60p

# YOUR COMPUTER

JANUARY 1983 BRITAIN'S BIGGEST-SELLING HOME COMPUTER MAGAZINE Vol. 3 No. 1

How tomorrow's technology will turn today's micros into antiques

Reviews: Graphics tablets Spectrum software

BBC turtle graphics

Plus plenty of games and features for the ZX-81, Dragon, Vic, Atom and Atari

Win an Oric

(continued from page 111)

```
4000dIF H=1;GOS.2020;R.
4002IF H=2;GOS.2030;R.
4004IF H=3;GOS.2040;R.
4006IF H=4;GOS.2010;R.
4008IF H=5;GOS.2060;R.
4010IF H=6;GOS.2070;R.
4012IF H=7;GOS.2080;R.
4014IF H=8;GOS.2050;R.
4016IF H=9;GOS.2120;R.
4018IF H=10;GOS.2090;R.
4020IF H=11;GOS.2100;R.
4022IF H=12;GOS.2110;R.
4023IF H=13;GOS.2160;R.
4024IF H=14;GOS.2130;R.
4026IF H=15;GOS.2140;R.
4027IF H=16;GOS.2150;R.
4028IF H=17;G=G-£100;GOS.2200;R.
4030IF H=18;GOS.2170;R.
4032IF H=19;GOS.2180;R.
4034IF H=20;GOS.2190;R.
4036IF H=21;GOS.2240;R.
4038IF H=22;GOS.2210;R.
4040IF H=23;GOS.2220;R.
4042IF H=24;GOS.2230;R.
4050R.
5000eIF H=1;GOS.2040;R.
5002IF H=2;GOS.2010;R.
5004IF H=3;GOS.2020;R.
5006IF H=4;GOS.2030;R.
5008IF H=5;GOS.2080;R.
5010IF H=6;GOS.2050;R.
5012IF H=7;GOS.2060;R.
5014IF H=8;GOS.2070;R.
5016IF H=9;GOS.2100;R.
5018IF H=10;GOS.2110;R.
5020IF H=11;GOS.2120;R.
5022IF H=12;GOS.2090;R.
5023IF H=13;GOS.2140;R.
5024IF H=14;GOS.2150;R.
5026IF H=15;GOS.2160;R.
5027IF H=16;GOS.2130;R.
5028IF H=17;G=G-£100;GOS.2180;R.
5030IF H=18;GOS.2190;R.
5032IF H=19;GOS.2200;R.
5034IF H=20;GOS.2170;R.
5036IF H=21;GOS.2220;R.
5038IF H=22;GOS.2230;R.
5040IF H=23;GOS.2240;R.
5042IF H=24;GOS.2210;R.
5050R.
6000fIF H=1;GOS.2150;R.
6002IF H=2;GOS.2240;R.
6004IF H=3;GOS.2110;R.
6006IF H=4;GOS.c;G=G-£200;GOS.2180;R.
6008IF H=5;GOS.2130;R.
6010IF H=6;GOS.2200;R.
6012IF H=7;GOS.2090;R.
6014IF H=8;GOS.c;G=G-£200;GOS.2220;R.
6016IF H=9;GOS.2010;R.
6018IF H=10;GOS.2170;R.
6020IF H=11;GOS.2050;R.
6022IF H=12;GOS.2210;R.
6023IF H=13;GOS.2030;R.
6024IF H=14;GOS.2230;R.
6026IF H=15;GOS.2070;R.
6027IF H=16;GOS.2190;R.
6028IF H=17;GOS.2140;R.
6030IF H=18;GOS.c;G=G-£200;GOS.2080;R.
6032IF H=19;GOS.2120;R.
6034IF H=20;GOS.2020;R.
6036IF H=21;GOS.2160;R.
6038IF H=22;GOS.c;G=G-£200;GOS.2040;R.
6040IF H=23;GOS.2100;R.
6042IF H=24;GOS.2060;R.
6050R.
7000gIF H=1;GOS.2090;R.
7002IF H=2;GOS.2200;R.
7004IF H=3;GOS.2130;R.
7006IF H=4;GOS.c;G=G-£200;GOS.2220;R.
7008IF H=5;GOS.2110;R.
7010IF H=6;GOS.2240;R.
7012IF H=7;GOS.2150;R.
7014IF H=8;GOS.c;G=G-£200;GOS.2180;R.
7016IF H=9;GOS.2070;R.
7018IF H=10;GOS.2230;R.
7020IF H=11;GOS.2030;R.
7022IF H=12;GOS.2190;R.
7023IF H=13;GOS.2050;R.
7024IF H=14;GOS.2170;R.
7026IF H=15;GOS.2010;R.
7027IF H=16;GOS.2210;R.
7028IF H=17;GOS.2100;R.
7030IF H=18;GOS.c;G=G-£200;GOS.2040;R.
7032IF H=19;GOS.2160;R.
7034IF H=20;GOS.2060;R.
7036IF H=21;GOS.2120;R.
7038IF H=22;GOS.c;G=G-£200;GOS.2080;R.
7040IF H=23;GOS.2140;R.
7042IF H=24;GOS.2020;R.
8000aF.V=0TO7;S?G=R?V;G=G+32;N.V;R.
8010bF.V=0TO7;S?H=R?V;H=H+32;N.V;R.
8100c!R=£00000000;R!4=£000000;H=G-£120;GOS.b
8110R.
8200aCLEAR0;P.$12;P."you crashed"
9999LINK £FFE3;G.15
```

# Accelerator

*Bob Boffin,*
*Woking,*
*Surrey.*

DRAGON

DRAGON OWNERS may be interested in this simple way to speed up their Basic programs.

The technique uses the ability of the Motorola 6809E microprocessor to run in three different modes. The first mode uses a clock rate of 0.9 MHz and is the one used by the Dragon by default. The second mode uses a clock rate of 1.8 MHz but the processor does not output addresses for the video chip so no display is produced, hence this mode is not very useful. It is the third mode which is of interest. In this mode the processor runs at either 0.9 MHz or 1.8 MHz, depending on the address being accessed. Addresses in the range 0000-7FFF hexadecimal or FF00-FF1F hexadecimal are accessed at 0.9 MHz. All other addresses are accessed at 1.8 MHz.

Since the Basic interpreter is located starting at address 8000 hex, if dual rate is selected it will run at 1.8 MHz except when it is accessing RAM. This gives a very significant improvement in Basic performance.

Selection of the processor mode is simple.

```
POKE &HFFD7,0   will switch dual rate on
POKE &HFFD6,0   will switch dual rate off
```

Pressing the Reset button on the side of the Dragon will also Reset the processor mode to normal. Any value may be Poked. It is the act of writing to the location which toggles the switch. If you Peek at these locations they always return the same result.

When you try this on your Dragon the first thing you will notice is that the cursor blinks very much faster. This is an easy way to tell which mode you are in.

A few simple benchmarks will show that the Dragon is now running appreciably faster. The biggest improvement will be found in number-crunching programs where most of the accesses will be at the faster rate. These can show up to 70 percent improvement.

There are some side-effects to using the dual rate. The notes produced by the Sound command will be about an octave higher. Do not use CLoad, CSave or any commands which use the cassette interface while in dual rate as the port address used is accessed at the higher rate. You can switch between the two rates within your Basic program if necessary.

# Array sort

*Alan Stevens,*
*Alvaston,*
*Derbyshire.*

MZ-80K

THIS BASIC listing of the Quicksort algorithm is for sorting an array, A(x), of elements, x, into order. The first part of the program simply generates 100 random numbers for the sort routine to work on. The sort subroutine itself consists of lines 1000-1130.

One of the interesting features of the program is that the sort subroutine calls itself recursively — a feature widely regarded as being not possible in Basic.

Those who believe that recursion is not possible in Basic are perhaps being confused by the fact that Basic does not support local variables — recursion is generally much more useful when local variables are available within the subroutine. The other interesting feature of my program, therefore, is the simulation of local variables by the use of subscripted variables. The subscript, S, is increased by one at each entry to the subroutine, and decreased by one at each exit.

*(continued on next page)*

(continued from previous page)

The program takes about 17 seconds to sort 100 random numbers, about 13 seconds for 100 numbers in reverse order.

The number of nested subroutines required depends on the ordering of the elements in, and the size of the array. For 100 randomly arranged numbers the average number of nested subroutines is 12 — found from experience, though occasionally, as many as 15 are needed.

The MZ-80K allows 15 nested subroutines, which is why arrays L(S) and HI(S) are Dimensioned as shown in line 20. The program may be simply modified for machines which support M nested subroutines by Dimensioning L(M+1) and HI(M+1).

If 15 (or M) nested subroutines are not sufficient, the following lines may be added to the program to extend its range:

```
105 F=0
165 IF F=1 THEN 100
1005 IF S>15 THEN F=1
1006 IF F=1 THEN 1130
```

These lines effectively reinitialise the sort which restarts with an already partly sorted array.

```
10   REM N=NUMBER of ELEMENTS TO BE SORTED:N=100
20   DIM L(16),HI(16),A(N)
30   REM GENERATE AND PRINT N RANDOM INTEGERS
40   FOR R=1 TO N
50   A(R)=INT(N*RND(1))
60   PRINT A(R);
70   NEXT R
80   PRINT:PRINT
90   REM  SET INITIAL CONDITIONS AND CALL QUICKSORT
100  LO=1:HI(1)=N:S=0
110  GOSUB 1000
120  REM PRINT SORTED ARRAY
130  FOR R=1 TO N
140  PRINT A(R);
150  NEXT R
160  PRINT
170  END
900  REM  QUICKSORT SUBROUTINE
1000 S=S+1
1010 L=LO:H=HI(S)
1020 M=A((L+H)/2)
1030 IF A(L)<M THEN L=L+1:GOTO 1030
1040 IF A(H)>M THEN H=H+1:GOTO 1040
1050 IF L>H THEN 1100
1060 T=A(L):A(L)=A(H):A(H)=T
1070 L=L+1:H=H-1
1080 GOTO 1030
1090 REM SET CONDITIOMNS AND RECALL QUICKSORT IF NECESSARY
1100 L(S)=L
1110 IF LO<H THEN HI(S+1)=H:GOSUB 1000
1120 IF HI(S)>L(S) THEN LO=L(S):HI(S+1)=HI(S):GOSUB 1000
1130 S=S-1:RETURN
```

# Blitz

*Shingo Sugiura,*
*Tokyo,*
*Japan.*

BBC

THIS PROGRAM is for the BBC Micro model B. It is similar to the Vic-20 game Blitz, although there are a few extras in this implementation. You are in control of an aeroplane rapidly losing altitude. Below you are skyscrapers which you must bomb and flatten enough to land. It plays a nice little tune when you succeed but when you crash the effect is spectacular. If you find the game too fast or too slow, change the speed value given in line 350.

```
30        SETS F0 KEY
40~140    DEFINES CHARACTERS
150       SOUND FOR BOMBING
160,170   INSTRUCTIONS
180       DEFINES BOMB$
190~300   SET SCREEN
310       INITIALISE STRINGS AND VARIABLES
320       TESTS FOR SPACE BAR
330       PRINTS AEROPLANE
340       CHECK IF PLANE HAS LANDED
350       SLOW DOWN THE PLANE
360       CHECKS IF PLANE HAS CRASHED
370       CALCULATES PLANE'S NEXT POSITION
380       SEE IF BOMB IS ON THE SCREEN
390       PRINT BOMB
400       SEE IF BOMB HAS HIT BUILDINGS
410       EMPTY KEYBOARD BUFFER
420       GOTO320
430~550   INSTRUCTIONS PROCEDURE
560~600   CRASHING PROCEDURE
610~630   BOMB PROCEDURE
640~660   DESTRUCTING PROCEDURE
670~720   INITIALISING STRINGS AND VARIABLES
730~740   POSITION CALCULATING PROCEDURE
750~790   LAND PROCEDURE
800~870   SMASHING PROCEDURE
```

```
10 REM BLITZ
20 REM (C) SHINGO SUGIURA
30 *KEY0"RUN!M"
40 VDU23,224,90,126,90,126,90,126,90,126
50 VDU23,225,90,126,90,90,126,90,90,126
60 VDU23,226,102,126,102,126,102,126,102,126
70 VDU23,227,0,0,24,24,36,126,90,126
80 VDU23,228,60,60,24,24,60,90,126,90
90 VDU23,229,24,24,24,24,60,126,102,126
100 VDU23,230,0,32,112,248,252,127,63,0
110 VDU23,231,0,0,0,1,241,255,253,1
120 VDU23,232,126,60,24,60,126,126,60,24
130 VDU23,233,32,124,254,127,63,31,31,31
140 VDU23,234,0,8,4,102,249,248,252,252
150 ENVELOPE1,1,11,-6,1,10,30,60,127,0,0,-127,126,0
160 MODE7
170 PROCINSTRUCTIONS
180 IF DL=1 THEN DESTRUCT$=" "+CHR$(10)+CHR$(8)+" " ELSE DESTRUCT$=" "+CHR$(10
)+CHR$(8)+" "+CHR$(10)+CHR$(8)+" "
190 MODE2
200 COLOUR134:CLS
210 VDU23;10;32;0;0;0;
220 FOR BUILD%=2 TO 18
230 COLOUR0
240 A=RND(3)+223
250 FOR HEIGHT%=29 TO RND(C*4)+(20-C*2) STEP-1
260 PRINT TAB(BUILD%,HEIGHT%);CHR$(A)
270 NEXT HEIGHT%
280 PRINTTAB(BUILD%,HEIGHT%+1);CHR$(A+3)
290 SOUND1,-15,RND(200),1
300 NEXT BUILD%
310 PROCINIT
320 FIRE$=INKEY$(0):IF FIRE$=" "THEN PROCBOMB
330 COLOUR1:PRINTTAB(X,Y);AERO$:SOUND0,-5,100,2
340 VDU26:IF POINT((X+2)*64+32,(30-Y)*32)=0 AND POINT((X+3)*64+32,(30-Y)*32)=0
AND POINT((X+4)*64+32,(30-Y)*32)=0 THEN PROCLAND
350 FOR SPEED=1 TO 80:NEXT SPEED
360 IF POINT((X+4)*64+32,(31-Y)*32)=0 THEN PROCCRASH:PROCDROP
370 PROCMOVE
380 IF FIRE=0 THEN 320
390 COLOUR3:PRINTTAB(XB,YB);BOMB$
400 YB=YB+1:IF YB>=28 OR POINT(XB*64+32,(30-YB)*32)=0 THEN PROCDESTRUCT ELSE 3
30
410 *FX15,0
420 GOTO320
430 DEFPROCINSTRUCTIONS
440 PRINTTAB(13,10);CHR$(141);CHR$(132);"BLITZ"
450 PRINTTAB(13,11);CHR$(141);CHR$(132);"BLITZ"
460 PRINTTAB(2,13);CHR$(135);"YOU MUST DESTROY THE BUILDINGS FLAT"
470 PRINTTAB(3,14);CHR$(135);"ENOUGH TO LAND YOUR AEROPLANE"
480 PRINTTAB(5,18);CHR$(135);"PRESS SPACE BAR TO DROP BOMB"
```