

# Getting Started



# Reading the Listings

Note that all programs in this book have been printed out using the Program Lister program (see pages 18-25). The program formats the listings so that they are easier to read than would otherwise be the case.

One of the most useful features of the BBC Micro's red function keys is to allow entry of teletext control codes and user-defined graphic symbols. These are used by pressing one of the keys at the same time as pressing either SHIFT or CONTROL. The codes produced by these keys produce some weird effects when printed out, so to avoid confusion, it is generally not advisable to print these. The Program Lister program gets round this by converting the codes into unambiguous forms which can be easily printed and understood. The actual output is a direct representation of the keys which need to be pressed to produce the effect desired.

There is another common problem when entering program listings from a book or magazine. It is often hard to tell how many spaces are to be included within a string. The Program Lister counts the spaces in the program for you, and if there are more than two in a row, it outputs the number of spaces required, rather than just printing a vast empty string. One or two spaces are printed directly. The output from the program we've mentioned is enclosed within square brackets, so you'll know that anything within these brackets refers to information produced by the Program Lister and must be entered with care. Here's how to work out what they mean.

[fsn]	means	shift + function key 'n'
[fcn]	means	control + function key 'n'
[chn]	means	character with ASCII code n (see below)
[spcn]	means	n spaces

**[chn]**

Normally, these codes are not available from the keyboard, but you can program any of them directly into one of the function keys. For example, if the character with code 255 is required (that is, the program refers to ch255) simply type:

> VDU 255

Then after the character appears enter:

> \*KEY 0 ' character here'

The 0 can be, of course, the key of your choice. (Once the character has appeared, use the edit keys to 'bringt down' into the quotes.)

## **OS 0.10**

All the programs in this book were written on OS 1.20. Most programs will work without modification, but programs with user-defined graphics will need their character codes changed. On OS 1.20, user-defined characters (before explosion) are defined in the ASCII code range 128 - 159 inclusive, but in OS 0.10 they are in the range 224 - 255. Therefore, the codes will need to have 96 added to them.

Also, you'll discover that using the function key in conjunction with SHIFT or CONTROL will not work, so the codes will have to be copied into the keys using the method described above, and the keys are unshifted. Some operating system calls are not supported in OS 0.10, so we've avoided using these as much as possible.

Three of the programs in this book (Mandala, Pontoon and Play Your Cards Right) will not fit on the computer if you have discs. To get them to fit, you need to disable the DFS. You do

this by entering:

```
*BUILD DWNLOAD
  1 *KEY 0 *TAPE|MFORA%=0TO(TOP-PAGE)ST
EP4:A%!&E00=A%!PAGE:NEXT:PAGE=&E00|MOLD|
MRUN|M
  2
Escape
```

The routine is now on your disc for you to use later on. Before you try to load the long program, enter \*EXEC DOWNLOAD and then load your program. Press f0 and your program will run as normal.

# Program Lister

This is the program which was used to list all the programs in this book. As we have explained, the program takes away most of the incompatibilities between the BBC Micro and the printer. On most printer/micro combinations, either the £ or the # is available (but not both) and often the wrong one is printed.

The program also deals with embedded character codes generated using the SHIFT/CTRL function keys. It converts the codes to list as [fs0] and the like for the sequence ' SHIFT-function 0' or [fc9] for ' CTRL- function 9' .Other codes handled by this routine, such as [ch255], refer to ' character code 255' .Multiple spaces are indicated with [spc5] to show five spaces. One or two spaces are simply shown as spaces within the string.

Another feature of Program Lister is its ability to restrict line widths and page lengths. This uses, in our case, an inbuilt function of the Epson FX80, but can be implemented with other printers by software in the program which counts lines. The program also allows a title and leading sentence to be entered. This is centred before the program is printed out.

## How to use it

The program must be in a \*SPOOL format. We find this a useful format in which to store backup copies. Once a spooled version has been made (as shown in the User Guide on page 402), run the Lister program. It will first ask whether you wish to enter a sentence which will be printed before the listing. If not, the next stage of the program is skipped. If you indicate you wish to enter a sentence, you do so, then press ' SHIFT-function 0' which will cause a new line to be printed. Any other SHIFT/CTRL function keys will act as if they are the space bar

when printed, but the line-centring routine will act as if it is a character. That is, the word will not be split if it falls at the end of a line. The other commands within the program are self-explanatory.

## **Epson FX80 Printer Codes**

These are generally sent as a series of ESCape codes (similar in operation to the BBC' s VDU statements).

LINE 940

PROC£ This procedure prints out the £ sign

LINE 1000

PROCINITIALISE This initialises the printer, as if it had been turned off then on again, thus clearing any previous settings

LINE 1380

VDU 2,1,27,1,ASC"@ " Initialises printer as above

VDU 1,27,2,ASC"Q",%,x% Defines printer width of x% characters

LINE 1390

VDU 1,27,1,ASC"I",1,49 Allows use of ' undefined'characters such as £

VDU 1,27,1,ASC"G" Puts printer in double-strike mode

VDU 1,27,1,ASC"N",1,y% Specifies the number of lines (y%) to be skipped at the bottom of the page

LINES 1510 - 1740

PROCENTER(S\$) This is a very useful routine which centres the string S\$, in the middle of screen of width w%. This procedure takes into account the SHIFT - function codes as previously explained.

```

10 REM *****
20 REM *** ***
30 REM *** PROGRAM LISTER ***
40 REM *** FOR ***
50 REM *** EPSON FX-80 ***
60 REM *** (C) I. HUTT 83 ***
70 REM *** ***
80 REM *****
90
100 VDU 3
110 @%=&90A:WIDTH 0
120 MODE 7
130 ON ERROR GOTO 1170
140 PRINT TAB(10,3);CHR$(141);"[fs3]Pr
ogram Lister[fs7]"
150 PRINT TAB(10,4);CHR$(141);"[fs4]Pr
ogram Lister[fs7]"
160 PRINT TAB(10,5);"[fs1]=====
===="
170 PROCINITIALISE
180 PRINT'"Do you wish to place a se
ntence at startof the listing (Y/N) ? ";
190 IF FNYN("N") THEN INPUT LINE'"[fs
1]Sentence :[fs7]"S$ ELSE S$=""
200 PRINT'"[fs6]Define page size[fs1]?
";
210 IF FNYN("N") THEN size=TRUE:PROC_s
ize ELSE size=FALSE
220 INPUT'"[fs1]Filename of program[fs
7]",F$
230 IF LEN(F$)<3 THEN PROCCAT(F$):GOTO
220
240 F%=OPENUP(F$)
250 IF F%=0 THEN CLOSE #0:PROCCAT(F$):
PRINT'" """;F$;" "" does not exist.":GOTO
220
260 IF size THEN 360

```



```

270 PRINT"[fs1]Line width A-80,B-40,C-
other (A,B,C) ";
280 REPEAT KEY$=GET$
290 UNTIL KEY$="A" OR KEY$="B" OR KE
Y$="C" OR KEY$=CHR$(13)
300 IF KEY$="A" OR KEY$=CHR$(13) THEN
w%=80
310 IF KEY$="B" THEN w%=40
320 IF KEY$="C" THEN INPUT'"Width 2-80
",w%:IF w%=0 THEN w%=80 ELSE IF w%<2 OR
w%>80 THEN 320
330 VDU 2,1,27,1,ASC"Q",1,w%,3
340 PRINT'"Line width ";w%;" character
s."
350
360 VDU 15,3
370 REM Strip off first line ie. ">LIS
T"
380
390 REPEAT C%=BGET #F%:UNTIL EOF #F% O
R C%=ASC">"
400 IF C%<>ASC">" THEN PRINT'"[fs1]Not
[fs8]*SPOOLed[fs9] Format.":GOTO 630
410 REPEAT UNTIL BGET#F%=13
420
430 REM start printer
440
450 PRINT''
460 VDU 2
470 PROCCENTER(S$)
480 R%=TRUE
490
500 REM list modified program from dis
c to printer
510
520 Q%=FALSE
530 REPEAT L%=BGET #F%
540 IF L%=62 AND R% THEN 570
550 IF L%=ASC"``" THEN PROCf ELSE

```

```

PRINT
  FNCHAR(L%);
  560   IF L%=13 THEN R%=TRUE ELSE R%=FA
LSE
  570   UNTIL L%=62 AND R%
  580
  590 REM close file and stop printer
  600
  610 IF size THEN VDU 1,12:PROCINITIALI
SE ELSE PRINT''''
  620 VDU 3
  630 CLOSE #F%
  640 END
  650
  660 REM this converts ASCII code from
program to a string that is printed
  670
  680 DEF FNCHAR(C%)
  690 IF C%=34 THEN Q%=- (Q%+1) :=" "":REM
detect quotes
  700 IF C%=32 AND Q% THEN PROCSPACES :R
EM detect multiple spaces
  710 IF C%=0 THEN =" "
  720 IF C%=13 THEN Q%=FALSE:=CHR$(13):R
EM line feed
  730 IF C%<128 THEN =CHR$(C%) :REM norm
al characters
  740 IF C%<138 THEN ="[fs"+STR$(C%-128)
+"]" :REM shft function codes
  750 IF C%>143 AND C%<154 THEN ="[fc"+S
TR$(C%-144)+"]" :REM ctrl function codes

  760 ="[ch"+STR$(C%)+"]" :REM other cod
es >127
  770
  780 REM deal with multiple spaces
  790
  800 DEF PROCSPACES
  810 IF NOT Q% THEN ENDPROC

```

```

820 LOCAL S%
830 S%=1
840 REPEAT C%=BGET#F%
850     IF C%=32 THEN S%=S%+1
860     UNTIL C%<>32
870 IF S%>2 THEN PRINT"[spc";S%;"]"; E
LSE PRINT SPC(S%);
880 IF C%=34 THEN Q%=- (Q%+1)
890 IF C%=ASC "`" THEN PROC`:C%=0
900 ENDPROC
910
920 REM deal with "£" sign
930
940 DEF PROC£
950 VDU 3,96,2,1,134
960 ENDPROC
970
980 REM initialise printer
990
1000 DEF PROCINITIALISE
1010 VDU2,1,27,1,ASC"@",1,27,1,ASC"6",3
1020 ENDPROC
1030
1040 REM y/n reply function
1050
1060 DEF FNYN(D$)
1070 LOCAL K$
1080 *FX 15
1090 REPEAT K$=GET$
1100     UNTIL K$="Y" OR K$="y" OR K$="N"
OR K$="n" OR K$=CHR$(13)
1110 IF K$=CHR$(13) THEN K$=D$
1120 PRINT ;K$
1130 IF K$="Y" OR K$="y" THEN =TRUE ELS
E =FALSE
1140
1150 REM error handling routine
1160
1170 CLOSE #0

```

```

1180 VDU3
1190 IF ERR<>17 THEN REPORT:PRINT" at 1
ine ";ERL:END
1200 PRINT'"[fs3]Do you wish to quit p
rogram (Y/N)[fs7]?";
1210 IF FNYN("Y") THEN PRINT'"[fs1]PRO
GRAM TERMINATED[fs7]":END
1220 PRINT'"Do you wish to re-initialis
e printer ?";
1230 IF FNYN("N") THEN 120 ELSE 180
1240 DEF PROC_size
1250 PRINT'"Default size 40*36."
1260 w%=40:l%=36
1270 PRINT'"Change size (Y/N) ?";
1280 IF NOT FNYN("N") THEN 1360
1290 INPUT'"Page width in characters 2-
80 ",w%
1300 IF w%=0 THEN w%=40
1310 IF w%<2 OR w%>80 THEN 1290
1320 INPUT'"Page length in lines 1-66 "
,l%
1330 IF l%=0 THEN l%=36
1340 IF l%<2 OR l%>66 THEN 1320
1350 IF l%=66 THEN PRINT"NO skip over p
erferation"
1360 x%=w%
1370 y%=66-l%
1380 VDU 2,1,27,1,ASC"@",1,27,1,ASC"Q",
1,x%
1390 VDU 1,27,1,ASC"I",1,49,1,27,1,ASC"
G",1,27,1,ASC"N",1,y%,3
1400 ENDPROC
1410 DEF PROCCAT(DR$)
1420 IF LEFT$(DR$,1)<>": " THEN 1490
1430 D%=VAL(MID$(DR$,2,1))
1440 IF DR%<1 OR DR%>3 THEN *.0
1450 IF DR%=1 THEN *.1
1460 IF DR%=2 THEN *.2
1470 IF DR%=3 THEN *.3

```

```

1480 ENDPROC
1490 * .
1500 ENDPROC
1510 DEF PROCCENTER(S$)
1520 IF S$="" THEN ENDPROC
1530 LOCAL C%,P%,C$,T$
1540 w%=w%-1
1550 FOR P%=1 TO LEN(S$)
1560     T$=""
1570     REPEAT C$=MID$(S$,P%,1)
1580         T$=T$+C$
1590         P%=P%+1
1600         UNTIL (P%-1)>=LEN(S$) OR LEN(T
$)>w% OR C$="[fs0]"
1610     P%=P%-1
1620     IF C$="[fs0]" THEN 1680
1630     IF P%>=LEN(S$) THEN 1680
1640     REPEAT P%=P%-1:T$=LEFT$(T$,LEN(T
$)-1):UNTIL RIGHT$(T$,1)="" OR LEN(T$)=
1
1650     IF RIGHT$(T$,1)="" THEN T$=LEFT
$(T$,LEN(T$)-1)
1660     IF LEN(T$)=1 THEN T$=MID$(S$,P%,
w%-1)+"-":P%=P%+w%-2
1670     FOR C%=1 TO LEN(T$)
1680         IF MID$(T$,C%,1)>="[fs0]" THEN
T$=LEFT$(T$,C%-1)+" "+MID$(T$,C%+1)
1690     NEXT
1700     PRINT TAB((w% DIV 2)-(LEN(T$) DI
V 2));T$
1710     NEXT
1720 PRINT
1730 w%=w%+1
1740 ENDPROC

```

# Procedure and Function Lister

This program, which is very handy when you're redebugging programs, lists all the procedure names and functions, in line order. To use the program, proceed as follows:

Load the program you are going to analyse.

Then enter, in direct mode, PRINT ~PAGE and write down the answer you get

Now enter PAGE = TOP + 256 and press RETURN.

Now enter NEW and press RETURN again.

Follow this by CHAINing this LISTER program. It will ask you to enter the PAGE value you wrote down earlier.

When you enter this value, prefix it with &.

When you press RETURN, the program will list out the procedures and functions, as in this example:

```
PAGE=&1C00
```

```
140  PROC_SHUFFEL
260  PROC_PLAY_GAME
560  PROC_INIT_GAME
630  PROC_PLACE_1st_HANDS
900  PROC_HUMAN_MOVE
950  PROC_SHOW_OUTCOME
1120 PROC_SETUP
1220 PROC_INTRO
1310 PROC_DRCD (X%, Y%, S%, C%)
1520 FNNUM (NUM)
1590 FNVALUE (card, ace)
1710 FNSUIT (card)
1760 PROC_BGET
1900 FN_COMMAND (BUY)
2000 PROC_TWIST
2190 FN_ANOTHER
```

```
2250 PROC_PL(CH$,S$)
```

All done!

>

Here' s the listing for this handy utility:

```
10 REM FUNCTION & PROCEDURE LISTER
20 MODE 7
30 INPUT'''PAGE="page$
40 page=EVAL(page$) AND &FF00
50 IF (!page AND &0000FFFF)=&FF0D THE
N PRINT'"No program!''':END
60 A%=page
70 REPEAT N%=(A%?1*256)+A%?2:L%=A%?3
80   FOR P%=4 TO L%
90     IF A%?P%=&DD THEN PROC_def
100    NEXT
110    A%=A%+L%
120    UNTIL ?A%=&0D AND A%?1=&FF
130 PRINT'"All done!"
140 END
150 DEF PROC_def
160 REPEAT P%=P%+1
170   UNTIL A%?P%=&A4 OR A%?P%=&F2 OR
P%>L%
180 IF P%>L% THEN ENDPROC
190 PRINT'TAB(5-LEN(STR$(N%)) );N%; " ";

200 IF A%?P%=&A4 THEN PRINT"FN"; ELSE
PRINT"PROC";
210 REPEAT P%=P%+1
220   VDU A%?P%
230   UNTIL P%=L% OR A%?(P%+1)=58 OR A
%?(P%+1)<33 OR A%?(P%+1)=61
```

