NEXT
305 ERASE gr%:DIM gr%(8,8):gx%=1:gy%=1
:GOTO 55
310 CLS #2:PEN #2,1:LOCATE #2,2,1:PRIN
T #2,"Are you sure you have [S]tored a
way"
315 LOCATE #2,2,2:PRINT #2,"the design
, after you finished it ?"
320 LOCATE #2,10,3:PRINT #2,"Yes or No
(Y or N)."
325 k$=UPPER$(INKEY$):IF k$="Y" OR k$=
"N" THEN 330 ELSE 325
330 ON INSTR("YN",k$) GOTO 335,425
335 CLS #2:LOCATE #2,4,2:PRINT #2,"Sav
e or Load Characters (S or L)"
340 k$=UPPER$(INKEY$):IF k$="S" OR k$=
"L" THEN 345 ELSE 340
345 IF k$="S" THEN 390
350 CLS #2:LOCATE #2,2,2:INPUT #2,"Fil
ename [8 Max]-",name$
355 IF LEN(name$)>8 THEN 350 ELSE CLS
#2:name$=UPPER$(name$)
360 LOCATE #2,8,2:PRINT #2,"Press PLAY
, then any key":WHILE INKEY$="":WEND
365 CLS #2:LOCATE #2,7,2:PRINT #2,"Loa
ding [;name$;].".
370 LOAD"!"+name$:OPENIN"!"+name$+".CH
R":FOR t%=1 TO 224:INPUT #9,re%(t%):NE

```

```

XT:CLOSEIN
375 CLS #2:LOCATE #2,9,2:PRINT #2,"Pro
gram now in memory."
380 FOR t=1 TO 3000:NEXT:CLS #2:LOCATE
#2,9,2
385 PRINT #2,"** Information area **":
GOTO 60
390 CLS #2:LOCATE #2,2,2:INPUT #2,"Fil
ename [8 Max]-",name$
395 IF LEN(name$)>8 THEN 390 ELSE CLS
#2:name$=UPPER$(name$)
400 CLS #2:LOCATE #2,6,2:PRINT #2,"Pre
ss REC+PLAY, then any key"
405 WHILE INKEY$="":WEND:CLS #2:LOCATE
#2,7,2:PRINT #2,"Saving [;name$;].".
410 SAVE"!"+name$,b,39000,1792:OPENOUT
"!"+name$+".CHR":FOR t%=1 TO 224:WRITE
#9,re%(t%):NEXT
415 CLOSEOUT:CLS #2:LOCATE #2,12,2:PRI
NT #2,"Saving complete"
420 FOR t=1 TO 3000:NEXT
425 CLS #2:LOCATE #2,9,2:PRINT #2,"**
Information area **":GOTO 60
430 CLS #2:LOCATE #2,4,2:INPUT #2,"Ent
er character code [32-255]-",char$
435 IF VAL(char$)<32 OR VAL(char$)>255
THEN 430 ELSE CLS #2
440 v%=VAL(char$):IF re%(v%-31)=0 THEN
465 ELSE CLS #2
445 LOCATE #2,2,2:PRINT #2,"Already be
en redefined, Change [Y/N]"
450 k$=UPPER$(INKEY$):IF k$="Y" OR k$=
"N" THEN 455 ELSE 450
455 IF k$="Y" THEN 465 ELSE CLS #2:LOC
ATE #2,9,2
460 PRINT #2,"** Information area **":
GOTO 60
465 GOSUB 695:v%=(v%-32)*8:FOR t%=0 TO
7
470 POKE 39000+v%+t%,by%(t%+1):NEXT:CL
S#2:LOCATE #2,4,2
475 PRINT #2,"Character has been store
d away.":re%(v%/8+1)=1:FOR t=1 TO 3000
480 NEXT:CLS #2:LOCATE #2,9,2:PRINT #2
,"** Information area **":GOTO 60
485 CLS #2:PEN #2,1:LOCATE #2,10,2:PRI
NT #2,"Exit Generator [Y/N]"
490 k$=UPPER$(INKEY$):IF k$="Y" OR k$=
"N" THEN 495 ELSE 490
495 IF k$="Y" THEN MODE 1:PRINT"Progra
m aborted":END
500 CLS #2:LOCATE #2,9,2:PRINT #2,"**
Information area **":GOTO 60
505 CLS #2:LOCATE #2,3,2:PRINT #2,"Ver
tical or Horizontal Plane (V/H)"

```

```

510 k$=UPPER$(INKEY$):IF k$="V" OR k$="
H" THEN 515 ELSE 510
515 IF k$="V" THEN 525 ELSE FOR t%=8 T
O 1 STEP -1:FOR g%=1 TO 8
520 ro%(g%,t%)=gr%(g%,ABS(t%-9)):NEXT:
NEXT:GOTO 530
525 FOR t%=8 TO 1 STEP -1:FOR g%=1 TO
8:ro%(t%,g%)=gr%(ABS(t%-9),g%):NEXT:NE
XT
530 FOR t%=1 TO 8:FOR g%=1 TO 8:gr%(t%
,g%)=ro%(t%,g%):LOCATE #5,t%,g%
535 IF gr%(t%,g%)=1 THEN PEN #5,0:p=1
ELSE PEN #5,2:p=0
540 PRINT #5,CHR$(255);:PLOT mx1%+(t%*
4)-2,my1%-(g%*2),p
545 PLOT mx1%+(t%*4),my1%-(g%*2):PLOT
mx2%+(t%*2),my2%-(g%*2):NEXT:NEXT
550 CLS #2:LOCATE #2,9,2:PRINT #2,"**
Information area **":GOTO 55
555 code%=32:CLS #2:LOCATE #2,7,2:PRIN
T #2,"To Screen or Printer (S/P)"
560 k$=UPPER$(INKEY$):IF k$="S" OR k$="
P" THEN 565 ELSE 560
565 IF k$="P" THEN 615:ELSE CLS #2:FO
R t%=0 TO 223:IF re%(t%+1)=0 THEN 535
570 CLS #2:FOR t%=(code%-32) TO 223:IF
re%(t%+1)=0 THEN 585
575 CLS #2:LOCATE #2,1,2:PEN #2,0:PRIN
T #2,STR$(t%+32);:PEN #2,1:FOR g%=0 TO
7
580 a%=(t%*8):p%=PEEK(39000+a%+g%):PRI
NT #2,STR$(p%);:NEXT:GOTO 590
585 CLS #2:LOCATE #2,13,2:PEN #2,1:PRI
NT #2,STR$(t%+32);"-UNCHANGED"
590 LOCATE #2,2,3:PEN #2,1:PRINT #2,"[
F] Finished [N] Next [C] Character"
595 k$=UPPER$(INKEY$):IF k$="F" OR k$="
N" OR k$="C" THEN 600 ELSE 595
600 ON INSTR("FNC",k$) GOTO 675,605,61
0
605 NEXT:GOTO 675
610 GOSUB 680:GOTO 570
615 CLS #2:PEN #2,1:LOCATE #2,3,2:PRIN
T #2,"Put printer on line, press any k
ey"
620 WHILE INKEY$="":WEND:'FOR t%=0 TO
223:IF re%(t%+1)=0 THEN 580
625 FOR t%=(code%-32) TO 223:IF re%(t%
+1)=0 THEN 645
630 CLS #2:LOCATE #2,1,2:PEN #2,0:PRIN
T #2,STR$(t%+32);:PEN #2,1
635 PRINT #8,"SYMBOL";STR$(t%+32);:FOR
g%=0 TO 7:a%=t%*8:p%=PEEK(39000+a%+g%
)
640 PRINT #2,STR$(p%);:PRINT #8,STR$(p

```

```

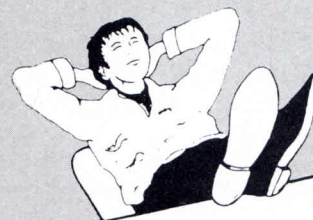
%);:NEXT:PRINT #8,CHR$(13):GOTO 650
645 CLS #2:LOCATE #2,13,2:PEN #2,1:PRI
NT #2,STR$(t%+32);"-UNCHANGED"
650 LOCATE #2,2,3:PEN #2,1:PRINT #2,"[
F] Finished [N] Next [C] Character"
655 k$=UPPER$(INKEY$):IF k$="F" OR k$="
N" OR k$="C" THEN 660 ELSE 655
660 ON INSTR("FNC",k$) GOTO 675,665,67
0
665 NEXT:GOTO 675
670 GOSUB 680:GOTO 625
675 CLS #2:LOCATE #2,9,2:PRINT #2,"**
Information area **":GOTO 55
680 LOCATE #2,1,2:PRINT #2,CHR$(18);:L
OCATE #2,2,2:PEN #2,1
685 INPUT #2,"Enter start code (32 to
255) ",code%
690 IF code%<32 OR code%>255 THEN 680
ELSE RETURN
695 FOR t%=1 TO 8:q%=0:z%=0:FOR g%=8 T
O 1 STEP -1
700 IF gr%(g%,t%)<>1 THEN 705 ELSE z%=
z%+2^q%
705 q%=q%+1:NEXT:by%(t%)=z%:NEXT:RETUR
N
710 RESTORE 870:FOR t%=0 TO 18:READ d:
POKE mem%+t%,d:NEXT:CALL mem%
715 SYMBOL 255,254,254,254,254,254,254
,254,0
720 WINDOW #1,2,39,15,19:PAPER #1,2:CL
S #1
725 WINDOW #2,2,39,22,24:PAPER #2,2:CL
S #2
730 WINDOW #3,2,11,2,11:PAPER #3,2:CLS
#3
735 WINDOW #4,15,39,2,12:PAPER #4,2:CL
S #4
740 WINDOW #5,3,10,3,10:PAPER #5,2:CLS
#5
745 PEN 1:LOCATE 3,20:PRINT STRING$(38
,207):LOCATE 3,25:PRINT STRING$(38,207
);
750 LOCATE 3,12:PRINT STRING$(10,207):
LOCATE 16,13:PRINT STRING$(25,207)
755 FOR t%=16 TO 19:LOCATE 40,t%:PRINT
CHR$(207):NEXT
760 FOR t%=3 TO 11:LOCATE 12,t%:PRINT
CHR$(207):NEXT
765 FOR t%=3 TO 12:LOCATE 40,t%:PRINT
CHR$(207):NEXT
770 FOR t%=23 TO 24:LOCATE 40,t%:PRINT
CHR$(207):NEXT
775 FOR t%=31 TO 159 STEP 16:PLOT t%,3
68,1:DRAW 0,-128:NEXT
780 FOR t%=368 TO 240 STEP -16:PLOT 31

```

```

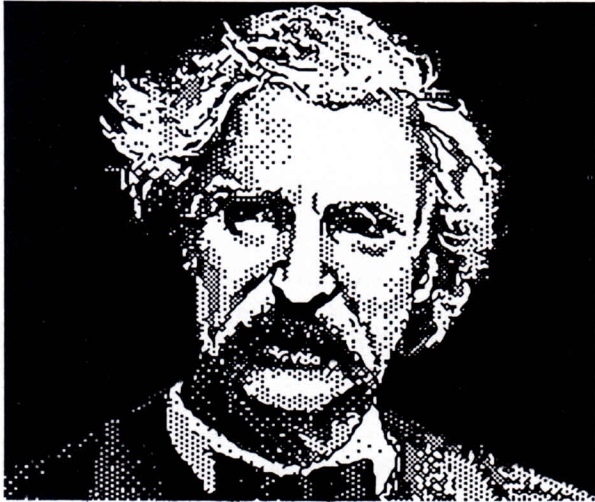
,t%:DRAW 128,0:NEXT
785 PLOT 15,15:DRAW 610,0:DRAW 0,50:
DRAW -610,0:DRAW 0,-50
790 PLOT 15,384:DRAW 162,0:DRAW 0,-1
62:DRAW -162,0:DRAW 0,162
795 PLOT 15,94:DRAW 610,0:DRAW 0,82:
DRAW -610,0:DRAW 0,-82
800 PLOT 624,384:DRAW -402,0:DRAW 0,
-178:DRAW 402,0:DRAW 0,178
805 PLOT 238,368:DRAW 370,0:DRAW 0,-
18:DRAW -370,0:DRAW 0,18
810 PLOT 238,336:DRAW 370,0:DRAW 0,-
18:DRAW -370,0:DRAW 0,18
815 PLOT 270,304:DRAW 66,0:DRAW 0,-6
6:DRAW -66,0:DRAW 0,66
820 PLOT 510,304:DRAW 66,0:DRAW 0,-6
6:DRAW -66,0:DRAW 0,66
825 LOCATE #5,1,1:PRINT #5,CHR$(22)+CH
R$(1)
830 PEN #4,1:PAPER #4,0:LOCATE #4,2,2:
PRINT #4," *CHARACTER GENERATOR* "
835 LOCATE #4,2,4:PRINT #4," MODE 0 [S
IZES] MODE 1 "
840 FOR t%=6 TO 9:LOCATE #4,4,t%:PRINT
#4," "
845 LOCATE #4,19,t%:PRINT #4," ":NE
XT
850 PEN #2,1:LOCATE #2,9,2:PRINT #2,"*
* Information area **"
855 PEN #1,1:LOCATE #1,3,2:PRINT #1,"[
I] INVERSE [R] ROTATE [V] VIEW"
860 LOCATE #1,3,3:PRINT #1,"[C] CLEAR
[S] STORE [E] EXIT"
865 LOCATE #1,3,4:PRINT #1,"[M] MIRROR
[T] TAPE [D] DATA":RETURN
870 DATA &01,&08,&07,&21,&00,&A5,&11,&
58,&98,&7E
875 DATA &12,&0B,&23,&13,&78,&B1,&20,&
F7,&C9

```



Give your fingers a rest . . .

All the listings from this month's issue are available on cassette.



# BUDDING GENIUS?

If you're a budding Einstein (i.e. Amstrad programmer) we'd like to see what you have to offer. We are constantly in need of good quality software for our aggressive marketing program and would like to make contact with you. Genuine devalued Aussie dollars are offered! Give us a call or send us a sample.

## Strategy Software

Box 5000  
Glen Waverley  
Victoria 3150

(03) 560 4324

### NEW FROM PRIDE

SUPERSPRITES	(D)	\$36.50
SUPERSPRITES	(C)	\$27.50
PRINTER PAC II	(D)	\$36.50
PRINTER PAC II	(C)	\$27.50
* F.I.D.O.*	(D)	\$36.50

\* AVAILABLE ON ROM

CREATE FAST MOVING ANIMATED GAMES EASILY. MENU DRIVEN AND INCLUDES ANIMATION DEMO.

FOR USE WITH EPSON COMPATIBLE PRINTERS. PROPORTIONAL SHADING TEXT DUMPS/ BUFFER OPTION.

FILE AND DISC ORGANISER. ORGANISE AND MANIPULATE YOUR FILES AND DISCS. A VERY USEFUL UTILITY.



Bankcard & Mastercard  
Welcome

P.O. BOX 288, MORISSET  
NSW 2264  
(049) 732754

#### PRIDE UTILITIES

ODDJOB	(D)	\$36.50 *
TRANSMAT	(D)	\$36.50 *
TRANSMAT	(C)	\$27.50
SPIRIT	(C)	\$27.50
TOMCAT	(C)	\$19.50
SYSTEM X	(D)	\$26.50 *
SYSTEM X	(C)	\$18.50
SCRIPTOR	(C)	\$17.50

\* AVAILABLE ON ROM

DK'Tronics Chip Kit  
256k RAM expansion kit for  
Amstrad PCW 8256

\$145

### FROM K.D.S. ELECTRONICS

- \* 8-BIT PRINTER PORT \$54.50
- \* POWER CONTROLLER - 8  
SWITCHING RELAYS : 5 AMPS AC  
OR DC UP TO 240 VOLTS. CONTROL  
LIGHTS FANS ETC. \$145.00
- \* PRINTER 'T' SWITCH CONTROLS  
2 PRINTERS BY EITHER SOFTWARE  
OR MANUAL CONTROL. \$78.00

PRICES MAY VARY AS STOCK  
IS REPLACED. WE ALWAYS  
REDUCE PRICES WHERE  
CIRCUMSTANCES ALLOW.  
ALL PRICES INC. P&P.

64k MEMORY  
EXPANSION  
FROM DK'TRONICS --  
\$145.00

#### ESP LIGHT PENS

ESP 464 (cas)	\$45.00
ESP 464 (disc)	\$65.00
ESP 6128 (disc)	\$65.00

#### KONIX SPEED KING

JOYSTICK STEEL SHAFTED  
\* SHAPED TO FIT YOUR  
HAND  
\* INDUSTRIAL MICRO  
SWITCHES --- \$35.00

# BUSINESS COMPUTING

WITH THE AMSTRAD



Opening a  
window on  
vertical  
software

# IS IT A BIRD? IS IT A 'PLANE? IS IT A CHOCOLATE BAR?

*The answer is it could be all three but as every one knows by now it's Amstrad's new PC - the Airo.*

We had our first view of the Airo a couple of weeks back, within days of the new machine arriving in Australia. Before we go any further into this review it should be pointed out that this writer is no fan of PCs in general and their operating system in particular. As the pollies are found of saying - 'having said that let me say this', the Airo is different. I will endeavor not to use the term compatible because I believe the Airo goes beyond mere compatibility.

### *What's a compatible?*

O.K. for those who don't know we'd better explain 'compatible with what?'. Back in the good old days (1983) IBM introduced their first desktop Personal Computer (hence PC) The operating system they chose was by Microsoft in preference to the then unfinished CP/M-86. This operating system is commonly known as MS-DOS or, by those familiar with it, MESS-DOS. I'm not sure if IBM set out to make their machine the standard or if they just happened along at the right time with a product which (almost) fits on your desk or if it happened by default, but the end result over the last three years has been that the IBM-PC has certainly set the standard. Until now, many other manufacturers have spent lots of dollars in producing desktop computers which would run the same software as the IBM-PC. What in fact has happened is that it is the operating system (MS-DOS) which has determined compatibility more than the hardware.

The other major reason all this has occurred is that IBM were smart enough to produce a computer to which third-party vendors could add hard disks, monitors and various expansion cards. This produced a vast market for these items and other manufacturers were virtually forced to follow IBM's lead and produce machines which would accept these peripherals.

So there we are, a potted history of PCs.

How does the Airo match up and what makes it different? - read on.

### *Hardware*

Let's look at hardware first. The immediate thing that hits you on seeing the Airo is its compact size compared with other PCs and the IBM in particular. This is of prime importance if Amstrad are to get one on everyone's desk - and I presume that's their intention. The base unit, which contains either single or dual 5 1/4 drives and the CPU is around 12 inches wide and 14 inches deep. Atop the base unit sits either a mono or color monitor with facilities for tilting this way and that. The keyboard is a standard PC style with numeric keypad and function keys. The major difference hardware between Airo and its competition is its appendage. Yes, it has a mouse! Not that rodents are particularly rare in today's PC world but I believe the Airo is the first to offer one as standard equipment. As a 'mouser' from way back (on the Apple Mac) I can only say it's about time someone did it.

On opening up the base unit you are presented with a rather large hole taking up about one third of the main box. This is also standard equipment and to Amstrad's credit their 'hole' is smaller than everyone else's. Don't worry, it's not an overgrown ashtray for sloppy programmers, it's known as the expansion bus. There's a very good reason why the Airo's expansion bus is smaller than the opposition - it's because most of the others don't include anything like as much standard equipment as Amstrad has and extra slots are needed to fit things like graphic cards, color cards, RS232 cards and all the things that you really need but are 'optional extras' with the others. Make no mistake - if you're shopping around for a PC don't forget to take this into account. The Airo has three full-length expansion slots which, bearing in mind the standard equipment, should be ample for just about everyone.

The base unit is of course the main work-horse of the system and is where the disk drives live. This is where things get a little complicated. The options are as follows: single 360k drive, dual 360k drives, single 360k drive with 10 megabyte hard disk, single 360k drive with 20 megabyte hard disk or dual drives with 10 or 20 megabyte hard disks (using one of the expansion slots for a card which houses the hard disk. Hence there are a total of 6 options with respect to storage.

Standard memory on all models is 512k of RAM with the option to upgrade to 640k. How the extra RAM will be

installed is not clear at this stage, but more than likely it will be a simple matter of plugging in an extra chip. There is no provision for expanding RAM further than 640k but this is hardly necessary as most applications running under MS-DOS are unable to recognize more than 640k.

Other items contained in the base unit are a parallel printer port, standard RS232c serial interface (for communications or serial printers), mouse interface, color/mono graphic card, battery backed security RAM and real-time clock and (most important) a joystick port. The battery backed RAM and real-time clock are particularly nice touches providing data security in the event of power outage and the ability to time & date stamp documents respectively. These are the kind of things that most users don't miss until they've had them and then can't do without if they are unavailable. Top marks on this one.

The keyboard is of the standard PC type with 95 keys including 10 function keys, cursors keys and numeric keypad. The keyboard is the best produced by Amstrad to date and leaves the PCW keyboard for dead. The keys are firm without being 'hard' and the keyboard is 'ergonomic' enabling the user to set his/her own tilt level.

Probably the most unusual aspect of the Airo is the mouse and it's difficult to anticipate how people will react. Having moused around on an Apple Mac for about 15 months I wouldn't be without one. I don't want to get into the software area at this stage but the mouse is inextricably bound up with the environment under which most of the Airo's software will run. As a piece of hardware most people will find it invaluable once they are used to it. On our early production run machine I found the mouse to be a little on the sensitive side but Mitsubishi-AWA assure me that production run models will be less touchy.

There are two monitors available for use with the Airo, one being a high-res color monitor with a potential of 16 colors at a resolution of 640 x 200 pixels, the other being a little out of the ordinary for a PC in that it is an off-white background with black characters in text mode and grey shading in graphics mode. The shading is very similar to the Apple Mac and again, we have no problems using this type of screen day-in, day-out over 15 months. Mono screen resolution is 640 x 400. I have a personal preference for the mono screen for business use but I guess it would depend on the individual application. The graphics used on both monitors are essentially identical to those on the IBM and should cause no problems with software designed for that machines graphic card.

A printer is not supplied with the system although one will be available when the machine is released. The model is designated the DMP-3000 and is similar in style to the DMP-2000 with the exception that it's color matches that of the PC1512. The difference between it and the DMP-2000 lie in the graphics drivers which are incompatible with the CPC range. However, this doesn't mean that if you have both an Airo and a CPC you can't use the same printer on both as the

DMP-2000 and 3000 will handle text-only from both computers.

## Software

Documenting the software supplied with the machine would fill this magazine several times over (and probably will over the coming months and years). Let's begin with the operating systems, that's right - plural!

The first of these is MS-DOS 3.2 from Microsoft. MS-DOS was licensed to IBM and is supplied with IBM PCs under the title of PC-DOS. Anyone familiar with CP/M or CP/M PLUS won't be too lost using MS-DOS although naturally the commands are different in some cases. Most applications purchased by the user will be run under this system. More interesting is that Amstrad have supplied DOS PLUS from Digital Research as well. DOS-PLUS is compatible with MS-DOS and CP/M-86 and enables users to operate their software under the GEM environment. Compatibility with CP/M -86 should not be discarded as worthless because CP/M-86 is by far the most popular operating system in Japan and the Japanese have developed some startling software using this system. DOS-PLUS also has the capability to run several programs at once, enabling the use of print spoolers etc.

What's GEM? I hear you ask. Again that's a story on it's own but it's often described as WIMP (Windows, Icons and Mouse Processing) - there are other uses for the acronym but this is a family magazine. Basically, GEM makes the system easy to use. It's far easier to point the mouse and click on a button than it is to start typing a series of commands to find a file hidden along paths and in sub-directories of sub-directories of directories (no- I didn't repeat myself, that's typical MS-DOS). GEM stands for Graphic Environment Manager and it is just that. The user is presented with a graphic representation of a desk top and files are presented as Icons. Icons live in folders and this makes life very easy because it is so similar to the way most people work - open a folder and get the file. To run GEM Paint for example, you would simply click on the program's Icon. Again, full marks to Amstrad for making life easy for the novice PC user and easier for those experienced in the field.

The other major items of software supplied are GEM Paint and Locomotive Basic 2. It's almost worth spending the extra money on a color monitor just for GEM Paint. Paint is an artist's tool, an executive doodler and is extremely addictive to children. Could I suggest you don't use Paint until the little ones are off to bed? Failure to ignore this advice could mean you having to buy another Airo just to keep them off your back - you've been warned! Paint provides various tools to draw different shapes on the screen with the ability to fill irregular shapes with any available color. It provides a paintbrush (no kidding) with the user having the ability to alter

the size of the brush head as well as the pattern to be used when painting. Also provided are various other tools for drawing lines, using text and lots more. A very exciting piece of software which, whilst totally unnecessary to most business users will provide hour upon hour of fun and delight for those willing to put in the time.

Locomotive Basic 2 is also very exciting but for different reasons. First and foremost - it's fast, very fast. Because the Airo uses the Intel 8086 processor as it's main chip it is a considerably faster machine than the IBM PC with a clock speed of 8 Mhz as opposed to big blue's 4.77 Mhz. If that doesn't mean anything to you, don't worry about it - the main thing is that the machine won't keep you waiting. Locomotive Basic 2 adds to this advantage because it is faster than IBM's Basic2 and Microsoft's MBASIC adding up to a mighty fast combination. We could fill a page with tables showing how Locomotive Basic measures up against the others in various benchmark tests but we don't have the space, if you're interested send us a SAE and we'll send you a copy. We'll be looking more closely at this Basic in the near future but this is supposed to be a hardware review so we'll just whet your appetite a little by telling you that Locomotive Basic 2 has all the features of the version running on the PCW machines as well as windowing which enables you to run a program in one window whilst looking at the listing in another window. There are lots more features but I'm afraid they'll have to wait for another time.

Other software supplied includes a pile of mainly undocumented utility software which the hackers will find very useful. Anyone familiar with the software supplied with the PCW will be aware of the type of thing available.

Software licensed by Amsoft will also be available at the usual ridiculously cheap prices. Software available at the launch date (sometime in November) includes Wordeasy (a version of Wordstar), Supercalc 3, Reflex (a very powerful relational database licensed from Borland International) and Sidekick (a Borland utility suite providing things like a calculator, telephone dialing and an index card system, all of which are available whilst you are running another application). Games available include Snooker, Cyrus Chess (see review this section), Golf, Pitstop, Summer Games II and Winter Games. As well as the Amsoft range there are literally thousands of other programs available from third-party vendors which will run without any compatibility problems at all.

Speaking of software compatibility, I should mention that Amstrad have tested a large number of programs (which I have no intention of listing in full) including Symphony, Lotus 123, DBase II & III, Turbo Pascal, RBase 5000, Word Perfect, Wordstar 3.3 & 2000 (v1.00), Framework 1.1 and of course, Microsoft's Flight Simulator. Suffice it to say that I am satisfied beyond any shadow of doubt that the Airo is capable of running any IBM PC software that the user is likely to want to use. Incidentally, it is possible for those interested to run

CP/M software with a card available from Digital Research which contains a Z-80 chip and comes with the CP/M operating system.

There are far more aspects of the PC1512 or Airo than can be covered in one article but I hope I've given a fair indication of the machine and it's capabilities.

In summary, Amstrad have set new standards in the PC world. They are flying with the big boys now and the machine's success will very much depend on who sells them. I doubt that you'll find this Amstrad hidden amongst the washing machines and video recorders! It's going to require a deal of knowledge to be able to sell these machines and give them (and the buyer) the backup they deserve. The National Banks and BHPs of the world aren't likely to pop into their local bulk store to pick up a sixpack! Selling the Airo to big business is going to require a degree of professionalism which hasn't so far been required on Amstrad computers. Provided Mitsubishi-AWA aren't tempted by the prospects of quick sales and provide adequate training and backup through their dealer network then I'm sure the Airo will be a tremendous success - it certainly deserves to be.

What's that you say? - you want to know the prices? Mmmmmm, well O.K., but don't tell anyone because they're not released yet.

Single drive	Mono monitor	\$1499.00
Dual drive	Mono monitor	\$1949.00
Single drive	Color monitor	\$1899.00
Dual drive	Color monitor	\$2349.00

These prices include software as detailed below as well as the mouse, serial interface, parallel interface, joystick port, lightpen port, speaker with volume control, EGC graphics, battery backed clock and security RAM with 512k RAM as standard and 3 expansion slots. Available in the near future will be 10 & 20 megabyte hard disks priced at around \$1000 and \$1500 respectively. Remember, all prices are approximate and are subject to change without notice.

Software supplied consists:

MS-DOS 3.2  
DOS Plus  
GEM  
GEM Desktop  
GEM Paint  
Network support  
RAM drive  
Locomotive Basic 2  
Heaps of utilities!

KP



**THE PCW8256 manuals are by no means prime examples of bad documentation. But certain sections are relatively hostile, and others poorly structured.**

The manual writers may have begun with good step-by-step intentions, yet somehow they have got lost in sidetracks and backtracks, and occasionally appear unable to distinguish between what is important from the user's angle and what they believe to be the cleverest features of the product.

I know of intelligent people who are still grappling with the LocoScript instructions after weeks of patient study.

Therefore much of Mastering the PCW is rightly given over to rewriting the documentation on LocoScript and CP/M. Content-wise there's little in the book that you couldn't find in Volume 1 of the User Guide, but the organisation of the material is superior – plainly the result of time having been spent actually using the machine in a variety of everyday circumstances, including using it to write the book itself.

The revamping of the original documentation is obvious almost from the start. After a few introductory remarks, containing the obligatory outline of what computers do and the usual good advice about taking care of discs and so on, the section on LocoScript – which forms the major part of the book – starts with the direct printing mode using the PCW as a memory typewriter.

This may seem an odd approach until you remember that direct printing involves no disc access. It is therefore a good way of learning to use the printer with its quirks, and the keyboard with its dedicated word processing keys and alternative character sets, without having to bear in mind disc management at the same time.

Working with disc files is then treated in fairly digestible chunks arranged in increasing order of difficulty, and culminating in detailed but lucid chapters on the complicated subjects of page layout and stored format files.

A final chapter in the LocoScript section provides a confidence booster by dealing with less daunting topics such as saved phrases, disc-group

## Help for the struggling tyro

names, and handling long documents.

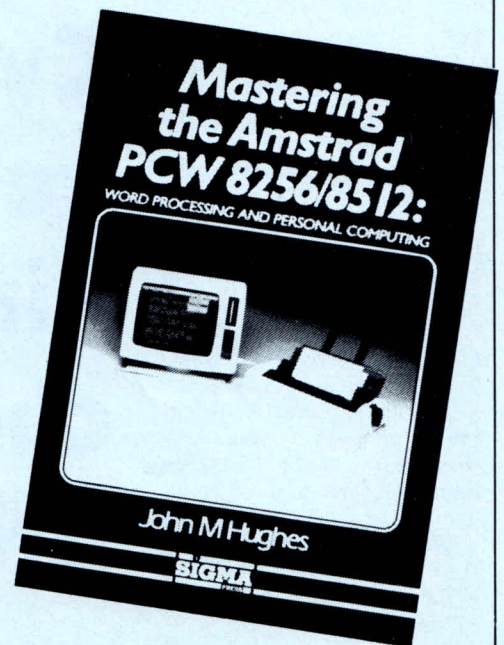
I have only one criticism of the LocoScript section. The author says he has been working with a pre-release version 1.2 of the program, but assumes that most people will possess the original version 1.0. Here lies an inherent problem with writing books on recently released computers in order to capitalise on a new niche in the market.

All the excellent advice the author gives about getting round the bugs and limitations of version 1.0 has been made redundant in one fell swoop since Amstrad began to distribute version 1.2, and announced a free update to those who had bought earlier versions.

Whereas the chapters on LocoScript represent an expansion of the User Guide, this is not true of those devoted to CP/M. The official CP/M documentation and the tree-structured Help facility on side 4 of the system discs supplied with the PCW deal with built-in commands and external utilities in greater detail than does Mastering the PCW. Again, however, the new book is more coherent than the former.

This is no great achievement when you remember that parts of the official documentation seem to have been written with the aim of keeping CP/M information secret from the masses, and perpetuating the idea that computer scientists live on a higher plane than Joe Bloggs who doesn't have a PhD in the subject.

But there again, it's not that easy a task to explain in simple language some of the niceties of CP/M, and it's at least as well done in Mastering the PCW as in the many other books



available on the same topic.

The last part of the book concentrates on commercially available applications software, using as detailed examples SuperCalc 2 and CamBase – not everybody's cup of tea in the way of spreadsheets or databases, but typical enough to illustrate what can be done with the PCW once you get beyond LocoScript.

The last chapter gives an overview of two accounting packages – Sagesoft Popular Accounts and Camsoft Payroll – and one alternative word processing program, New Word.

The book is written in an easy, jargon-free style, and is peppered with imaginative and apt expressions. Being told that editing a template is not unlike peeling an onion, or that watching a long document being saved is about as exciting as watching paint dry, livens up what could be a dry subject, and neatly encapsulates the experiences of those who know LocoScript.

If you're an expert coming to the PCW you won't need a book like this. In fact much of the fun of using a new machine or a new piece of software lies in the haphazard discovery of hidden features, and the satisfaction gained from finally understanding the documentation.

If you're not a computer buff, and particularly if you've been struggling with the first volume of the PCW User Guide, Mastering the PCW will be £9 well invested.

**Gabriel Jacobs**

DID you know that there are 14,772,512 ways of placing 16 Queens on a 16 x 16 chessboard so that no Queen attacks another?

A more useful fact is that almost anyone can test a chess program, particularly if it has a self-play (or demo) option like **Cyrus II** from Intelligent Chess Software.

First give it some simple end games. Now the majority of programs can win the K,Q v K ending, but Cyrus is the first micro program I have seen that can manage, at one minute per move, the K,R v K ending.

Even more remarkable is that it can win, at three minutes per move, the K, B, B v K ending.

This made me try the K, B, N v K, a tough test for any player, but the program, not surprisingly, does not know which corner to drive the King into and eventually failed itself on the 50 move rule – another neat feature.

The second, simple test is the N move mate. Cyrus has a "Problem" mode which can handle up to five move mates. It performs two movers in about eight seconds on average and three move mates take only a minute or so.

This is fast. One reason is probably because Cyrus, when it has only one possible move, makes such "no choice" moves immediately without analysing the outcome. Obviously, you might say, but the majority of chess programs don't have this feature.

A third test is opening traps like the Blackburne thus:

1. P-K4 P-K4
2. N-KB3 N-QB3
3. B-B4

This is the well known Giouco Piano, but Black now offers a pawn with 3. ---- N-Q5.

I have never seen any player (program or human) avoid this trap unless protected by book knowledge. Cyrus does not "know" it so:

4. N\*P Q-N4!

Really poor players will now lose quickly by playing:

5. N\*P

because it forks the Queen and Rook, but Q\*NP threatens mate in one. Cyrus has an

## Cyrus II, a veritable king among chess programs

Analysis option which displays what it thinks is the best line, over the next three or four moves, for both sides, and clearly sees the trouble, so played correctly:

5. B\*NP+ K-K2

and then, even more surprisingly, played:

6. O-O

That is it, again correctly, gives up the Knight for two pawns and a slightly better development.

A final and very tough test for any program is the Saveedra end game puzzle (see Figure I). This famous problem has baffled many people for over 80 years because it tests appreciation of subtleties like repetition of position, stalemate and under-promotion.

Indeed the best main frame programs of just 10 years could barely cope with it, and

they cost thousands of pounds to develop and run.

The puzzle is: White to play and win. Obviously White must promote the pawn, but it seems that Black can stop this by checking the White King continuously.

See if you can solve it before reading further.

The first move is forced, anything else loses the pawn, so P-B7. Cyrus, at three minutes per move, now played impeccably – Black keeps checking with the Rook but the White King, avoiding draw by repetition and a Rook skewer, moved steadily down the Knight file until it could play K-QB2!

This blocks the skewer, but Black now plays R-Q5 (see Figure II) threatening stalemate if the pawn promotes to a Queen.

Many people just cannot, or

will not, consider the under-promotion to a Rook. As it happens, Cyrus cannot under-promote either, although it fully appreciates the stalemate, so I did it this minor service and it then, almost immediately, saw the win and did so.

In conclusion Cyrus not only plays very good chess but has a number of very attractive features and options such as 3-D display with clocks, allowing moves to be retracted and/or replayed, saving and loading games, printing a game record, forcing a move at any time and changing levels, all in an easily learnt command mode.

Quite simply this is the best micro chess program I have ever seen, and very highly recommended.

**David Alliss**

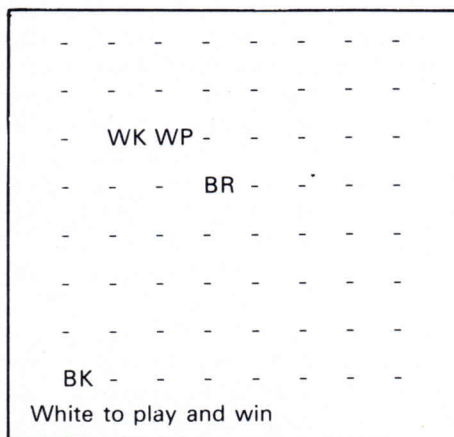
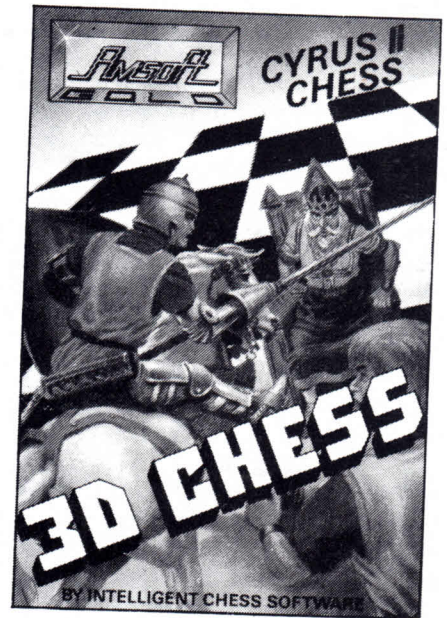


Figure I: The Saveedra end game

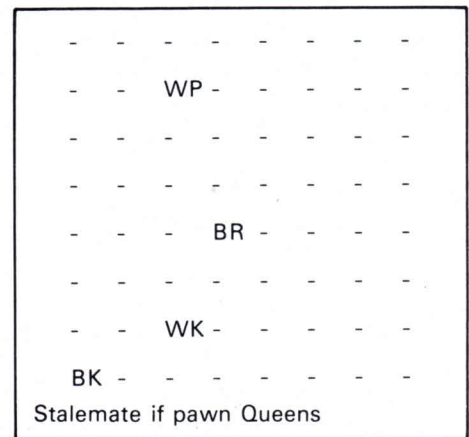


Figure II: Black now plays R-Q5

**I** N a recent review I only touched upon the fact that there is the world of difference between stock control and stock recording. I also suggested that the software houses must not play on the fact that the average Amstrad user does not even realise that this difference exists.

In writing this piece I am not rubbishing stock recording, or other simple stock systems.

An analogy may make this point clearer. Just because we can not afford a Rolls-Royce is no reason to mock the Metro. The former is the ultimate while the latter is the first practical means of transport for many drivers.

Just the same situation exists in stock management systems. Stock control is the ultimate, whereas stock recording is your first practical system.

I well remember being run ragged trying to shoehorn a full stock control system onto a 512k 16 bit mainframe with 320 megabytes of disc storage, the ability to run up to 32 programs concurrently, each with up to 10 files open.

Certainly the system had a very large number of customers, suppliers, items of stock and transactions to process, but even had these figures been far smaller a micro would have been as much use as triangular wheels on a bicycle.

The following list indicates the *very least* that needs to be carried out in a stock control system. The number of times any or all of these actions need to be done, or the value of the stock, is of no importance:

- 1: Record all stock items as they leave the warehouse.
- 2: Record all stock items as they enter the warehouse.
- 3: Reconcile what is in the warehouse and fit for issue with what the computer records indicate.
- 4: Calculate when a stock-out will occur.
- 5: Calculate which items must be purchased in order to replenish the warehouse with just sufficient stock to meet the likely demand for those items.
- 6: Incorporate into all calculations:
  - (a) The delay a supplier claims you will incur before receiving a delivery.

# TAKING STOCK

(b) How accurate these claims prove to be.

- 7: Maintain a full cross-reference system so that if stock-outs occur due to unusually high demands for any item, suitable alternatives may be supplied.
- 8: Record all monies spent on stock, tied up in stock and committed to future purchases of stock, and so on.

**JO STORK  
examines how far  
a micro can go  
down the road to  
full stock control**

If this made your head spin, think what it does to a computer. Some of these actions are self-evident, but others need explanation. Take heart, it is not too difficult to understand, even if carrying them out has broken many a spirit.

The businessman must know how many day's stock exists for every item in the warehouse. What complicates matters is that the rate at which even a single item is issued from the warehouse may be far from constant.

Many items are seasonal, weather dependent, fashion triggered or have limited shelf life. A stock of 1,200 Christmas trees in February is the definition of bad news while you store them for nine months. Similarly, having only 1,200 at the beginning of November could cause a stock-out long before many children have seen their first Santa. Saying you sell an average of 35 Christmas trees per day

through the year is as nonsensical as having 35 days of fresh cream in the stores.

The first thing that is required to achieve full stock control is therefore the ability to modify the calculations of when and how many items to order with the effects of the unpredictability of the demand for each item. The arithmetic is not particularly complex, but can only be achieved if an accurate history of every movement of every item over a reasonable period of time is available.

If you kept only 500 articles of stock, each of which have different variabilities of sales, you can now see that the computer must store plenty of sales history data as well as perform a fair few calculations.

Other factors need taking into consideration. Discovering you have 20 days stock for an item may not upset you. Phoning up your supplier and discovering he can not deliver any more for 50 days will ruin your day and hand the business to your competition. Consequently any stock system must, in this example, warn you at least 50 days in advance that an order needs to be placed for the item. This means storing the supplier's ability to replenish your warehouse.

Your supplier is in business to show a profit, and may have several "deals" available. They may allow discounts of increasing value, the more you purchase or the sooner you pay. Alternatively they may offer special reductions, such as on old products.

You, likewise, wish to operate profitably and so need to know these levels of discount as well as what are your stock-holding costs, borrowing costs and so on. If you are going to

have a sensible pricing policy which reflects the costs your suppliers charge, the amount of stored information has gone up again, as have the number of calculations required.

And I'm not finished yet. Let us take the case of a small workshop manufacturing Thingummys, each of which normally require a 3mm widget in their construction. If however, 3mm widgets are unavailable, either 3mm doodahs or 4mm widgets are suitable but slightly more expensive alternatives. This has a couple of effects.

- The usage of the alternative parts, doodahs, becomes higher than history indicated, thereby significantly raising the risk of doodahs now stocking out.

- Replenishment of this alternative may require more or less time than does the original component.

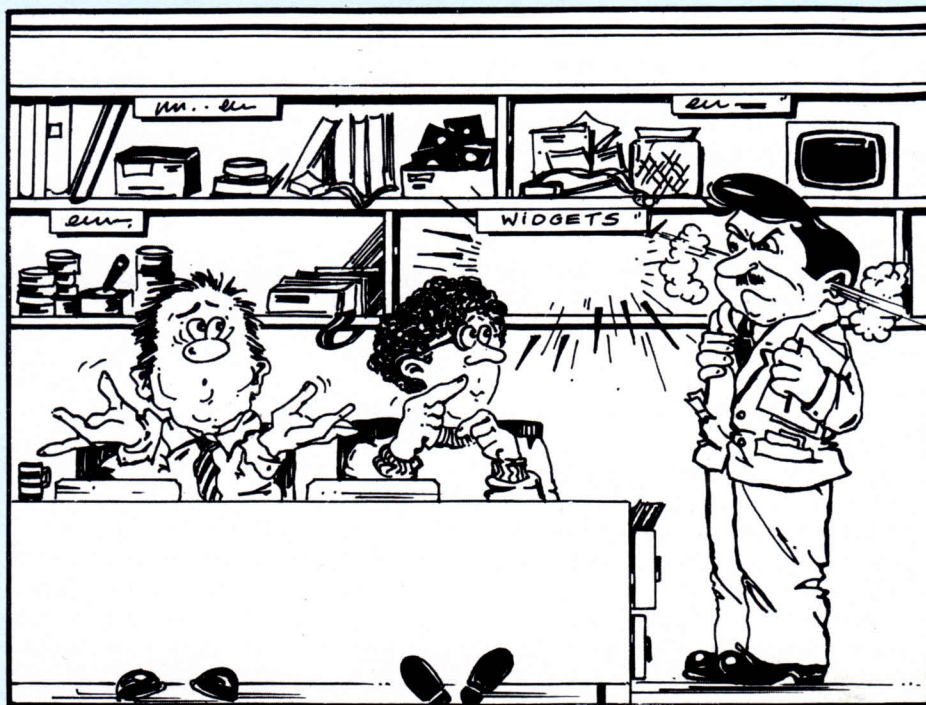
Telling our Thingummy manufacturer that he is about to stock out of 3mm widgets is not enough. He must also know if his workshop will come to a complete standstill for lack of suitable alternatives being available at the same time.

The suggested replenishment of 3mm widgets which the computer offers must, therefore, take into account all the resources which are available, not just the most commonly applied. All this takes up yet more storage and requires still more processor power.

What should be clear, even from this brief summary, is why I so nearly failed to squeeze a full stock control system on to a medium sized mainframe. Equally evident should be why an Amstrad is not up to the task, and why the software supplier who calls his precious program a control system is innocently or otherwise trying to hoodwink you.

Having said all this, do not write off the micro as a silicon joke. Providing you do not have more than about 300 stock items, 250 customers and 150 suppliers to manage, it is perfectly capable of performing some of the eight activities above. This will fall far short of full stock control, but can be extremely valuable nonetheless.

Examining the eight points which constitute stock control we discover that the Amstrad can handle the first two items in the list – recording the



*"Telling our Thingummy manufacturer that he is about to stock out of 3mm widgets is not enough..."*

acceptance from the suppliers into the warehouse and the issuing of the goods to the customers.

Come the bi-annual trauma of stocktaking the Amstrad can also compare what is on the shelves with what is in its records. However no worthwhile attempt is possible at computing how much of this stock is beyond dates and so on.

Provided your stock movements are moderately constant throughout the year – that is, you sell furniture not ice cream – then calculation of when stock-outs might occur is possible, as is a simple estimate of the replenishment level required. This will prove far from infallible, and many businessmen would be wise to over-ride the computer's recommendations on occasions.

So far, of the first five necessary actions I listed as being the most basic requirements of a stock control system, the Amstrad performs two in full and three in part. The 6th and 7th in my list are completely beyond a micro. It was these areas which created all my headaches on the mainframe.

As for the 8th and last, some costing is possible on the Amstrad. Simple calculations based on the value of the items entering and leaving the warehouse is reasonable, as is the value of the contents of the stores. It is the computations which

deal with forecasting what the future value of inventory will be which are impossible.

Adding all the eight steps together, we find a score of:  $1+1+.5+.5+0+0+.5=4/8=50\%$  for these activities. Far short of full stock control, as I said.

Returning to my early analogy, it would be stupid to be stuck at home for lack of transport merely because you can not afford a Rolls-Royce. Even simple stock recording systems can quickly pay for themselves, by saving you time and money compared to a pen and paper method.

My advice to you is: **Do not be fooled by the claims of the software suppliers into thinking all your stock problems are over once you buy their wonder-child.**

When choosing your system ignore the advertising and look closely at the file sizes, the record contents and the data which can be reported. This will give you a short-list of those systems which appear to offer the necessary facilities, irrespective of the claims made. The final selection must be the package which is easiest to use.

Finally, before signing a cheque, ask yourself the question: "Am I disciplined enough to keep the records up to date at all times?" If you have any doubts, you had better stick to zapping aliens instead.

## GUNFRIGHT

### CLEAN UP THE TOWN

Silence falls over the Wild West town. You're standing in the main street facing the ugliest outlaw you have seen in months.

A twitch, a crash of gunfire and Tombstone Pete lies dead. If you've ever fancied yourself as a bit of a Clint Eastwood you must buy Gunfright.

It is the latest release from Ultimate and as usual it employs their now famous 3D technique. Using this programming technique characters appear to walk in and out of the screen and around other characters.

In Gunfright you are the sheriff of a town which has been overrun by bandits. Your task is quite simple - track them down and shoom 'em dead.

Before you set out into the town you must collect your wages. This is done by shooting falling money bags.

This money is necessary to keep you in bullets and horses, and to pay the fines you incur for shooting innocent bystanders.

The game begins for real as you set out from the jail. The streets contain many people and you must not bump into any of them or you lose one of your three lives.

Having located the gunslinger you must shoot him before he shoots you. Once he has been hit you enter the quick-draw stage of the game.

A large picture of the outlaw moves around the screen and your gun sights rest at the bottom.

When the computer flashes, draw, quickly take aim and fire. But if you move the joystick before you are told to you are branded a cheat and unfortunately penalised.

Gunfright is the best Ultimate game to date - a simple theme which has been beautifully executed.

Steve Brook

Sound	7
Graphics	9
Playability	9
Value for Money	9
Overall	9

## Kung-Fu Master

### ADDICTIVE COMBAT

Kung-Fu Master, one of the latest original coin-op games for the Amstrad, comes from US Gold. You are the Kung-Fu Master and have to battle your way through the wizard's temple to rescue the maiden imprisoned there.

As you progress through the five levels the wizard sends out his minions to thwart you. These include henchmen who are easily defeated with one hit and knife throwers - whose knives you have to dodge - requiring two hits to stop them.

There are also dwarves who somersault on to you unless you can hit them first, and finally the guardians who are the strongest of the lot and require a number of well placed blows to defeat them.

If that isn't enough, pots drop from the ceiling. If you don't hit them before they land snakes crawl out and wriggle towards you. Dragons appear in a similar manner.

Killer bees come at you from both sides

of the screen, and mystic globes descend and hover before exploding into extremely dangerous fragments.

You start off with three lives and the top part of the screen displays your energy level which rapidly goes down if you come into contact with any of the people or devices.

When you meet up with a guardian his energy level is displayed and it's up to you to defeat him before your energy runs out.

Kung-Fu Master is addictive, and needs skill, speed and good timing to be able to master the different levels. It is the sort of game where if you fail you'll decide to have one more battle in an attempt to get further.

Overall a well-written, challenging game which will keep you going back for more.

Ian Duerden

Sound	6
Graphics	8
Playability	8
Value for Money	8
Overall	8

## Graham Gooch's Test Cricket

### TESTING CRICKET

This summer a bumper crop of cricket simulations have been released for many of the popular micros. Having just spent several hours playing Graham Gooch's Test Cricket I can confidently say that it is the best of the lot.

The program contains numerous statistics on the England and Australian teams.



from page 37

## SHADOWFIRE

These figures are an important part of the game as they determine a player's strengths when playing in simulation mode. Every item of data can be edited before beginning the game should you want to keep the figures up to date.

The program's graphics are excellent, with you looking down the wicket from behind the bowler.

The game can be played in arcade or simulation mode. When batting in arcade mode your control is restricted to the timing of the stroke, achieved by pressing the fire button at the correct time as the ball approaches you.

When bowling you must select whether you are bowling to legside or offside and then waggle your joystick like mad to get some pace on the ball.

Simulation mode doesn't involve any waggling when bowling.

Having selected legside or offside the player's statistics, combined with how often you have used him that innings, will determine how he performs.

When batting you must move the batsman on to his front or back foot for an attacking or defensive stroke and get the timing correct.

Graham Gooch's Test Cricket is the first cricket game that I have played which actually relies heavily upon your own skill. I can recommend it without hesitation.

Jon Revis

Sound	7
Graphics	9
Playability	9
Value for Money	9
Overall	9

### PAUSE FOR YAWNS

Ambassador Kryxix has been captured by the evil General Zoff, and unfortunately Kryxix was carrying the plans for a revolutionary new starship, the Shadowfire.

As mission controller you must guide the Empire's Enigma team aboard Zoff's skyfortress and rescue Kryxix before the plans are discovered.

None of the team have teleport ability, so you must first get a transport droid aboard the skyfortress. This can then be used to beam your chosen few across to the ship.

The first problem you will encounter is a locked door. I would suggest that you provide Sevrina, the humanoid female, with a few tools. Once the door is open you can begin to move each of the characters deeper into the ship.

Each character has a movement icon, and accessing this will display the directions currently available.

This method of moving the characters around gets pretty tedious, one good reason for only taking a couple of the Enigma force on board.

As you wander around the skyfortress you will be attacked by aliens.

Having been told which member is under attack you must select that character's icon, then their battle icon and finally inform them whether to attack or retreat.

Once completed you just sit back and wait to be told whether they survived or not. There is no arcade type shoot-out, just computer-mediated boredom, and the game carries on in this way ad

infinitum. It is a great pity that such excellent graphics have been wasted on a mediocre game such as this.

Carol Barrow

Sound	6
Graphics	9
Playability	6
Value for Money	6
Overall	6

## BIGGLES

### A BIGGLES TOO MANY

You've read the Biggles books, you've seen the Biggles film, now play the Biggles computer game. Based on the new film, the action takes place both in war-torn 1917 and present day London.

The complete program consists of four interconnected games which occupy both sides of the cassette.

The first program is Timewarp, which actually contains three of the four games.

In section one of Timewarp you are at the controls of a First World War biplane. Flying over enemy territory you must shoot down enemy planes and bomb field guns and ammo dumps.

Section two is a crude version of Commando in which you guide Biggles on foot behind enemy lines.

Part three takes place in London 1986 and is the most boring of the three sections. You and your time twin, Jim are involved in a roof top chase over London town.

You must both leap from roof to roof while avoiding the boys in blue. Your objective is to locate a secret code which will prove useful back in 1917 in the game on the second side of the cassette.

These first three games are not played in any particular sequence. As benefits the name, Timewarp, you can be whisked from one game and dumped in the middle of another at any time.

Side two of the cassette contains the fourth game The Sound Weapon, which is better than the first three. It is a combination of flight simulator and strategy game.

You must pilot a modern day police helicopter over enemy lines back in 1917, the aim of the game being to locate and destroy the evil, destructive sound weapon.

Despite the redeeming efforts of the final game I'm afraid this Biggles is doomed to failure.

Carol Barrow

Sound	8
Graphics	6
Playability	6
Value for Money	6
Overall	6

## GHOSTS 'n' GOBLINS

### RESCUE A DAMSEL

Ghosts 'n' Goblins is the latest coin-op video game to be released by Elite, following on from others such as Bomb Jack and Commando. It is typical of the platform and ladders type seen in the amusement arcades around the country.

In this fighting fantasy story you control a knight who has to fight and dodge his way through many perils in order to rescue the damsel from the clutches of the demonic overlord.

As you can well imagine the overlord has put all the obstacles in your path that he can think of.

These include horrors such as wakening corpses which rise up from the graves scattered around the graveyard, enormous ravens that swoop down on you and giant man-eating plants.

As you progress further towards your goal you will come across other entities such as ghostly knights, little red devils, floating ghouls and many more frightening fiends.

The control of your character is relatively simple with left and right movement, the ability to jump and duck and finally your trusty sword which throws bolts and can destroy some of the creatures.

If you come into contact with any of the overlord's demonic beings you are instantly turned into a pile of bones and will lose one of your five lives.

At first I thought it was just another platform game that you would play a couple of times and then put away, but in fact it is addictive and could well rival Manic Miner and Jet Set Willy as one of the best of this type of game on the market.

Ian Duerden

Sound	6
Graphics	8
Presentation	8
Value for Money	9
Overall	8

## MANDRAGORE

### DUNGEON ADVENTURE

Infogrames' first UK release is a full blooded Dungeons and Dragons adventure.

The game is set in a distant land, Mandragore, where the kindly King

Jorian has been killed by a shooting star.

Out of nowhere arrives Yarod-Nor, assuming power and bringing a reign of terror upon this once peaceful land.

You have to free the country of this tyrant by forming a team of four characters who have to roam the land and solve the mysteries of the nine chateaux, then confront Yarod-Nor in his own chateau.

There are four modes of play, the first being the map mode where plains, forests, hills, swamps, seas, mountains, villages, chateaux and your character are graphically displayed on the main screen.

As you move around the screen you will sometimes enter wandering monster mode in which you have to either fight and kill whatever creature is there or attempt to run away.

The villages are represented by a house symbol. Here you can enter village mode where you can buy, sell, barter and even steal things to help you on your quest.

Finally if you come across a castle you can search the rooms/grounds in an attempt to solve the mystery of the chateau while picking up any treasure you may find and avoiding or fighting any creatures you come across.

To control a character you input his or her number then a verb, and finally the letter which describes the object or creature. It sounds a little complicated, but is in fact quite easy once you've got the hang of it.

The graphics are a little block-like and there is no sound apart from the opening credits. However dungeon-type adventures rely on the imagination of the player, so the sound and graphics are secondary.

The game plays well once you become acquainted with the instructions and is quite a challenge.

The main drawback is that if you are using tapes there is a lot of setting up to do and a fair amount of loading time involved.

But don't let that put you off what is a very good game which will keep even the keenest adventurer happy for many hours. I thoroughly recommend it.

Ian Duerden

Sound	0
Graphics	7
Presentation	9
Value	8
Overall	8

Software reviewed in this section is not always available in Australia or New Zealand at the time of publication but should be so by the time our next issue is published.

Next month's reviews include:

Castle Blackstar  
Fairlight  
Gladiator  
Nick Faldo plays the Open  
Trashman  
Jack the Nipper  
Pacific  
Jewels of Darkness  
Rescue on Fractalus  
Mermaid Madness  
Powerplay  
International Karate  
Knight Games  
Nexus

## SPEECH!

### A WORD IN YOUR EAR

A speech synthesizer is one device that will eventually find its way on to most computer users birthday present lists.

By plugging a little box of electronic wizardry on to the back of your machine you can incorporate the spoken word into your own programs and amaze your friends.

The main problem that I have found with little black boxes is the fact that they soon pile up. If you're not careful you could find a wobbling column of speech synthesizer, RAM expansion and light pen units protruding shakily from the rear of your machine.

All you need then is a program like Cyrus II chess to insist that you remove them all before it will run.

Speech! from Superior Software gets around such problems by doing away with the hardware altogether. It performs its vocal magic using software alone, and all you forfeit is 7.5k of memory.

The program consists of two main sections, the first being the dictionary file. This file contains a list of English words and part words and their phonetic counterparts.

Initially it contains 299 entries and occupies 2,723 bytes of memory.

Superior has provided a simple to use editing program which allows you to add your own words to the dictionary.

The second part of the speech system contains the mechanics of the program, the English text to phonetics translator, the RSX operating system interface, and the phoneme pronouncer.

Speech! can be used via the two bar commands, ISAY and ISPEAK. Both are subject to the normal constraints imposed on RSX commands by Basic versions 1.0 and 1.1.

That is 464 users must send their speech to the program using a string variable, whereas 664 and 6128 owners can enclose their text in quotes directly after the command.

ISAY is a very competent command which makes sense of about 90 per cent of the text presented to it. This is demonstrated by the utility program SAYFILE which is supplied with the package.

SAYFILE will speak the text which is held in a word processor file - it spoke my Amsword documents with no trouble at all.

Should you find that the program has difficulty with any particular word you can overcome this in two ways.

Your first option is to use the editor program supplied and add the new word to the dictionary file. Having

loaded up the required dictionary file you can edit it using any of the editor's 16 commands.

A new word can be added using the insert command. Words are input using the format / English word = Phonemes/, for example:

/ TOMB = TUWM/

The complete dictionary can then be re-saved to tape or disc. Using this technique you can build up specialized dictionaries tailored to suit your own interests.

If the word you wish to pronounce is not one which you are likely to use on a regular basis a less permanent solution is provided by the second speech command, ISPEAK.

This allows you to put together a sequence of phonemes which will give a perfect pronunciation of the word you require.

At your disposal is an armoury of 49 phonemes, and the documentation provides an example word with each phoneme to illustrate its sound.

Here is a program which is a must for every Amstrad user. It can take a bit of getting used to at first, but you will soon get accustomed to its accent. Speech! is cheap, elegant, and has a great deal to say for itself.

John Revis

Sound	9
Graphics	N/A
Ease of Use	9
Value	9
Overall	9



# Enlist the aid of this Pilot to guide you through a quiz

By GORDON MILLS

**I** F ever a program could be described as of mixed parentage, our version of Pilot can. The form books would describe it as "Pilot, out of N. Dealey, via Chuck Carpenter, adapted by Gordon Mills for the Apple and BBC Micro and translated for the Amstrad by the A Team".

It is rather surprising that no version of Pilot seems to be available for the Amstrad CPC464 despite the fact that the Pilot language was originally written for teachers to help them produce computer-aided learning programs.

Perhaps one reason for this is that the "Super" Pilot such as that available on the Apple is even more complicated to learn than Basic.

One advantage of a simple Pilot is that beginners in programming can be more easily initiated into the techniques of simple program structures. The disadvantages are that it does not allow subroutines/procedures nor any form of mathematical calculation.

However, for the teacher who wishes to make up simple quizzes or tutorial material on a question/response basis, this simple Pilot may save considerable time when compared with a similar program written in Basic.

The version described here is based on the Apple version (lines 500-4900 are almost identical). It requires only a few simple command letters with each command followed by a colon as shown in Table I.

After typing in Program II this should be saved. On running the

<b>T:</b>	Text line.
<b>A:</b>	Accept user input.
<b>M:</b>	List of strings to check if there is a Match with the last user input.
<b>J:</b>	Jump to another line.
<b>C:</b>	Clear the screen.
<b>R:</b>	Remark or comment line.
<b>E:</b>	(or END:) End the Pilot program.
<b>S:</b>	Allows a score to be kept (that is requires a number).
<b>DONE</b>	To end writing a program and return to menu.

Table I: Table of commands

program, to write a new one press 1 (new program). Then enter the name which you wish to give it. After a few seconds, a zero followed by a question mark will appear.

This is your cue to start typing in Pilot. It is important not to put in unnecessary spaces and to ensure all letters are capitals.

The text command T: can be used to put any normal keyboard character on screen. Crude graphics can be displayed by using \*, + and a full-stop to build patterns spread over a number of lines.

All the main commands can be

modified by Y or N immediately following the command symbols and before the colon. This causes the command to operate only if:

*The modifier is Y and the last Match is positive,*

Or:

*The modifier is N and the last Match is negative.*

Additional symbols used are \* to precede labels of lines to which the programmer wishes to jump and \$ to precede string labels in lines accepting user input – these can be used later in the program in text lines. There has to be a space between the colon : and the \$ of the label.

Examples of their use are given in Program I or Program Ia and in the instructions contained within Program II.

When using the Match command more than one string can be checked providing each is separated by a comma. Note also that the last symbol on a Match line must always be a single quote (').

As there is a limitation of 33 characters per line, if the number of matches to be checked is greater than can be accommodated on one line, then the continuation Match lines must start with MN: as shown in line 6 of Program 1.

Apart from improving the integrity and structure of the Carpenter program, additional features include:

- A comprehensive menu: on running the program, press 1 to start a new program. All other commands are self-explanatory.

- Improved editing by allowing the insertion and deletion of lines in a Pilot program.

- An option for the Pilot program to be listed on a printer.

- Paged screen listing of the Pilot program.

- Maximum number of Pilot program lines is 400. This could be enlarged further if required by increasing the value of MX in line 130. Alternatively, initialising/editing delays could be reduced by decreasing MX.

- Instructions and commands together with a sample Pilot program included within the Basic program.

- The addition of a Score command, S: which can be followed by any number (positive or negative). If the

Score number is less than 999, then that number is added to the current score: if 999 or greater, the score is reset to zero. At the conclusion of the program, the score is printed out.

● A facility for Pilot programs to be saved to and loaded from cassette.

When writing or editing a program, to tell the micro that you have finished type DONE on the next line (without a colon) and you will be returned to the menu. Don't forget that you will need the command E: to end the program you are writing.

One of the simplest uses for Pilot in education is to present simple knowledge tests. Program I is intended to illustrate how program structures can be developed from the simple to the more complex, thereby providing an ideal teaching medium for beginners.

Question 1 allows one attempt

only. Question 2 shows how a multiple choice question can be written.

Questions 3 and 4 allow the user any number of attempts but it is important to include, as in Question 4, the option (?) to escape from what would be an endless loop if the answer is not automatically given after the user response.

Question 5 shows how to obtain two answers from one question. This involves a much more complex structure than the previous questions. The five types of questions demonstrated in Program I can serve as a model for similar questions in any discipline.

If you find Program I complex try Program Ia. This has only three questions and contains only one jump.

This Mini Pilot interpreter, although written in Basic, is surprisingly fast.

The only delays – of several seconds – occur on initialising and when inserting blank lines or deleting lines (under EDIT).

Also, the modular construction allows the expert programmer to readily experiment with additional commands if desired.

Now is your chance to experiment with a simplified Pilot. It may be that more elaborate versions are on the way, but this is a cheap method of finding out whether you think a more expensive Pilot with graphics and mathematical facilities could be a worthwhile acquisition.

I hope you and your friends – or pupils – don't groan too much at the riddles in Program I.

#### Program I

```

0 #1
1 C:
2 T:IF AN INDIAN WOMAN IS A SQUAW
3 T:WHAT IS AN INDIAN BABY?
4 A:
5 M:PAPPOOSE,PAPPOOSE,PAPPOOS,
6 MN:SQUAWKER,
7 SY:3
8 TY:GOOD
9 T:THE JOKE ANSWER IS SQUAWKER
10 T:NOW PRESS RETURN
11 A:
12 #2
13 C:
14 T:HOW MANY KINGS HAVE BEEN
15 T:CROWNED IN ENGLAND SINCE
16 T:1500 AD ?
17 T:
18 T:A - ONE
19 T:B - 15
20 T:C - 16
21 T:
22 T:TYPE A OR B OR C
23 A:
24 M:A,
25 TN:ANSWER A IS CORRECT BECAUSE
26 T:ONLY JAMES I WAS ALREADY KING!
27 SY:2
28 SN:-1
29 T:PRESS RETURN
30 A:
31 #3
32 C:
33 T:WHAT WAS THE LARGEST ISLAND
34 T:BEFORE AUSTRALIA WAS
35 T:DISCOVERED ?
36 A: $ANSWER
37 M:AUSTRALIA, '
38 TN: $ANSWER IS WRONG
39 SN:-1
40 T:CORRECT IS STILL AUSTRALIA!
41 SY:3
42 TN:NOW PRESS RETURN
43 AN:
44 JN:#3
45 T:GOOD. NOW PRESS RETURN
46 A:
47 #4
48 C:
49 T:IF YOU REALLY DON'T KNOW THE
50 T:ANSWER TO THE NEXT QUESTION,
51 T:TYPE ?
52 #4A
53 T:MAKE ONE WORD FROM THE LETTERS
54 T:NEW DOOR
55 A:
56 M:ONE WORD,ONEWORD, '
57 TY:GOOD. NOW PRESS RETURN
58 AY:
59 SY:3
60 JY:#5
61 M:?, '
62 SY:-1
63 SN:-2
64 TN:TRY AGAIN. PRESS RETURN
65 AN:
66 JN:#4
67 TY:THE ANSWER IS THE 2ND AND 3RD
68 TY:WORDS OF THE RIDDLE!
69 TY:PRESS RETURN
70 A:
71 C:
72 JY:#4A
73 #5
74 R:A QUESTION WITH TWO ANSWERS
75 C:
76 T:WHAT CAN BE ADDED TO NINE TO
77 T:MAKE SIX ?
78 T:TYPE ONE OF TWO POSSIBLE
79 T:ANSWERS
80 A:
81 M:-3,- 3,MINUS3,MINUS 3, '
82 SY:1
83 JN:#5A
84 T:GOOD. NOW TYPE ANOTHER
85 A:
86 M:S, '
87 JN:#5B
88 #5C
89 SY:2
90 T:EXCELLENT. MAX SCORE IS 14
91 T:PRESS RETURN TO FIND YOURS
92 A:
93 EY:
94 #5B
95 T:CORRECT IS S. (S + IX = SIX)
96 #5D
97 SN:-2
98 T:PRESS RETURN
99 A:
100 J:#5
101 #5A
102 M:S, '
103 SN:-2
104 SY:1
105 TN:TRY AGAIN
106 JN:#5
107 T:GOOD. WHAT IS THE OTHER ?

```

```

100 A:
109 M:-3,- 3,MINUS3,MINUS 3,
110 JY:*5C
111 T:CORRECT IS -3 (MINUS 3)
112 J:*5D

```

## Program Ia

```

0 R:QUIZ
1 C:
2 *FIRST
3 T:WHAT IS THE CUBE ROOT
4 T:OF 125 ?
5 A:
6 M:FIVE,5,
7 TN:GO AND GET A CALCULATOR !
8 JN:*FIRST
9 T:WELL DONE, BRAIN BOX.
10 T:PRESS RETURN TO CONTINUE
11 A:
12 C:
13 T:WHICH IS THE ODD ONE OUT?
14 T:A - 1
15 T:B - 4
16 T:C - 8
17 T:D - 14
18 T:E - 19
19 T:
20 T:ENTER A,B,C,D OR E
21 A:
22 M:D,
23 TN:WRONG !
24 TN:THE ODD ONE OUT IS D.
25 TN:IT SHOULD HAVE BEEN 13.
26 TY:CORRECT
27 TY:SEE IF YOU CAN GET THE
28 TY:NEXT QUESTION RIGHT.
29 T:PRESS RETURN TO CONTINUE
30 A:
31 C:
32 T:WHAT IS THE DECIMAL EQUIVALENT
33 T:OF THE BINARY NUMBER 1001101
34 T:(HINT: IT'S +VE)
35 A:
36 M:77,
37 TY:CONGRATULATIONS !
38 TN:HARD LUCK, THE ANSWER IS 77.
39 T:
40 T:END OF TEST.....
41 T:.....
42 T:
43 END:

```

## Program II

```

10 REM MINI PILOT
20 REM (C) Computing with

```

```

30 REM the Amstrad.
100 REM INITIALISE
105 OPENOUT"dummy"
110 MEMORY HIMEM-1
120 CLOSEOUT
125 D0$="":D1$="":CP$="":M$="N"
130 MX=399:LL=33:R=19:JF=0:LF=7:NL=20:
Q=0:SC=0
140 DIM P$(1,MX),I$(1,R),S$(7),M$(9)
150 S$=STRING$(35,"A"):A$=S$
200 REM DATA
210 FOR I=1 TO 9:READ M$(I):NEXT
220 DATA NEW PROGRAM,LIST,EDIT,RUN,REP
LACE,SAVE,LOAD,END,INSTRUCTIONS
230 FOR I=1 TO 7:READ S$(I):NEXT
240 DATA A,T,M,J,C,E,S
499 REM CONTROL SECTION
500 GOSUB 1000:GOTO 600
510 GOSUB 1100:GOTO 600
520 GOSUB 1200:GOTO 600
530 GOSUB 1300:GOTO 600
540 GOSUB 1400:GOTO 600
550 GOSUB 1500:GOTO 600
560 GOSUB 1600:GOTO 600
570 GOSUB 1700:GOTO 600
580 GOSUB 1800:GOTO 600
590 GOSUB 1900:GOTO 600
600 ON Z GOTO 510,520,530,540,550,560,
570,580,590,500
999 REM TITLE
1000 GOSUB 5000:V=1:H=1:GOSUB 5200:GOS
UB 4900
1010 V=2:H=9:GOSUB 5200:PRINT"MINI-PIL
OT (GMM)"
1020 V=3:H=1:GOSUB 5200:GOSUB 4900
1030 FOR I=1 TO 9:V=4+I:H=10:GOSUB 520
0:PRINT I
1040 H=14:GOSUB 5200:PRINT M$(I):NEXT
1050 V=23:H=1:GOSUB 5200:PRINT"PRESS N
UMBER OF ITEM REQUIRED";
1060 GOSUB 6000:IF G<49 OR G>57 THEN G
OTO 1060
1070 Z=VAL(A$):RETURN
1100 REM NEW
1105 LT=LL:Z=1:V=1:H=1:IF A$="EDIT" TH
EN V=3:GOTO 1125
1110 CP$="":GOSUB 5000:C=0:GOSUB 5200:
PRINT"PROGRAM NAME? ";:LL=LF:GOSUB 610
0
1115 CP$=A$:IF CP$="" THEN 1110
1120 LL=LT:GOSUB 4500
1125 V=V+1:FOR A=C TO MX: GOSUB 4800:
IF A=MX THEN GOSUB 4300
1130 PRINT A?";:GOSUB 6100
1135 IF A$="EDIT" THEN Z=3:A=MX:GOTO 1
175
1140 IF A$="DONE" THEN Z=10:A=MX:GOTO

```

```

1175
1145 IF LEFT$(A$,1)="*" OR LEFT$(A$,1)
="!" THEN P$(0,A)="":P$(1,A)=A$:GOTO 1
175
1150 M=0:FOR B=1 TO LEN (A$)
1155 IF MID$(A$,B,1)<D0$ THEN GOTO 11
65
1160 P$(0,A)=LEFT$(A$,B-1):P$(1,A)=MID
$(A$,B+1,LEN(A$)-B+2):B=LEN(A$):M=1
1165 NEXT B:IF M THEN GOTO 1175
1170 V=V+1:GOSUB 5200:PRINT"COMMAND ER
ROR":A=A-1:V=V+1:GOSUB 5200
1175 NEXT A:RETURN
1200 REM LIST
1205 Z=10:IF CP$="" THEN RETURN
1210 GOSUB 5000:PRINT"PRESS S FOR SCRE
EN/P FOR PRINTER";:srm=0
1215 GOSUB 6000:IF G<>80 AND G<>83 THE
N GOTO 1215
1220 GOSUB 5000:IF G=80 THEN GOSUB 650
0
1225 V=1:H=8:GOSUB 5200:PRINT#srm,CP$:
NC=0:FOR A=0 TO MX:NC=NC+1
1230 IF P$(0,A)="" AND P$(1,A)="" THEN
A=MX:GOTO 1255
1235 GOSUB 4800:H=1:GOSUB 5200
1240 IF P$(0,A)<>"" THEN PRINT#srm,A;:
H=7:GOSUB 5200:PRINT#srm,P$(0,A);D0$;P
$(1,A):GOTO 1250
1245 PRINT#srm,A;:H=6:GOSUB 5200:PRINT
#srm,P$(1,A)
1250 IF G<>80 THEN GOSUB 4600
1255 NEXT A:GOSUB 6600:GOSUB 4700:RETU
RN
1300 REM EDIT
1310 GOSUB 5000:PRINT"PRESS I TO INSER
T":PRINT:PRINT:PRINT"OR A TO ALTER/ADD
"
1320 PRINT:PRINT"OR D TO DELETE MORE T
HAN ONE LINE"
1325 PRINT:PRINT"OR R TO RENAME PILOT
PROGRAM"
1330 GOSUB 6000:IF G<>73 AND G<>68 AND
G<>65 AND G<>82 THEN GOTO 1330
1335 IF G=68 THEN GOSUB 4400:GOTO 1390
1340 IF G=65 THEN GOSUB 4400:GOTO 1380
1345 IF G=73 THEN GOSUB 4400:GOTO 1360
1350 GOSUB 5000:PRINT"TO REPLACE "CP$;
" TYPE: ";LT=LL:LL=LF:GOSUB 6100:LL=LT
1355 CP$=A$:Z=10:RETURN
1360 PRINT:PRINT"NO. OF (BLANK) LINES
TO INSERT?"
1365 GOSUB 6100:NB=VAL(A$):IF NB<1 OR
NB>(MX-C) THEN PRINT"INVALID":GOTO 136
5
1370 FOR I=MX-NB TO C STEP-1:FOR I1 =

```

```

0 TO 1:P$(I1,I+NB)=P$(I1,I):NEXT I1:NE
XT I
1375 FOR I = C TO C+NB-1:P$(0,I)="" :P$(
1,I)="" :NEXT
1380 GOSUB 5000:IF C THEN PRINT C-1;"
";P$(0,C-1);D0$:P$(1,C-1)
1385 PRINT C;P$(0,C);D0$:P$(1,C):Z=1:A
$="EDIT":RETURN
1390 GOSUB 5000:PRINT"DELETE FROM ";C;
" TO ? ";GOSUB 6100
1392 ND=VAL(A$):IF ND<C+1 OR ND>MX THE
N GOTO 1390
1394 FOR I= C TO ND:P$(0,I)="" :P$(1,I)
="" :NEXT
1396 FOR I=ND+1 TO MX:P$(0,I-ND-1+C)=P
$(0,I):P$(1,I-ND-1+C)=P$(1,I):NEXT:GOT
O 1380
1400 REM RUN
1405 IF CP$="" THEN Z=10:RETURN
1410 FOR I= 1 TO R:FOR P=0 TO 1:I$(P,I
)="" :NEXT:NEXT
1420 F=0:GOSUB 5000
1430 PRINT:PRINT"RUN OF ";CP$:GOSUB 70
00
1440 FOR A=0 TO MX:IF LEN(P$(0,A))=2 A
ND RIGHT$(P$(0,A),1)<>M$ THEN GOTO 147
0
1450 FOR J= 1 TO 7:IF LEFT$(P$(0,A),1)
=S$(J) THEN ON J GOSUB 3000,3100,3300,
3400,3500,3600,3700
1460 NEXT:IF Q THEN A=MX
1470 NEXT A:Q=0
1480 PRINT:PRINT:PRINT"END OF RUN":PRI
NT:IF SC THEN PRINT"SCORE = ";SC
1490 SC=0:GOSUB 4700:Z=10:RETURN
1500 REM REPLACE STORED PROGRAM
1510 Z=6:GOSUB 6700:Z=10:RETURN
1600 REM SAVE TO CASSETTE
1610 CLS
1650 GOSUB 6800:Z=10:RETURN
1700 REM LOAD FROM CASSETTE
1710 CLS
1760 GOSUB 6900:Z=10:RETURN
1800 GOSUB 5000:PRINT"BYE-BYE":END
1900 REM INSTRUCTIONS
1910 GOSUB 5000:PRINT"?: TYPE WHATEVER
IS ON THIS LINE"
1920 V=2:H=4:GOSUB 5200:PRINT"(MAX=";L
L;"CHARACTERS/LINE)"
1930 H=1:V=4:GOSUB 5200:PRINT"A: WAIT
FOR USER INPUT"
1940 V=6:GOSUB 5200:PRINT"M: MATCH CHA
RACTERS WITH INPUT"
1950 H=4:V=7: GOSUB 5200:PRINT"FROM A:
(, BETWEEN ITEMS AND"
1955 V=8:GOSUB 5200:PRINT"END LINE WIT
H ')"
```

```

1960 H=1:V=10:GOSUB 5200:PRINT"J: JUMP
TO LABEL (STARTING *)"
1970 V=12:GOSUB 5200:PRINT"C: CLEAR TH
E SCREEN"
1975 V=13:GOSUB 5200:PRINT"R: REMARK O
R COMMENT"
1980 V=14:GOSUB 5200:PRINT"S: + OR - N
UMBER TO ADD TO SCORE"
1990 GOSUB 4700:Z=10
2000 GOSUB 5000:PRINT"Y OR N ARE MODIF
IERS AFTER MAIN"
2010 V=2:GOSUB 5200:PRINT"COMMANDS (AN
D BEFORE:) TO GIVE"
2020 V=3:GOSUB 5200:PRINT"ACTION IF LA
ST MATCH WAS"
2030 V=4:GOSUB 5200:PRINT"CORRECT(Y) O
R INCORRECT(N)"
2040 V=6:GOSUB 5200:PRINT"* IS USED BE
FORE EACH LABEL"
2050 V=8:GOSUB 5200:PRINT"$ MUST PRECE
DE A STRING LABEL IN"
2060 V=9:GOSUB 5200:PRINT"A: LINES IF
IT IS TO BE USED"
2070 V=10:GOSUB 5200:PRINT"LATER IN (S
AY) T: LINES"
2080 V=12:GOSUB 5200:PRINT"E: OR END:
STOPS THE PILOT RUN"
2090 V=14:GOSUB 5200:PRINT"DONE IS USE
D TO END INPUT MODE"
2100 GOSUB 4700
2110 GOSUB 5000:PRINT"SAMPLE PROGRAM":
PRINT
2120 PRINT" 0?R:NAME INPUT":PRINT" 1
? *BEGIN":PRINT" 2?T:TYPE YOUR NAME"
2130 PRINT" 3?A: $NAME":PRINT" 4?M:S
UPERMAN,SUPERWOMAN,"
2140 PRINT" 5?TY:DON'T BE FACETIOUS":
PRINT" 6?JY:*BEGIN"
2150 PRINT" 7?T:HELLO, $NAME":PRINT"
8?END:"
2160 PRINT:PRINT"ALWAYS PUT A SPACE BE
FORE $":GOSUB 4700:RETURN
3000 REM ASK
3010 PRINT" ?":P$=P$(1,A):L=LEN(P$
)
3015 K=1
3020 IF MID$(P$,K,1)="$" THEN I$(0,F)=
MID$(P$,K):GOSUB 6100:I$(1,F)=A$:Z$=A$
:F=F+1:K=L:K=K+1:IF K<=L THEN 3020 ELS
E RETURN
3030 K=K+1:IF K<=L THEN 3020 ELSE GOSU
B 6100:Z$=A$:PRINT:RETURN
3100 REM TYPE
3120 P$=P$(1,A):L=LEN(P$):K=1
3125 IF MID$(P$,K,1)="$" THEN 3150
3130 K=K+1:IF K<=L THEN 3125
3135 GOSUB 3220:RETURN
```

```

3150 U=K
3155 IF MID$(P$,U,1)=" " OR MID$(P$,U,
1)="" THEN Y=U:GOTO 3180
3170 U=U+1:IF U<=(L+1) THEN 3155
3175 GOSUB 3220:RETURN
3180 U=Y:V$=MID$(P$,K,U-K)
3190 H=F-1
3200 IF V$<>I$(0,H) THEN H=H-1:IF H=0
THEN 3200 ELSE GOTO 3220
3210 PRINT MID$(P$,1,K-1);I$(1,H);MID$(
P$,U,L-U+1):RETURN
3220 PRINT P$:RETURN
3300 REM MATCH
3310 M$="N":N=1
3320 P$=P$(1,A):L=LEN(P$):FOR E= 1 TO
L
3330 IF MID$(P$,E,1)<>D1$ THEN 3370
3350 IF MID$(P$,N,E-N)=Z$ THEN M$="Y":
E=L:GOTO 3380
3360 N=E+1
3370 IF MID$(P$,E,2)=" " OR MID$(P$,E,
1)="" THEN E=L
3380 NEXT E:RETURN
3400 REM JUMP
3410 Q=0:D=0
3420 IF P$(0,D)<>"" THEN 3440
3430 IF MID$(P$(1,A),1,LL)=MID$(P$(1,D
),1,LL) THEN A=D:D=MX:RETURN
3440 D=D+1:IF D<= MX THEN 3420 ELSE V=
V+1:GOSUB 5200
3450 PRINT:PRINT"JUMP TO UNFOUND LABEL
FROM LINE *":A=Q+1:V=V+1:RETURN
3500 REM CLEAR SCREEN
3510 GOSUB 5000:RETURN
3600 REM END
3610 A=MX:RETURN
3700 REM SCORE
3710 TS=VAL(P$(1,A)):IF TS>998 THEN SC
=0:RETURN
3720 SC=SC+TS:RETURN
4300 GOSUB 5000:PRINT MX;" IS THE LAST
POSSIBLE LINE.":PRINT
4310 PRINT" YOU MAY CONTINUE":PRINT
"BY SAVING THE PILOT PROGRAM"
4320 PRINT"AFTER INPUT OF THIS LINE ";
MX:PRINT:PRINT"THEN AMEND LINE 130 OF
THE"
4330 PRINT"PILOT INTERPRETER TO INCREA
SE":PRINT"THE VALUE OF 'MX' BEFORE REL
DAD"
4340 GOSUB 4700:GOSUB 5000:Z=10:RETURN
4400 GOSUB 5000:PRINT"STARTING AT LINE
? ";GOSUB 6100:C=VAL(A$)
4410 IF (C<1 AND A$<>"0") OR C>MX THEN
4400
4420 RETURN
4500 FOR A= 0 TO MX:P$(0,A)="" :P$(1,A)
```

```

="" :NEXT: RETURN
4600 REM PAGING LIST
4610 IF NC<NL THEN RETURN
4620 NC=0:GOSUB 4700:GOSUB 5000:V=1:H=
0:GOSUB 5200:PRINT CP$:RETURN
4700 REM PRESS SPACE BAR
4710 V=23:H=1:GOSUB 5200:PRINT"PRESS S
PACE BAR TO CONTINUE";
4720 GOSUB 6000:IF G<>32 THEN 4720
4730 G=0:RETURN
4800 REM SET TABS FOR LINE NUMBERS
4810 V=V+1:H=1:IF A<100 THEN H=2
4820 IF A<10 THEN H=3
4830 GOSUB 5200:RETURN
4900 FOR I = 1 TO LL+5:PRINT"*";NEXT:
RETURN
5000 CLS:RETURN
5200 LOCATE H,V:RETURN
6000 A$="":WHILE A$="":a$=INKEY$:WEND:
G=ASC(A$):RETURN
6100 REM INPUT CHECK
6110 S$=""
6120 GOSUB 6000:X=ASC(A$)
6130 S=LEN(S$):IF X=127 AND S THEN S$=
LEFT$(S$,S-1):PRINT CHR$(0)CHR$(32)CHR
$(0):GOTO 6120

```

```

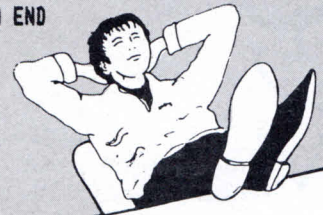
6140 IF X=127 THEN GOTO 6120
6150 IF X=13 THEN PRINT:A$=S$:RETURN
6160 IF X>31 AND X<91 AND S<LL THEN S$
=S$+A$:PRINT A$;
6170 GOTO 6120
6500 REM PRINTER ON
6510 sr#0:RETURN
6600 REM PRINTER OFF
6610 sr#0:RETURN
6700 REM PLACE FILE
6710 GOSUB 5000:PRINT"DO YOU WANT TO D
ELETE THE EXISTING FILE":PRINT:PRINT"O
F "CP$" ?"
6720 PRINT:PRINT"PRESS Y OR N"
6730 GOSUB 6000:IF G<>89 AND G<>78 THE
N GOTO 6730
6740 IF G=78 THEN Z=10:RETURN
6800 REM CASSETTE SAVE
6810 FOR A=MX TO 0 STEP-1:IF P$(0,A)=""
AND P$(1,A)="" THEN NEXT
6820 OPENOUT(CP$)
6830 FOR I=0 TO A
6840 WRITE#9,P$(0,I)
6850 WRITE#9,P$(1,I)
6860 NEXT
6870 CLOSEOUT

```

```

6880 RETURN
6900 REM CASSETTE LOAD
6905 INPUT"PROGRAM NAME? ",CP$:PRINT"W
AIT":GOSUB 4500
6910 OPENIN(CP$)
6915 A=0
6920 WHILE NOT EOF
6930 INPUT#9,P$(0,A)
6940 INPUT#9,P$(1,A)
6950 A=A+1
6960 WEND
6970 CLOSEIN
6980 RETURN
7000 FOR I=1 TO 2000:NEXT:RETURN
8000 END

```



**Give your fingers a rest . . .**  
All the listings from this month's issue are available on cassette.

**FULL AMSTRAD RANGE  
OVER 220 SOFTWARE  
TITLES IN STOCK AT  
ALL TIMES!**

**Mail order and  
Telephone specialists**

**Computer Oasis**

- AMSTRAD DUST COVERS
- DISK WIZARD
- GOLDMARK T.I.E.S.
- MONITOR EXTENSION LEADS
- DISCMASTER
- SPEEDTRANS

**Dealer inquiries welcome.**

**tech-soft**  
Computer Wholesale

324 STIRLING HIGHWAY, CLAREMONT, WA 6010  
Telephone: (09) 385 1885

# Get moving with SPRITES!

By ROLAND WADDILOVE



**A** SPRITE is a character which can move freely about the screen, independently of any other characters, and no matter what the program happens to be doing at the time.

It can move smoothly on and off the screen and pass in front of or behind other characters without disturbing them. They may be controlled either by hardware, software or a mixture of both.

As you can imagine, a sprite is a handy thing to have around when writing games programs, but unfortunately the Amstrad does not support them. However don't despair, it is possible to simulate many of their properties.

The best way would be to use machine code, which is extremely fast and would allow several sprites to be on the screen at once. Main problem there is that machine code is very difficult and time-consuming to write.

Can it be done in Basic? Well if you only want one or two sprites the answer is yes. We can use Basic interrupts to jump to a subroutine at a specified interval and print the character at the graphics cursor, exclusively ORing it with the background so that anything else on the screen is not altered in any way when the sprite is subsequently removed.

The XOR graphic option is set by printing one of the control characters, (see chapter 9 in the manual). After executing:

**PRINT CHR\$(23);CHR\$(1)**  
the actual colour displayed of any

point plotted or drawn is calculated by exclusively ORing the PEN number of the point with the PEN number of the background. Then:

**PRINT CHR\$(23);CHR\$(0)**  
restores normal operation.

To understand how this works and to be able to predict the resulting colour, we need to know a bit about binary. Our Amstrad can help us here, being an expert on the matter.

The 16 colour mode 0 is really the only suitable one for sprites so we will concentrate on it. The pens are numbered 0 to 15 and in binary we need four digits or bits. To find out what the binary equivalent of a number is simply ask your Amstrad!

**PRINT BIN\$(number,4)**  
where 'number' is between 0 and 15.

XOR is a logical operator treating its arguments as bit patterns. To find out what XOR does ask the expert:

**PRINT 0 XOR 0 – gives 0**  
**PRINT 0 XOR 1 – gives 1**  
**PRINT 1 XOR 0 – gives 1**  
**PRINT 1 XOR 1 – gives 0**

So to exclusively OR two numbers compare each bit and write down the result as above.

What is 8 XOR 5? Old know-all will tell you it is 13. Is it right?

**8 .....** 1000  
**5 .....** 0101  
**8 XOR 5 .....** 1101

To find out what 1101 is:

**PRINT &X1101 – gives 13**

So if you plot a point with graphics PEN 8 on top of a point with PEN 5 using the XOR option the result is PEN 13.

What happens if you plot the point

again with PEN 8?

**13 .....** 1101  
**8 .....** 1000  
**13 XOR 8 .....** 0101

The original colour is restored, 0101 is 5.

We can now predict the colour obtained when using the XOR option. After that struggle it's all downhill now – the rest is fairly straightforward.

TAG will enable characters to be printed at the graphics cursor using graphics commands to position and

```
10 REM Program...1
20 MODE 0: BORDER 0
30 FOR i%=0 TO 15
40 READ j%: INK i%, j%
50 NEXT
60 FOR i%=1 TO 7
70 PAPER i%: LOCATE 2*i%+2, 5
80 FOR j%=1 TO 15
90 PRINT " "; CHR$(8); CHR$(10);
100 NEXT
110 NEXT
120 PRINT CHR$(23); CHR$(1)
130 TAG: PLOT -10, -10, 8
140 WHILE 1
150 FOR x%=-30 TO 640 STEP 4
160 MOVE x%, 200: PRINT CHR$(231);
170 FOR delay=0 TO 50: NEXT
180 MOVE x%, 200: PRINT CHR$(231);
190 NEXT
200 WEND
210 DATA 0, 6, 9, 24, 1, 8, 10, 26
220 DATA 7, 6, 7, 24, 7, 8, 7, 26
```

Program 1

colour them. By selecting our pens and inks carefully a character can appear to pass in front of or behind another character.

If a ball is printed using PEN 8 on top of an object printed using PEN 5 the resulting colour is 13. If PEN 13 is set to the same colour as PEN 5 you cannot see the ball, so it appears to be behind the object. But if PEN 13 is set to the same colour as PEN 8 the ball can be seen, so it appears to be in front of the object.

Program I shows how to move a ball across the screen, passing in front of and behind alternate stripes. First the colours are set, then seven stripes

are printed using pens 1 to 7. The XOR option is set, the text and graphics cursors joined and the main loop entered.

Sprites should move independently of the other characters and the program in general once set moving. To do this we need to use interrupts (see chapter 10 in the manual). An interrupt is a signal sent to the processor telling it to drop whatever it is doing and go and do something else, returning when it has finished.

Four timers are available for use by the programmer. The timers are pretty fast, ticking away at 50 times a second. If we write our subroutine to

move our sprite at line 1000, then we can say:

## EVERY 10 GOSUB 1000

This causes the Amstrad to carry out the subroutine five times a second or every fifth of a second, no matter what it is doing. The delay can be any value allowing the sprite to be moved at any speed.

Watch out if you are using very short delays as so much time may be spent moving the sprite that there isn't enough time to carry on with the rest of the program.

Program II shows how we can combine interrupts with our sprite moving routine.

```

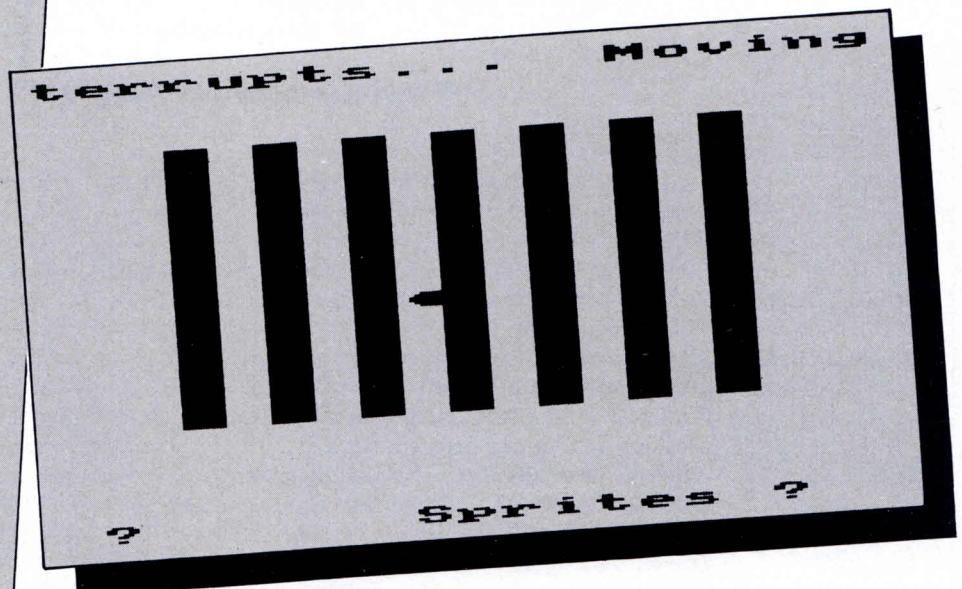
10 REM Program...2
20 MODE 0:BORDER 0
30 FOR i%=0 TO 15
40 READ j%:INK i%,j%
50 NEXT
60 FOR i%=1 TO 7
70 PAPER i%:LOCATE 2*i%+2,5
80 FOR j%=1 TO 15
90 PRINT " ";CHR$(0);CHR$(10);
100 NEXT
110 NEXT
120 x%=-32:xstep%=8:y%=200:ystep%=0
130 PLOT -10,-10,8
140 PRINT CHR$(23);CHR$(1)
150 EVERY 10,3 GOSUB 290
160 a$="Moving a coloured ball using
interrupts... ":b$a$
170 c$=" Sprites ? ":d$c$
180 WHILE INKEY$<>" "
190 IF LEN(b$)<20 THEN b$b$a$
200 LOCATE 1,1:PRINT LEFT$(b$,20)
210 b$=MID$(b$,2)
220 IF LEN(d$)<20 THEN d$c$d$
230 LOCATE 2,25:PRINT RIGHT$(d$,18);
240 d$=LEFT$(d$,LEN(d$)-1)
250 WEND
260 END
270 :
280 REM Interrupt subroutine
290 TAG
300 xx%=(x%+xstep%) MOD 640
310 yy%=(y%+ystep%) MOD 400
320 MOVE x%,y%:PRINT CHR$(231);
330 MOVE xx%,yy%:PRINT CHR$(231);
340 TAGOFF:x%=xx%:y%=yy%:RETURN
350 :
360 DATA 0,6,9,24,1,8,10,26
370 DATA 7,6,7,24,7,8,7,26
    
```

Program II

```

20      Sets the mode and border.
30-50   Sets the ink colours.
60-110  Prints the coloured stripes.
120     Sets the start position and direction.
130     Sets the colour.
140     Sets the XOR graphic option.
150     Enables the interrupt routine.
160-170 Defines the messages.
180-250 Scrolls the messages in opposite directions.
290     Joins the graphics and text cursor.
300-310 Works out the new coordinates.
320     Erases the old ball.
330     Prints the new ball.
340     Separates the cursors, updates position.
    
```

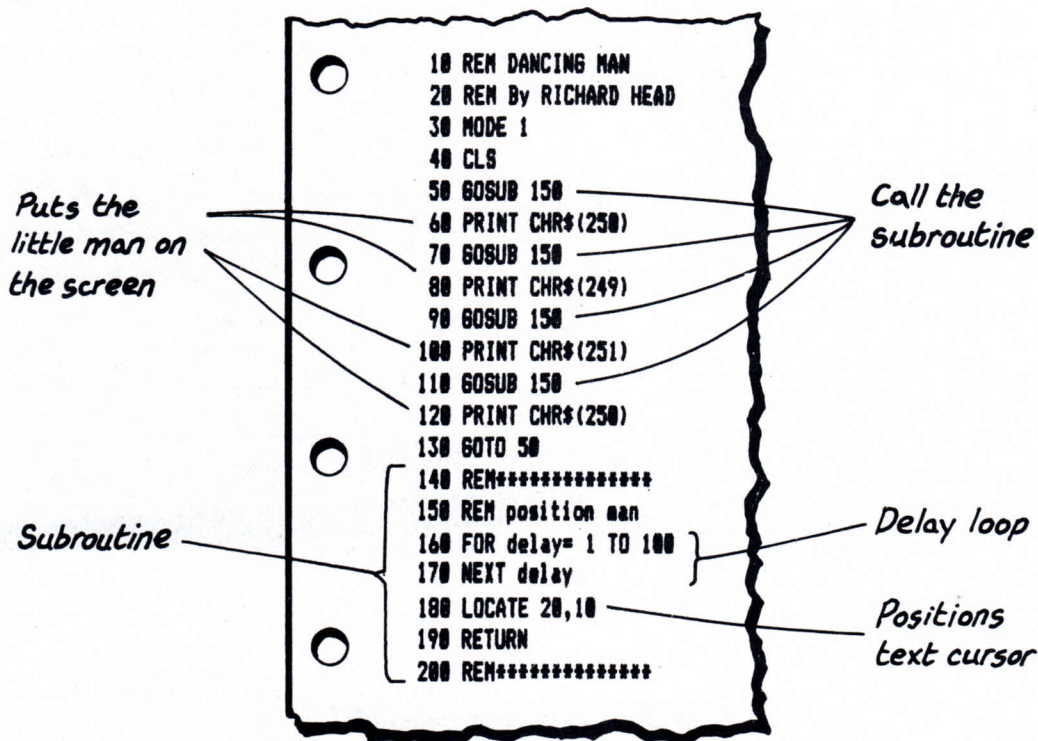
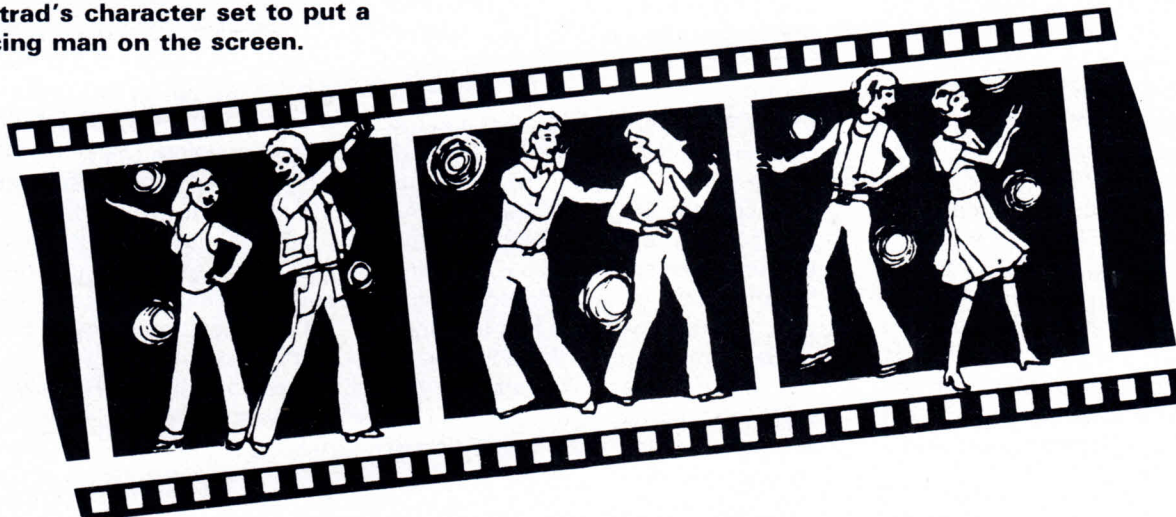
Program II: How it works



# Amstrad Analysis

By Trevor Roberts

**A**NALYSIS is the part of a regular series where short, simple programs come under scrutiny. This month's program uses the Amstrad's character set to put a dancing man on the screen.



- |  |  |
|--|--|
| <p><b>10,20</b> Two REM statements saying what the program is called and who wrote it.</p> <p><b>30</b> Selects Mode 1.</p> <p><b>40</b> Clears the screen.</p> <p><b>50,70,90,110</b> Call the subroutine which starts at line 150 and ends at 190.</p> <p><b>60,80,100,120</b> These lines do the work of printing the little man on the screen. The differing figures in the brackets of the CHR\$( ) produce a different little man.</p> <p><b>130</b> The GOTO sends the program back to line 50, causing the program to repeat endlessly.</p> <p><b>140,200</b> The REMs and asterisks are just used to show where the subroutine is hidden.</p> | <p><b>150-190</b> Form the subroutine.</p> <p><b>150</b> Makes a note of what the subroutine does.</p> <p><b>160-170</b> The FOR . . . NEXT loop just slows things down a little so we have time to see the man.</p> <p><b>180</b> The LOCATE makes sure that the man is printed at the same position each time the subroutine is called. As the different characters overprint each other, so the man appears to dance.</p> <p><b>190</b> The RETURN ends the subroutine, sending the Amstrad back to the line following the line that called the subroutine.</p> |
|--|--|



## TESTing pixels

Can you help me use the Amstrad TEST command?

K.Vance

•TEST is useful when you need to know the colour of a certain pixel.

PRINT TEST(x,y)

returns the PEN colour of the screen coordinate x,y.

## Pondering the PCW

I am considering buying an Amstrad PCW 8256 and would like to ask a few questions on it beforehand.

I am not what you'd call a computer minded person, and the idea of actually buying and owning one of these I find quite a daunting prospect.

I think that the Amstrad PCW 8256 sounds really good value for money - especially as it includes the printer. One thing worries me - the screen.

It seems rather "stripey" when you look across the characters and it was quivering and a bit out of focus at the edges. Is there any way of rectifying this?

I have also heard that LocoScript is complicated for first time users - it does look rather offputting on the screen.

I wondered if there was any way you could alter it to suit your personal needs?

I would also like to know what is meant by LocoScript constantly formatting a file for the printer while it is being edited, and why the text below the cursor position is continuously scanned and reformatted on screen as the cursor passes over it.

LocoScript constantly creates a TEMP file - what is a TEMP file?

And as a final side issue, I wondered if

the word processor could be used for continuous text, for example manuscripts and stories? And if so, how many discs for, say, a 200 page novel?

Catherine Barber

•There's no way you can alter the screen presentation. This out of focus effect doesn't happen in the same place on all machines and on some it doesn't even occur at all.

It's the luck of the draw really, and it's not all that important anyway as the main body of text is centered on the screen away from the edge.

The stripey effect you refer to is the resolution of the monitor and is standard for medium resolution equipment. It's a trade off for a low cost package.

To answer your second point, LocoScript may look complex to the first time user.

Like any sophisticated package it needs some getting used to, but Amstrad have made every effort to make it as user-friendly as possible, and it is not as bad as it may first appear.

There is no way to alter it to suit your own needs. As with most professional software, it is a complete package, protected against copying and therefore inaccessible to the average user.

Most word processors save files to disc, and when it comes to printing them in a particular format you either have to put embedded commands in the text beforehand, or the files are formatted by the computer in some other way as they are about to be printed.

The disadvantage with the first method is that you don't see on screen what you are going to get as a final result.

The disadvantage with the second is that because the computer is being used to format the printing, concurrent applications become slow.

LocoScript formats for the printer all the time. This slows down such functions as save, and search/replace, but speeds

up concurrent applications, such as editing another file or disc management while printing is taking place.

The simplest way to explain a TEMP file is as follows. LocoScript can handle a file of any length, depending on the disc storage space at your disposal, but this cannot be handled by the RAM all in one go.

So a TEMP file is continuously created on disc and is automatically recalled into memory as you scroll through the document.

The problem with working on a lengthy manuscript is that it can't all be held in one file, so you might, for instance, choose to have one file per chapter.

These have to be printed out separately, but because LocoScript has a facility for starting page numbers at any number, the task is made simple.

A 200 page novel, written double spaced on A4 paper, would need at least two double sided discs.

So if you keep back-ups - as any self-respecting writer should do - you'll need a minimum of four discs.

## Missing Memory

I think my Amstrad 664 has a fault. I was using a program which started:

```
10 HM=43903
20 MEMORY 43799
```

On typing RUN, I received the error message 'Memory Full in 20'.

The firmware guide says when no external ROM is fitted HIMEM to &A6F6 contains user-defined characters.

Directly after switch on I peeked and found that memory was actually only clear to &A67B.

Can you please explain what's going on?

H.A. Stubbs

•The CPC664, having a built-in disc drive, sets HIMEM lower than the CPC464. This is because the disc drive needs some of the Amstrad's RAM for its workspace.

On a standard CPC464 HIMEM is 43903 (&AB7F). On a CPC464 with a disc drive or a standard CPC664 or 6128 with no added ROMs HIMEM is 42619 (&A67B).

So when in line 20 you give the command:

```
MEMORY 43799
```

you are attempting to reclaim some of the disc work area. The Amstrad will not let you do this as it could cause an amount of problems - the main one being that you probably wouldn't be able to save your program to disc.

If you attempt to set memory to a value higher than HIMEM was when you turned your Amstrad on, to prevent you accidentally overwriting the disc work area and the foreground and background data, the Amstrad will give the error message Memory Full.

Therefore you should replace lines 10 and 20 with:

```
10 HM=HIMEM-100
20 MEMORY HM
```

This will reserve 100 bytes for your own use starting at HM and then set HIMEM to just below your reserved memory.

Don't forget, though, that each time you run the program, HIMEM will be lowered by 100 bytes. To run it again you should type:

```
GOTO 30
```

## Handling Errors

I'm rather pleased with my Amstrad, much better in all respects than by previous computer, a Colour Genie, which did, however, have an extremely

useful utility called Error Handler.

Once loaded, before entering the program being worked on, all the errors in the problem program were highlighted with ???

Does such a program exist for the Amstrad? The program is published by Gumboot Software, the Colour Genie User Group label.

Can anyone advise of an easy way of transposing the following into Amstrad Basic:

```
PRINT @ (i-1)*120+3*(j-1),c1$
```

I like to try to get programs from other computers to run on the Amstrad, but have little or no success with the above.

What stops me from loading programs from other CP/M machines? I thought this was a universal disc system?

**R. Cross**

•We haven't seen a utility like the one for the Genie. Always start your programs with ON ERROR. Then you can set up your own error handler. Try this program.

```
10 ON ERROR GOTO 80
20 REM Rest of program
30 PRINT "Program ";
40 PRINT "running...";
50 PRINT "ok!"
60 END
70 REM
80 REM Error handler
90 MODE 1
100 PAPER 0:PEN 1
110 PRINT "Error";ERR;"in line";ERL
120 END
```

now add:

```
35 Garbage
```

and you'll see when you run the program that the error is trapped and you are placed in the Basic editor at line 35.

The answer to your translation query is as follows:

```
10 a%=(i-1)*120+3*(j-1)
20 LOCATE a% MOD 40,a%40
30 PRINT c1$
```

Although the Amstrad can run CP/M it uses a slightly non-standard disc system. The software has to be transferred on to 3 inch Amstrad discs.

## Utilizing Potential of the PCW8256

We have recently bought the Amstrad PCW8256 and are thrilled with its word processing capacity, but our reason for buying the PCW was that we wanted both a word processor and a useful domestic computer, so on the computer side of things we are a little disappointed.

It is obvious that there is great potential there - but how to get at it and what to do with it?

Our teenage children are also frustrated seeing numerous games and language programs for sale, and they wonder why we can't use them.

Why is there so little software available for the PCW? It also seems to us that the accompanying instruction book on Basic seems to spend pages saying very little.

I am not a complete beginner to computing, having used a mainframe Vax system at college for the last two years, but having spent literally hours and hours pouring over the book I felt that I have learned little that is of any practical use.

It doesn't really tell you how to construct a program or even suggest any useful programs, and it also assumes a more than basic understanding of computing.

Nothing daunted, I set off to the library to borrow books with simple programs in them written for Amstrad computers - but not the PCW it seems.

Is the Mallard Basic different from the

Basic used by other Amstrad computers, and if so how do you alter programs?

We were delighted to find your magazine and were impressed by its presentation and style but again we have to ask, how relevant will it be to PCW users?

I did try typing in the Easter dating program and to my joy found that it worked, but I hesitated to type in either of the games for fear that after all that work I might get the same results as with the Amstrad games books.

If your published programs do work on the PCW then of course it would be ideal for us to buy your quarterly disc.

I'm sorry if this all sounds rather complaining, but we really are desperate to get on to some real computing. We would value any advice you can offer about available books and software.

**Mrs. Lesley R. Wells**

•Not being new to computing but having spent two years on a Vax is probably a positive disadvantage. Most of my life is spent sitting in front of a Vax terminal, so I speak from experience when I say that you have to get used to being frustrated by the limited power of micros when compared even to the small 11/730, let alone the MicroVax II or 8000 series.

Having said that, I take your point about the lack of games software for the PCW. It was, of course, designed as a business machine (and there's an enormous amount of CP/M business software available), so it's not really surprising that games software houses don't see it as an enormous potential market.

I don't see this situation changing much in the future. Look at the gigantic software base for the IBM PC, then look at the games available for it - far more than on the PCW, admittedly, but only a tiny proportion of the total.

I agree about the Basic manual, but again Basic was included as a sort of PCW extra. Mallard Basic on the PCW does differ in certain respects from other dialects, even from its brothers on other

Amstrad machines.

Look carefully at things like screen coordinates and escape sequences which drive the screen.

Gabriel Jacobs

## *New Interest*

I thought I would write to thank you for a new interest that you brought into our lives. We are two retired people and everyone was telling us that we ought to take up an interest.

We tried a few things, then someone said try a computer. Up till then I thought computers were for the young bright boys. Anyway we thought about it and decided on an Amstrad CPC464. It was one of the best ideas we ever had.

Now our lives are filled with adventures and games. We have learned to play chess and programming in a small way.

So I think a computer is a great therapy of OAPs.

One question - how do you get past the glowing rock in Forest at World's End?

**L.A. Parker**

• TOUCH ROCK and go EAST.

## *Farm Database*

I have a PCW 8256 and urgently require to perform the following tasks:

List calves by breed and ear tag numbers, each to contain source address, date and price paid, up to 16 week rearing details including food costs, vet fees, final weight, to whom sold (fullest details), price, date and dead carcass weight.

Details of children using horse riding facilities, on hourly basis and as resident guests for seven and fourteen day periods. Names, addresses, telephone numbers, ages, income received and outgoing expenses, Dates etc.

Details of sales of bagged milk powder to farmers. Individual farm details, names addresses and telephone numbers, details of contract supply arrangements including delivery frequency, price per tonne, quantities, dates, details of carrier, costs etc.

Please advise if you know of any suitable and recommended programs available to perform any or all of these tasks. Your assistance will be welcome.

**J. Ellis-Vaughan**

•All you require is a database system. These range from simple electronic card-index programs to powerful relational databases such as Dbase III.

Since you don't specify that you need to perform any difficult calculations or complex searches, an electronic filing system should be adequate.

See demonstrations, if you can, of Cardbox, DMS Plus, Delta, FMS, Amsfile and Micropen. But the market is brimming with database programs that will do precisely what you want, so ask about any other available

Our selection would be Cardbox from Caxton. It's easy to use, fairly fast and fully interactive.

## *Missing Screens*

I have a problem with that marvellous game, Roland in Time.

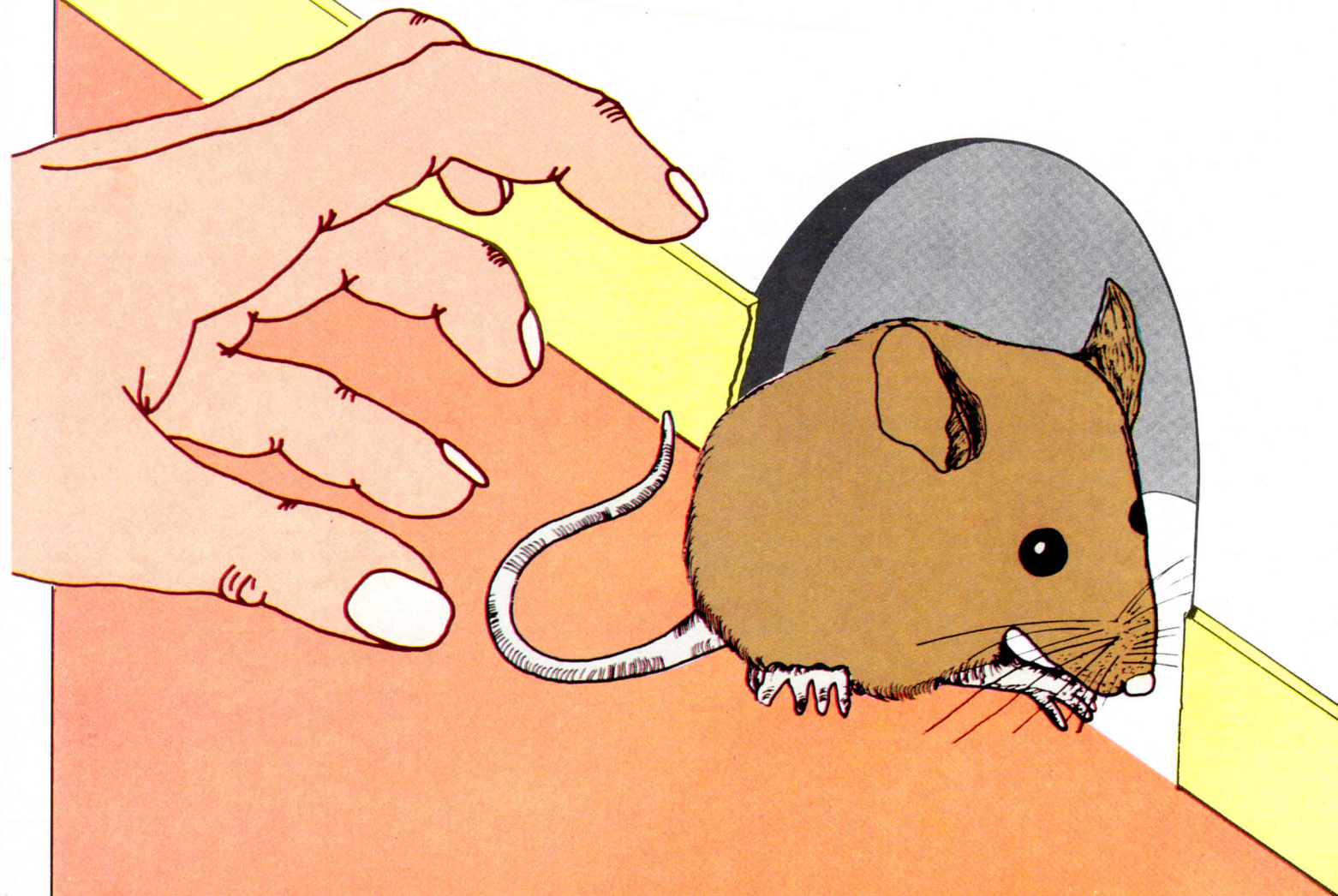
I have trouble finding the following screens: Stepping Out, Dome 9 Spaceport and Where Eagles Dare. Can somebody give me some help?

Also, why is it that I can never get past the 'End of Time' screen at Time Zone 10?

**Tim Burden**

•Sorry, but we're going to have to leave that one to our expert games players out there.

# Pounce on a



## VARIABLES

**ad2,3** Machine code addresses for subroutines.  
**s%** Score.  
**t%** Best score.  
**x% y%** Coordinates of man.  
**g% h%** Coordinates of mouse.  
**mou** Ascii values of mouse character.  
**man** Ascii values of man characters.  
**dir** Mouse direction indicator.  
**i% j%** Coordinate changes for screen locations to be tested.  
**k% m%** Mouse direction change indicators.  
**x\$** Inkey variables for responses.

**n** Loop counter.  
**tit** Ascii data read for title.  
**in** Keyboard or joystick inputs.  
**b,c,d** Machine code routine variables.  
**ch** Ascii value of screen location.

## WINDOWS

**# 0** Whole screen.  
**# 1** Mouse title.  
**# 2** Score and best score.  
**# 3** "GOT HIM" window, when mouse is captured.  
**# 4** Main graphics area.  
**Mouse Game C.P.C.-464 & 664 only.**

# MOUSE !

## Get to grips with GRAHAM REDMAN's challenging chase game

**M**OUSE is a very simple graphics game written mainly in Basic in which you chase a mouse, gradually cornering him in order to pounce.

The mouse moves continuously in a diagonal direction over the whole screen, bouncing off the walls.

Using the joystick or cursor keys you can move yourself around the screen, but as you do you leave a trail of "blocks" behind which act like a wall and will confine the mouse so you can catch it.

To do this all you have to do is place yourself over the mouse. Of course the lowest score is the best in this game, so the secret is to act quickly and pounce. My record so far is 15.

Although this is a Basic program there are two very simple machine code routines. One returns the Ascii value of an input through the keyboard or joystick and the other returns the Ascii value of a character at a particular screen location.

The program is written entirely in character (low resolution) graphics in Mode 1 and uses LOCATE and PRINT for movement.

The two mouse characters and the wall "block" are defined characters (from Ascii 201), while the man characters are from the Amstrad Ascii set. Each time the mouse moves it tests the Ascii value of two squares ahead and to one side of it (see Figure

1). It has to do this in order to know which way to turn when it detects a wall. If, however, it comes to a character diagonally in front it will wipe it out and go through the gap.

Having an unpredictable mouse adds a new dimension to the game, hence the warning in the introduction - "BEWARE: THIS MOUSE CAN GNAW".

Mouse is well documented with REM statements and should not be too difficult to follow.

In order to get quick keyboard response minimum speed key values are used on line 50. However this can cause problems during programming and testing if a break or error occurs due to de-bounce.

To overcome this a break is trapped in order to restore the normal speed key values at line 1350.

Errors are not trapped, so their source can be examined, but the small Enter key has been redefined (line 40 in the program) so that if it is pressed once it will restore normal working of the keyboard.

In the case of a syntax error the large Enter key must be pressed first in order to come out of the edit. Then the small Enter key can be pressed and the edit can then be done.

If all this seems complicated then the best idea is to miss out the SPEED KEY 1,1 on line 50, enter the whole program, debug it, then complete line 50.

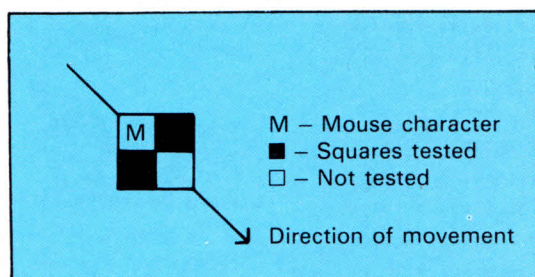


Figure 1: How the mouse moves

```

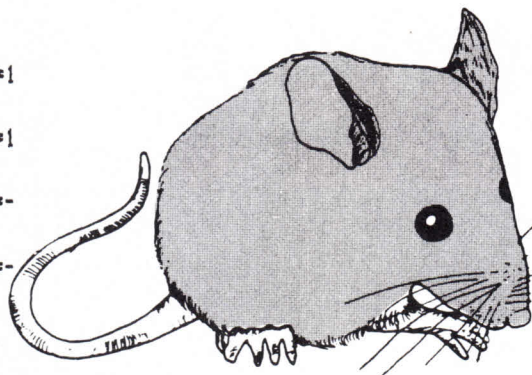
10 REM****MOUSE BY G.REDMAN****
20 REM(c)Computing with the Amstrad
30 REM****INITIAL SETTING UP****
40 ON BREAK GOSUB 1350:KEY 139,"SPEED
KEY 10,5:INK 1,25:PEN 1"+CHR$(13)
50 RANDOMIZE TIME:SPEED KEY 1,1
60 ad2=34999:GOSUB 1080:ad3=35020:GOSU
B 1240
70 REM**MOUSE & MARKER CHARACTERS**
80 SYMBOL AFTER 200:SYMBOL 201,0,0,60,
126,34,12,0,0
90 SYMBOL 202,0,16,60,126,60,40,0,0
100 SYMBOL 203,0,0,60,60,60,60,0,0
110 REM*COLOUR AND WINDOW SETTINGS*
120 BORDER 4:INK 0,0:INK 1,0:INK 2,0:I
NK 3,0:PAPER 0:CLS
130 WINDOW#2,2,39,25,25:WINDOW#4,2,39,
2,23:PAPER#4,0:PAPER #2,2:PEN#2,0
140 t%=10000
150 GOSUB 750:REM**TITLE & INSTR**
160 REM***DRAW BORDER***
170 CLS:INK 3,0:PEN 3:PRINT STRING$(40
,143)
180 FOR n=2 TO 25:LOCATE 40,n:PRINT C
HR$(143):NEXT
190 LOCATE 1,24:PRINT STRING$(40,143)
200 LOCATE 1,1:FOR n=1 TO 25:LOCATE 1,
n:PRINT CHR$(143):NEXT:INK 3,6
210 REM***SCORE WINDOW***
220 CLS#2:LOCATE#2,3,1:PRINT#2,"Your S
core":s%=0:LOCATE#2,20,1:PRINT#2,"Best
Score":LOCATE#2,30,1:IF t%>10000 THE
N PRINT#2,t%
230 REM* SOUND VARIABLES & START**
240 ENT 1,1,12,6,1,-12,6:ENV 1,6,-1,1
250 x%=INT(RND(1)*30)+4:y%=INT(RND(1)*
20)+3:g%=INT(RND(1)*30)+4:h%=INT(RND(1
)*20)+3:dir=INT(RND(1)*4)+1:PEN 2
260 REM***MAIN PROGRAM LOOP***
270 IF dir=1 OR dir=2 THEN mou=202 ELS
E mou=201
280 LOCATE g%,h%:PRINT CHR$(mou)
290 PEN 0:LOCATE#2,13,1:PRINT#2,s%:s%=
s%+1
300 GOSUB 950:REM**MOVE MAN**
310 LOCATE g%,h%:PRINT CHR$(143):PEN 2
320 GOSUB 400:REM**MOVE MOUSE**
330 LOCATE (g%+i%),h%:GOSUB 1190:REM**
LOOK FOR WALL**
340 IF k%=1 THEN m%=1:GOSUB 460:REM**C
HANGE MOUSE DIRECTION**
350 m%=0:k%=0:LOCATE g%,(h%+j%):GOSUB
1190
360 IF k%=1 THEN m%=2:GOSUB 460
370 k%=0:m%=0:GOTO 270
380 REM***END OF MAIN LOOP***

```

```

390 REM***MOVEMENT ROUTINE***
400 IF dir=1 THEN g%=g%+1:h%=h%+1:i%=1
:j%=1
410 IF dir=2 THEN g%=g%+1:h%=h%-1:i%=1
:j%=-1
420 IF dir=3 THEN g%=g%-1:h%=h%-1:i%=-
1:j%=-1
430 IF dir=4 THEN g%=g%-1:h%=h%+1:i%=-
1:j%=1
440 RETURN
450 REM**DIRECTION CHANGE ROUTINE**
460 IF x%=g% AND y%=h% THEN GOTO 570
470 SOUND 1,160,6,15,1,1,0
480 IF dir=1 AND m%=1 THEN dir=4:RETUR
N
490 IF dir=1 AND m%=2 THEN dir=2:RETUR
N
500 IF dir=2 AND m%=1 THEN dir=3:RETUR
N
510 IF dir=2 AND m%=2 THEN dir=1:RETUR
N
520 IF dir=3 AND m%=1 THEN dir=2:RETUR
N
530 IF dir=3 AND m%=2 THEN dir=4:RETUR
N
540 IF dir=4 AND m%=1 THEN dir=1:RETUR
N
550 IF dir=4 AND m%=2 THEN dir=3:RETUR
N
560 REM***MOUSE CAPTURED ROUTINE***
570 IF x%>36 THEN x%=36
580 IF y%>20 THEN y%=20
590 IF x%<3 THEN x%=3
600 IF y%<3 THEN y%=3
610 WINDOW#3,(x%-2),(x%+2),(y%-2),(y%+
2):PAPER#3,3:CLS#3:PEN#3,2:LOCATE#3,2,
2:PRINT#3,"GOT":LOCATE#3,2,4:PRINT#3,"
HIM"
620 ENT 1,15,7,2,15,-7,2
630 FOR n=100 TO 250 STEP 2:SOUND 1,n,
3,6,0,1,0:NEXT
640 SOUND 1,210,50,6,0,0,9
650 IF s%<t% THEN t%=s%-1
660 LOCATE #2,30,1:PRINT #2,t%
670 LOCATE 5,24:PEN 1:PRINT " PRESS SP
ACE BAR WHEN READY "
680 WHILE INKEY$<>" ":WEND
690 REM***REPEAT GAME***
700 CLS#4:LOCATE 5,24:PEN 3:PRINT STRI
NG$(31,143):PEN 2:LOCATE 12,13:PRINT "
Another game ? (Y/N)"
710 x%=UPPER$(INKEY$):IF x%="Y"THEN CL
S #4:GOTO 220
720 IF x%="N" THEN GOTO 1350
730 GOTO 710
740 REM***TITLE & INSTRUCTIONS***
750 WINDOW#1,8,33,2,9:PEN#1,1
760 RESTORE 900

```



```

770 FOR n=1 TO 104
780 READ tit:PRINT#1,CHR$(tit);
790 NEXT
800 PEN#1,2:PRINT#1:FOR n=1 TO 13:PRIN
T#1,CHR$(202);CHR$(201);:NEXT n
810 PEN#4,3:LOCATE#4,1,8:PRINT#4," U
sing the joystick or the cursor keys
you must chase the mouse";PEN#4
,2:PRINT#4,CHR$(202);:PEN#4,3:PRINT#4
,"and capture him."
820 PRINT #4:PRINT#4," To do this
you must move yourman to the sa
me position as the mouse. The trai
l you leave behindwill act like a
wall and confine themouse so you can
capture him."
830 PRINT#4:PRINT#4," Do this as fa
st as you can. The LOWEST score is t
he best."
840 PEN 2:LOCATE 5,23:PRINT "BEWARE, T
HIS MOUSE CAN GNAW !!!!"
850 LOCATE 10,25:PRINT "PRESS THE SPAC
E BAR.";
860 INK 1,14:INK 2,24:INK 3,15
870 WHILE INKEY$<>" ":WEND
880 CLS#4:RETURN
890 REM**DATA FOR TITLE*****
900 DATA 214,32,32,32,32,215,32,32,214
,143,215,32,32,214,32,215,32,32,214,14
3,215,32,32,214,143,215
910 DATA 143,215,32,32,214,143,32,32,1
43,32,143,32,32,143,32,143,32,32,213,2
10,32,32,32,143,210,32
920 DATA 143,213,215,214,212,143,32,32
,143,32,143,32,32,143,32,143,32,32,32,
208,215,32,32,143,208,32
930 DATA 213,32,213,212,32,212,32,32,2
13,143,212,32,32,213,143,212,32,32,213
,143,212,32,32,213,143,212
940 REM***INKEYS/JOY m/c ROUTINE***
950 CALL ad2+1:in=PEEK(ad2):man=248
960 PEN 1:LOCATE x%,y%:PRINT CHR$(203)
970 IF in=8 OR in=242 THEN x%=x%-1:man
=250
980 IF in=9 OR in=243 THEN x%=x%+1:man
=251

```

```

990 IF in=11 OR in=240 THEN y%=y%-1:ma
n=249
1000 IF in=10 OR in=241 THEN y%=y%+1:m
an=249
1010 IF x%>39 THEN x%=39
1020 IF x%<2 THEN x%=2
1030 IF y%>23 THEN y%=23
1040 IF y%<2 THEN y%=2
1050 PEN 2:LOCATE x%,y%:PRINT CHR$(man
):PEN 0:SOUND 1,200,1,2,0,0,0
1060 RETURN
1070 REM***POKE INKEYS m/c***
1080 c=INT(ad2/256):b=ad2-256*c
1090 RESTORE 1160
1100 FOR n=ad2 TO ad2+13
1110 READ d:IF d=999 THEN d=b
1120 IF d=998 THEN d=c
1130 POKE n,d
1140 NEXT n
1150 RETURN
1160 DATA 0,62,0,50,999,998,205,27
1170 DATA 187,208,50,999,998,201
1180 REM**CHECK CHAR POSITION**
1190 CALL ad3+1
1200 ch=PEEK(ad3)
1210 IF ch=143 OR ch=203 THEN k%=1
1220 RETURN
1230 REM**POKE CHAR CHECK m/c***
1240 c=INT(ad3/256):b=ad3-256*c
1250 RESTORE 1320
1260 FOR n=ad3 TO ad3+13
1270 READ d:IF d=999 THEN d=b
1280 IF d=998 THEN d=c
1290 POKE n,d
1300 NEXT n
1310 RETURN
1320 DATA 0,62,244,50,999,998,205,96
1330 DATA 187,208,50,999,998,201
1340 REM*****END ROUTINE*****
1350 CLS:INK 1,25:PEN 1:LOCATE 1,13:PR
INT "Type R to RUN again or E to END p
rogram.":SPEED KEY 10,5
1360 x%=UPPER$(INKEY$):IF x%="E" THEN
END
1370 IF x%="R" THEN RUN
1380 GOTO 1360

```



Give your fingers a rest . . .

All the listings from this month's issue are available on cassette.

# Going for a scroll

**S**CROLLER is a short, simple program that scrolls a message letter by letter from right to left across the screen, repeating it endlessly in a "wrap around" display.

When you run the program it

asks you for the message you want displayed and the line you'd like it to scroll across. The program does the rest, displaying the message until you press a key.

If you want to incorporate Scroller in your own programs it

couldn't be easier. The subroutine that does the work (GOSUB 270) can just be renumbered and merged with your programs. All you have to do is to set up the two variables mentioned in its REM statements.

Pete Bibby

## SUBROUTINES

**GOSUB 90** Checks and accepts the message to be displayed, adding a space to it. It also accepts the line the scrolling is to take place on, again trapping erroneous input.

**GOSUB 270** Does the actual work using two FOR...NEXT loops. The first loop has the message appearing from the right, growing letter by letter, travelling towards the left. This loop stops when the whole message has completed its journey from right to left and is displayed on the screen.

The next loop carries on the scrolling, displaying the message in the centre of the line, giving a wrap around effect. This is achieved by taking a letter off the front of the message, adding it to the end of what's left and then overprinting the original with this new string. The delay loops of lines 400 and 480 can be left

out if you want to see how fast your micro can work.

## VARIABLES

**message\$,  
submessage\$**

Hold the string to be scrolled plus a space which separates the messages.

**aline, subline**

Determine the screen line that the display is to use.

**sublength  
subdisplay\$**

The length of the message. At first this is full of spaces, but each time the first loop cycles a space is removed and a letter added in its place until it eventually holds the whole message. The second loop uses it to contain and display the dismembered message as it wraps around.

**substartposition**

Holds the position of the left end of the display, which is fixed in line 340 so that it's centred on the line.

```

10 REM SCROLLER
20 REM PETE BIBBY
30 REM (c) DATABASE PUBLICATIONS
40 MODE 1
50 GOSUB 90:REM Accept message
60 GOSUB 270:REM Scroll message
70 END
80 REM*****
90 REM Accept and check message
100 LOCATE 1,10
110 PRINT "What message do you want s
    crolled?"
120 LOCATE 1,12
130 INPUT message$
140 IF LEN(message$)>38 GOTO 100
150 REM put a space at the end of the
    message
160 message$=message$+" "
170 LOCATE 1,20
180 PRINT "Which line do you want it
    to appear on?"
190 LOCATE 1,22
200 INPUT aline
210 IF aline<1 OR aline>25 GOTO 170
220 submessage$=message$
230 subline=aline
240 CLS
250 RETURN
260 REM*****
270 REM Scroll the message
280 REM the subroutine needs thses va
    riables when called:
290 REM subline holds the line positi
    on
300 REM submessage$ holds the string
310 REM the delay loops slow things u
    p
320 sublength=LEN(submessage$)
330 subdisplay$=STRING$(sublength," ")
340 substartposition=INT((40-sublenqt
    h)/2+1)
350 REM first loop has string appeari
    ng from right
360 FOR loop=1 TO sublength
370 subdisplay$=LEFT$(subdisplay$,(su
    blength-loop))+LEFT$(submessage$,loop)
380 LOCATE substartposition, subline
390 PRINT subdisplay$
400 FOR delay=1 TO 500:NEXT delay

```

```

410 NEXT loop
420 REM second loop just cycles the o
    riginal message
430 REM until a key is pressed
440 WHILE INKEY$=""
450 subdisplay$=RIGHT$(subdisplay$,(s
    ublength-1))+LEFT$(subdisplay$,1)
460 LOCATE substartposition, subline
470 PRINT subdisplay$
480 FOR delay=1 TO 500:NEXT delay
490 WEND
500 RETURN
510 REM*****

```



Give your fingers a rest...

All the listings from this month's issue are available on cassette.

## SPECIAL OFFER

*Incredible Offer! Subscribe now and save up to \$10.95 on QWERTY the typing tutor.*



Normally priced at \$14.95 on tape and \$26.95 on disk, new subscribers may purchase QWERTY until November 15, 1986 for only \$4 on tape or \$14 on disk.

Existing subscribers may purchase QWERTY at the discount price of \$9 on tape or \$19 on disk.

Non-subscribers may purchase QWERTY at the regular price shown above.

Don't miss out - this offer definitely ends on November 15.





**M**ANY new owners of the CPC464 must already be experienced users, having advanced as I have through other micros. But for the total beginner the idea of typing in listings from magazines must seem daunting.

Your typing may lack the necessary speed and, in particular, the accuracy that is so vital. So here is Letter-Litter – a simple but enjoyable way of getting to know the keyboard

## Learn to use the keyboard

By PHIL TAYLER

of your Amstrad micro.

After selecting how long each round should last a series of letters or other symbols appear individually on

the screen. A little man, already provided by the Amstrad as CHR\$(248) will move automatically in the correct direction, provided that the right keyboard response has been made.

A simple hi-score table is included and the sound is modest, to say the least.

Now you'll need to find something to do with your new-found keyboard skills!

```

10 REM***** letter litter ****
20 REM***** by *****
30 REM***** phil tayler *****
40 REM*****
50 GOSUB 720 : REM set up titles etc.
60 hiscore%=0
70 x%=1:y%=1:score%=0
80 t=TIME
90 RESTORE
100 gametime=(TIME-t)/300
110 GOSUB 340 : REM to select letter
etc.
120 REM main game loop begins here
130 WHILE x%<rx% OR y%<ry%
140 a%=INKEY$:IF (a%<>char% AND a%<>"
") THEN SOUND 1,500,50,7:WHILE SQ(1)
=127:WEND
150 IF a%<>char% THEN 140
160 SOUND 1,100,3:SOUND 1,100,2,0
170 WHILE SQ(1)=127:WEND
180 LOCATE x%,y%
190 IF x%<rx% THEN 250
200 IF x%>rx% THEN 260
210 IF y%<ry% THEN 270
220 IF y%>ry% THEN 280
230 PRINT CHR$(7)
240 GOTO 300 : REM if it makes this l
ine, then x%=rx% and y%=ry%
250 PEN 2:PRINT CHR$(32):x%=x%-1:PEN
3:LOCATE x%,y%:PRINT CHR$(248):GOTO 1
40
260 PEN 2:PRINT CHR$(32):x%=x%+1:PEN
3:LOCATE x%,y%:PRINT CHR$(248):GOTO 1
40
270 PEN 2:PRINT CHR$(32):y%=y%-1:PEN
3:LOCATE x%,y%:PRINT CHR$(248):GOTO 1
40
280 PEN 2:PRINT CHR$(32):y%=y%+1:PEN
3:LOCATE x%,y%:PRINT CHR$(248):GOTO 1
40
290 WEND
300 score%=score%+1
310 IF gametime>timeallowed THEN 450
320 LOCATE x%,y%:PEN 2:PRINT CHR$(32)
:GOTO 100
330 END

```

```

340 REM *****
350 REM ***** initialization ****
360 REM *****
370 MODE 0:INK 0,20:BORDER 2:PAPER 2:
CLS
380 rx%=(RND(1))*20:ry%=(RND(1))*25:1
F rx%>19 OR rx%<1 OR ry%<1 OR ry%>24
THEN 380
390 char%=(RND(1))*90+33:char%=CHR$(c
har%)
400 IF char%<97 AND char%>90 THEN 390
410 LOCATE rx%,ry%:PAPER 1:PEN 3:PRIN
T char%
420 PAPER 2
430 WHILE INKEY%<>"":WEND
440 RETURN
450 REM *****
460 REM **** end of game procedures *
470 REM *****
480 IF score%<hiscore% THEN 640
490 CLS
500 FOR note=1 TO 5:READ a,b,c,d:SOUN
D a,b,c,d:NEXT note
510 DATA 1,478,50,12,1,426,50,12,1,37
9,50,12,1,358,100,12,1,478,100,12
520 PEN 3

```

```

530 LOCATE 5,5:PRINT " Well Done!"
540 LOCATE 5,5:PRINT " Well Done!"
550 LOCATE 5,7:PRINT " You have beate
n the"
560 LOCATE 5,9:PRINT "previous hiscor
e."
570 LOCATE 12,13:PRINT "Please type i
n your"
580 LOCATE 12,15:PRINT "name and then
press"
590 WHILE INKEY%<>"":WEND
600 LOCATE 7,18:PRINT "ENTER"
610 INPUT name$
620 MODE 1
630 hiscore%=score%:score%=0:GOTO 650
640 CLS: LOCATE 12,6:PEN 9:PRINT "You
r score..":score%
650 LOCATE 12,12: PRINT "Today's hisc
ore.":hiscore%:LOCATE 12,15:PRINT " b
y ";name$
660 FOR delay=1 TO 2500:NEXT delay
670 CLS
680 LOCATE 7,12:PRINT "Do you want to
play again?":LOCATE 12,15:PRINT "PRE
SS Y OR N "
690 INPUT answer$

```

### VARIABLES

- x% and y% Coordinates of the man as he moves around the screen.
- rx% and ry% Coordinates of the letter placed on the screen.
- char% Random value which gives a random letter/symbol of char\$.
- score% and hiscore% Need no explaining!
- name\$ Current hi-scorer.

### Notes

The main game loop is enclosed in a WHILE ... WEND loop between lines 130 and 290. The subroutine at lines 720 to 1040 sets up the title screen, etc. The subroutine at lines 350 to 440 selects and places a random letter on screen. The one at lines 450 to 710 is brought into play at the end of the time limit.

```

700 IF INSTR("yY",answer$)<>0 THEN CL
S: GOTO 80
710 END
720 REM *****
730 REM ***** title screen set-up **
740 REM *****
750 MODE 0
760 INK 0,2
770 PAPER 0
780 PEN 10:LOCATE 6,2:PRINT "Welcome
to ":LOCATE 5,4:PEN 5:PRINT "LETTER
LITTER"
790 LOCATE 1,8: PRINT " A letter or an
other"
800 LOCATE 1,10:PRINT"symbol will app
ear"
810 LOCATE 1,12:PRINT"on the screen."
820 LOCATE 1,14:PRINT"Press the right
key"
830 LOCATE 1,16:PRINT"and a small man
will"
840 LOCATE 1,18:PRINT"appear. Keep th
e key"
850 LOCATE 1,20:PRINT"pressed and he

```

```

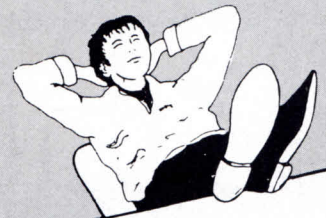
will"
860 LOCATE 1,22:PRINT"collect the let
ter."
870 LOCATE 4,25:PRINT "Press <SPACE>"
880 IF INKEY(47)<>0 THEN 880
890 CLS
900 LOCATE 1,1: PEN 5:PRINT "Don't fo
rget to use"
910 LOCATE 1,3: PEN 5:PRINT "the shif
t key when"
920 LOCATE 1,5: PEN 5:PRINT "necessa
ry."
930 LOCATE 1,7
940 PEN 1:PRINT "You can decide the"
950 LOCATE 1,9:PEN 1:PRINT "game leng
th..."
960 LOCATE 1,13:PEN 5:PRINT "a) 30 se
conds "
970 LOCATE 1,15:PEN 5:PRINT "b) 1 min
ute "
980 LOCATE 1,17:PEN 5:PRINT "c) 2 min
utes "
990 LOCATE 1,22:PEN 1:PRINT "press a,
b or c"

```

```

1000 WHILE INKEY$<>" " :WEND
1010 LOCATE 1,23:PRINT
1020 INPUT letter$:IF INSTR("aAbBcC",
letter$)=0 GOTO 1010
1030 IF letter$="a" OR letter$="A" TH
EN timeallowed =30 ELSE IF letter$="b
" OR letter$="B" THEN timeallowed =60
ELSE IF letter$="c" OR letter$="C" T
HEN timeallowed =120
1040 RETURN

```



Give your fingers a rest...

All the listings from this month's issue are available on cassette.

AMSTRAD CPC464 SOFTWARE



**ADDICTIVE....**

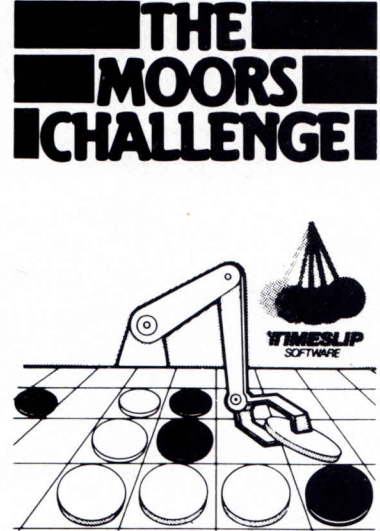
AMSTRAD CPC464 SOFTWARE



**PERPLEXING....**

WHAT MORE DO YOU WANT?  
All programs \$9.95 inc. p & p from

AMSTRAD CPC464 SOFTWARE



**CHALLENGING.**

**STRATEGY SOFTWARE**

P.O. BOX 5000,  
GLEN WAVERLEY, 3150.

Dealers Enquiries Welcome

## BOOK OFFER

You've read the review - now buy the book!

# AMSTRAD COMPUTING

IAN SINCLAIR



**WITH THE CPC464**

### HOW TO MASTER YOUR AMSTRAD

Just as you would find it difficult to learn a foreign language from a dictionary, you will probably find it difficult to master your Amstrad from the manual alone. This book takes you step by step, in detail, through the commands. You are encouraged to try out your ideas, write your own programs and extend your capabilities. Soon you will see how much more your computer can do for you. You will find how business calculations are worked out, how vivid displays are obtained, how attention-catching sounds can be arranged. This book, hand in hand with the encyclopaedic manual, will lead you to greater enjoyment and a better return from your investment in the Amstrad.

The ideas and programs presented in this book are equally applicable to the CPC 664 and 6128 as they are to the original Amstrad CPC 464.

AMSTRAD COMPUTING

\$17.95 INCLUDING HANDLING.

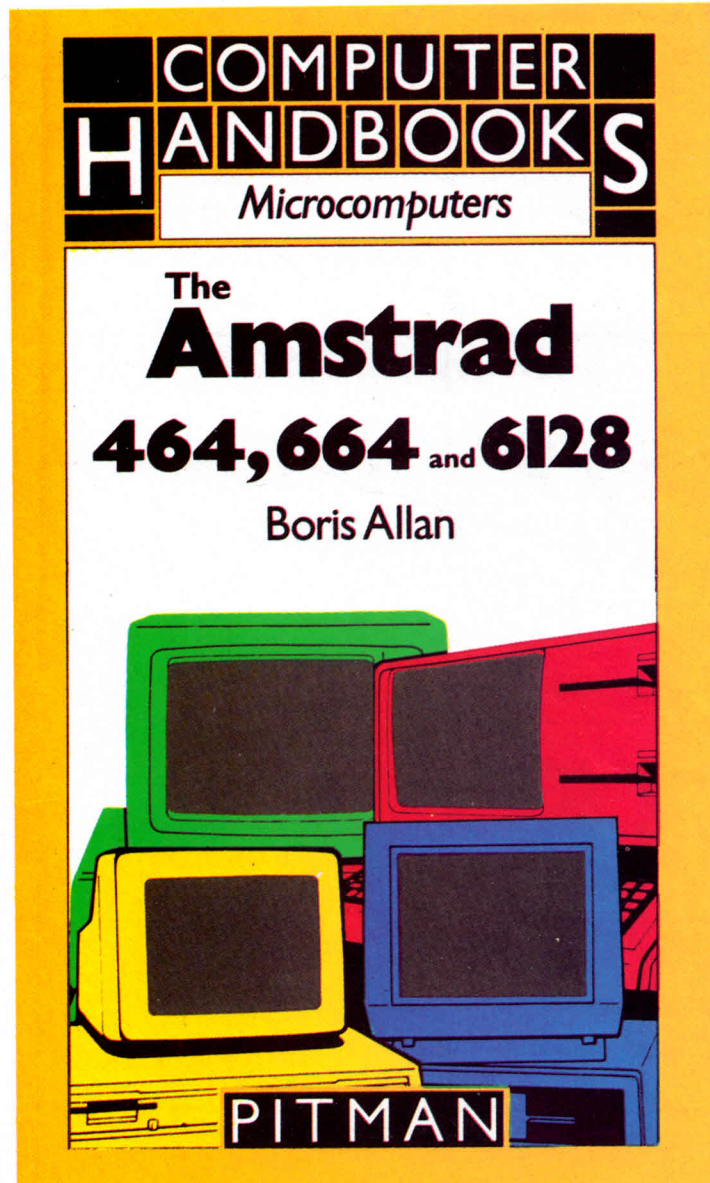
PLEASE USE THE ORDER FORM ON PAGE 57.

**POWER IN YOUR POCKET!****CPC HANDBOOK**

This Handbook shows how the Amstrad CPC computers can be programmed in a rational manner. The intention is to produce a practical working guide to the CPC computers; not, however, by giving listings of many small examples, but by showing how to construct substantial systems. It is for this reason that detail is concentrated on two main applications: File handling (and statistics), and programming using turtle graphics.

The orientation is towards the promotion of effective programming strategies. Though the study of programming strategies uses Locomotive Basic (using Tape or Disk), such strategies have relevance for programming in general, and for CP/M and LOGO in particular. Owners of the CPC 6128 will find all the elements of Basic programming herein, apart from extensions to Basic with bank switching.

The Handbook also gives a short account of both CP/M and LOGO.



**AMSTRAD COMPUTER HANDBOOK  
FOR THE CPC 464,664, 6128  
\$9.95 INCLUDING HANDLING.**

**PLEASE USE THE ORDER FORM ON PAGE 57.**

<b>\$50</b>	<b>\$25</b>	<b>\$45</b>
	<b>\$70</b>	
<b>SPACE TO LET</b>	<b>\$100</b>	<b>\$25</b>
<b>\$15</b>		<b>\$25</b>
<b>\$15</b>		

**\$150**  
**(03) 560 4324**

As usual, all the latest news direct from the U.K.

Gandalf the adventure wizard is back with hints and tips galore for Amstrad adventurers.

More puzzles with Aleatoire

We feature a machine code screen clock routine as well as a fast fill that can be used to fill any shape.

Evaluations of the latest software from Knight Games to Gladiator, Pacific to Mermaid Madness.

More games to key in including one for PCW owners

In our business section we continue to look at the Airo and other PCs, an update on Pocket Worstar, an in-depth review of DR Draw and we put financial software under the microscope.

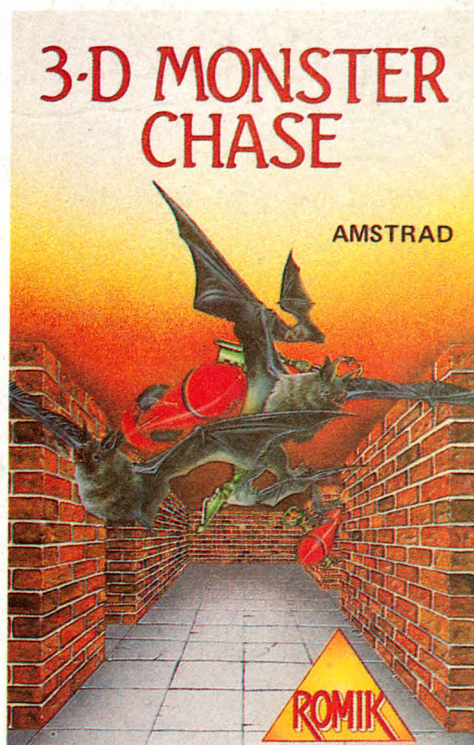
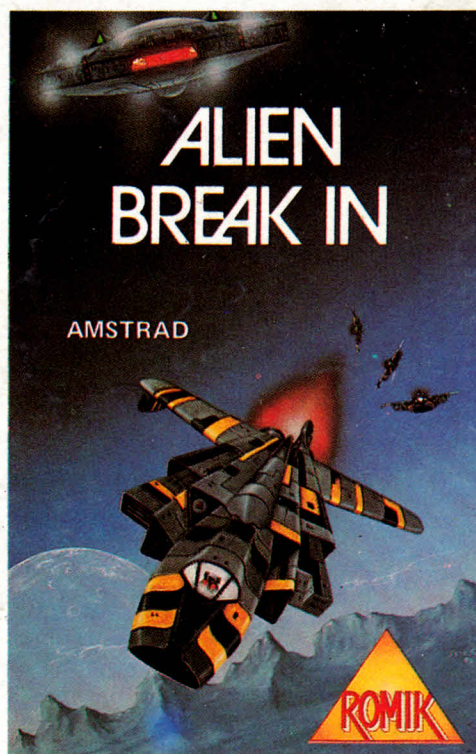
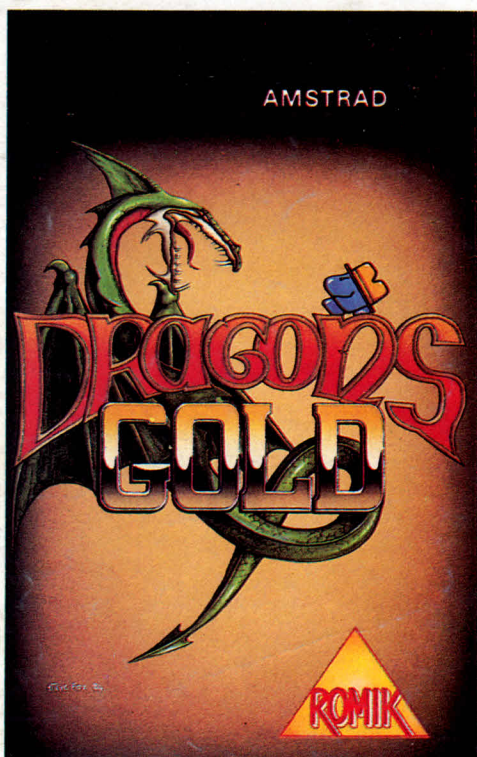
Can DK'Tronics silicon disc replace a second drive? - you'll find out next month.

We compare two of the current leaders in the Forth language stakes.

As usual, we'll feature the best of your letters in our regular Postbag spot. You could learn lots reading this!

In short, don't miss the November Issue of CWTA - it's our best yet!

# FOUR OF A KIND



\$9.95 Each on cassette  
\$29.95 for all four on cassette  
\$39.95 for all four on one disk

Strategy Software, Box 5000, GlenWaverley, Victoria 3150  
(03) 560 4324  
Bankcard & Mastercard welcome