

General Utilities

Slow Down

This nifty program will slow down your BBC Micro so that you can see what the machine is actually doing when it carries out commands such as PLOT. You turn it on with *FX 14,4. You can slow it down even further with *FX 14,3. What your BBC Micro previously did in microseconds now takes seconds.

```
10 origin=&B00
15 delay=100
20 FOR P=0 TO 1
30 P%=origin
40 [OPT P*3
50 PHP:PHA
60 TXA:PHA
70 TYA:PHA
80 LDA #delay
90 .loopy LDX #255
100 .loopx DEX
110 BNE loopx
120 DEY
130 BNE loopy
140 PLA:TAY
150 PLA:TAX
160 PLA:PLP
170 RTS:]
180 NEXT
190 !&220=!&220 AND &FFFF0000 OR origi
n
```

Graphical VU-Meter

You should not attempt to use this program unless you are quite experienced at dealing with electrical circuits. The program produces two on-screen line charts. One represents the logarithmic value of the music output and the other shows the linear value.

```
10 REM Graphical VU-Meter
20
30 REM Connect (Ear/DIN) output of
40 REM Music Source to CH0 (ADVAL1)
50 REM input (pin 15) and Analogue
60 REM ground (pin 8 or 5) of
70 REM 15-way D connector on back
80 REM of computer (analogue in)
90
100 W%=0
110 S=1
120 MODE0
130 VDU 23,0,10,32,0;0;0;
140 *FX 16 1
150 *FX 190 8
160 A%=ADVAL1
170 Y1%=FNLN
180 Y2%=FNLI
190 M%=Y2%
200 REPEAT CLS
210     S%=TIME
220     X1%=0
230     REPEAT X%=TIME-S%
240         IF ADVAL1<>0 THEN A%=(3*ADVAL(
1) + 2*A%)/6
250         MOVE X1%*2,Y1%
```

```

260      Y1%=FNLN
270      DRAW X%*2,Y1%
280      N%=Y2%
290      Y2%=FNLI
300      MOVE X1%*2,M%
310      IF Y2%>M% THEN M%=Y2%
320      DRAW X%*2,M%
330      MOVE X1%*2,N%
340      DRAW X%*2,Y2%
350      X1%=X%
360      IF Y2%<1023 PRINT TAB(0,1);"LOG:>";STRING$(Y2%-512)DIV50,">";SPC(15-
POS)
370      IF Y2%>1023 THEN VDU 19,7,1,0,
0,0 ELSE VDU 19,7,7,0,0,0
380      UNTIL X%>=640
390      UNTIL FALSE
400 END
410 DEF FNLN=(LN(A%+100)-4.6)*150
420 DEF FNLI=(A%/9)+512

```

Poster

Now you can write messages to your loved one in letters so large you can read them from the other side of the street (if you live in a pretty narrow street).

Follow the prompts to set the character size, then enter your message which can be up to 255 characters long.

The output of this program is very impressive. This program can even cope with user-defined graphics.

```
10 REM POSTER
20
30 MODE 3
40 ON ERROR IF ERR=17 THEN 310 ELSE V
DU3:REPORT:PRINT" at line ";ERL:END
50 DIM PARBLK 8
60 PRINT TAB(37,2);"POSTER"
70 INPUT'"Width of paper in characters (8 to 256) : "W%
80 IF W%<8 OR W%>256 THEN PRINT'"Please enter size in the range 8 to 256 inclusive."::GOTO 70
90 M%=W% DIV 8
100 INPUT'"Width of character Pixel : "X%
110 PRINT"Length of character Pixel (1 to ";M%;"")";
120 INPUT " : "Y%
130 IF Y%<1 OR Y%>M% THEN PRINT'"Please enter size in the range 1 to ";M%;;GOTO 120
140 INPUT LINE'"Sentence : "S$
150 VDU 2
160 FOR C%=LEN(S$) TO 1 STEP -1
```

```

170 C$=MID$(S$,C%,1)
180 PROC_SHAPE(C$)
190 FOR X=1 TO 8
200 FOR PX=1 TO X%
210 FOR Y=1 TO 8
220 P%=FN_PIXEL(X,Y)
230 IF ASC(C$)<127 THEN O$=C$ ELSE O$+
" * "
240 IF P% THEN PRINT STRING$(Y%,O$); E
LSE PRINT STRING$(Y%, " ");
250 NEXT Y
260 PRINT
270 NEXT PX
280 NEXT X
290 NEXT C%
300 VDU 1,12
310 VDU 3
320 END
330 DEF FN_PIXEL(C,R)
340 IF C<1 OR C>8 THEN =FALSE
350 IF R<1 OR R>8 THEN =FALSE
360 LOCAL C%
370 C%=PARBLK?R
380 IF (C% AND 2^(C-1))=0 THEN =FALSE
ELSE =TRUE
390 DEF PROC_SHAPE(C$)
400 LOCAL A%,X%,Y%
410 ?PARBLK=ASC(C$)
420 X%=PARBLK MOD 256:Y%=PARBLK DIV 25
6
430 A%=&0A:CALL &FFF1
440 ENDPROC

```

Orbit Predictor

This clever program, based on one written by D. A. Haines G4IPZ, calculates - from data obtained from the University of Surrey satellite control (UOSAT) - twenty respective orbits, their equatorial crossing times, and the angular placement west of the zero meridian. The primary data for each day can be obtained from the UOSAT Command Centre answerphone on 0483 61202. (This number is worth ringing for the message, which is always interesting, even if you don't intend running the program.)

The AOS times for the British Isles are about ten minutes after the equator crossing. The UOSAT downlink is on 145.825 megahertz and can be received - when the satellite link is operating - on most amateur NBFH two-metre equipment. The satellite is fitted with many interesting features, including synthesized speed speech for reports on status and the like. The satellite was malfunctioning while this book was being written ('so we are effectively deaf and blind', the plaintive Command Centre recorded message reported when we rang it), but the data produced by the program is still valid.

Note how the user inputs are accepted. They can be edited using the cursor keys if mistakes are made, by moving left and right and overtyping. Quite a neat and useful feature!

Here' sa sample run, which shows that equator crossings between 320 degrees west and 8 degrees west would afford the best coverage of the UK:

=====

TIME 14:08 DEG W 342.00

Time	DEG W
15:43	5.87
17:18	29.74
18:54	53.61
20:29	77.47
22:04	101.35
23:40	125.22
01:15	149.08
02:50	172.96
04:26	196.83
06:01	220.70
07:36	244.57
09:12	268.43
10:47	292.30
12:23	316.17
13:58	340.05
15:33	3.91
17:09	27.78
18:44	51.65
20:19	75.52
21:55	99.39

And here' sthe program listing. Notice that it allows you to have the data just on the screen or to produce (line 170) a hard copy of the information on your printer.

```

10 REM ORBIT PREDICTOR
20
30 ON ERROR GOTO 1250
40 MODE 7
50 @%=&02020A
60 PRINT'''TAB(10);"[fs3]Orbit Predic
tor[fs7]"
70 PRINT TAB(9);"[fs3]=====
==[fs7]"
80 PRINT'''[fs2]UOSAT Command Centre
(UK)[fs7]"
90 PRINT'"[fs3](Answerphone) 0483 612
02[fs7]"
100 PRINT'"[fs3]UOSAT downlink 145.82
5 Mhz[fs7]"
110 PRINT'''[fs1]TIME [fs3]HH.MM[fs1]:
[fs2]00.00";STRING$(5,CHR$(8));
120 time$=FN_INTIME
130 hour=VAL(LEFT$(time$,2))
140 min=VAL(RIGHT$(time$,2))
150 PRINT'''[fs1]Equator crossing :[fs
3]DEG W[fs2]000.00";STRING$(6,CHR$(8));
160 deg=FN_INDEG
170 PRINT'''[fs3]Do you require a hard
copy (Y/N) :[fs2]No ";
180 HC=FALSE
190 PRINT STRING$(3,CHR$(8));
200 REP$=FNKEY("YyNn")
210 IF DELT THEN 190
220 IF RET THEN 270
230 IF REP$="Y" OR REP$="y" THEN PRINT
"Yes";:HC=TRUE ELSE PRINT"No ";:HC=FALSE
240 REPEAT KEY=GET

```

```

250 UNTIL KEY=13 OR KEY=127
260 IF KEY=127 THEN 190
270 A=23.87
280 hour=(60*hour)+min
290 B=95.358
300 CLS
310 IF HC THEN VDU 15,2,1,10,1,10 ELSE
VDU 14
320 PRINT STRING$(27,"=") '
330 PRINT " TIME ";LEFT$(time$,2);":";R
IGHT$(time$,2);TAB(14);"DEG W ";RIGHT$( "
[spc3]" +STR$(INT(deg)),3);".";RIGHT$("00
"+STR$(INT(0.5+((deg-INT(deg))*100)),2)

340 PRINT 'TAB(4);"Time";TAB(14);"DEG
W" '
350 FOR I=1 TO 20
360 hour=hour+B
370 IF hour>1440 THEN hour=hour-1440

380 Z=hour/60
390 X=INT(Z)
400 C=INT((Z-X)*60)
410 deg=deg+A
420 IF deg>360 THEN deg=deg-360
430 Q=(INT(100*deg))/100
440 PRINT"[spc4]";RIGHT$("00"+STR$(X
),2);":";RIGHT$("00"+STR$(C),2),Q
450 NEXT
460 PRINT'STRING$(27,"-") '
470 IF HC THEN VDU 1,10,1,10,1,10,1,10
,3 ELSE VDU 15
480 *FX 15 1
490 END
500 DEF FN_INTIME
510 LOCAL time$
520 time$="0000"
530 time$=FN_INSRT(time$,FNKEY("012"),
1)

```

```

540 IF DELT THEN 530
550 IF RET THEN =time$
560 PRINT LEFT$(KEY$,1);
570 IF time$<>"2" THEN time$=FN_INSRT(
time$,FNKEY("0123456789"),2) ELSE time$=
FN_INSRT(time$,FNKEY("0123"),2)
580 IF DELT THEN VDU 8:GOTO 530
590 IF RET THEN =time$
600 PRINT KEY$;". ";
610 time$=FN_INSRT(time$,FNKEY("012345
"),3)
620 IF DELT THEN VDU 8,8:GOTO 570
630 IF RET THEN =time$
640 PRINT KEY$;
650 time$=FN_INSRT(time$,FNKEY("012345
6789"),4)
660 IF DELT THEN VDU 8:GOTO 610
670 IF RET THEN =time$
680 PRINT KEY$;
690 *FX 4 1
700 REPEAT KEY=GET
710 UNTIL KEY=13 OR KEY=136
720 IF KEY=136 THEN VDU 8:GOTO 650
730 *FX 4
740 =time$
750 DEF FN_INDEG
760 deg$="000.00"
770 deg$=FN_INSRT(deg$,FNKEY("0123"),1
)
780 IF DELT THEN 770
790 IF RET THEN =VAL(deg$)
800 PRINT KEY$;
810 IF deg$="3" THEN deg$=FN_INSRT(deg
$,FNKEY("012345"),2) ELSE deg$=FN_INSRT(
deg$,FNKEY("0123456789"),2)
820 IF DELT THEN VDU 8:GOTO 770
830 IF RET THEN =VAL(deg$)
840 PRINT KEY$;
850 deg$=FN_INSRT(deg$,FNKEY("01234567

```

```

89"), 3)
    860 IF DELT THEN VDU 8:GOTO 810
    870 IF RET THEN =VAL(deg$)
    880 PRINT KEY$;
    890 VDU 9
    900 deg$=FN_INSRT(deg$,FNKEY("01234567
89"), 5)
    910 IF DELT THEN VDU 8,8:GOTO 850
    920 IF RET THEN =VAL(deg$)
    930 PRINT KEY$;
    940 deg$=FN_INSRT(deg$,FNKEY("01234567
89"), 6)
    950 IF DELT THEN VDU 8:GOTO 900
    960 IF RET THEN =VAL(deg$)
    970 PRINT KEY$;
    980 *FX 4 1
    990 REPEAT KEY=GET
1000     UNTIL KEY=13 OR KEY=136
1010 IF KEY=136 THEN VDU 8:GOTO 940
1020 *FX 4
1030 =VAL(deg$)
1040 END
1050 DEF FNKEY(ACC$)
1060 *FX 4 1
1070 REPEAT *FX 15
1080     FLAG=FALSE
1090     DELT=FALSE
1100     RET=FALSE
1110     SKIP=FALSE
1120     KEY$=GET$
1130     FOR LOOP=1 TO LEN(ACC$)
1140         IF KEY$=MID$(ACC$,LOOP,1) THEN
FLAG=TRUE
1150     NEXT
1160     IF KEY$=CHR$(136) THEN DELT=TRUE
:KEY$="":FLAG=TRUE
1170     IF KEY$=CHR$(13) THEN RET=TRUE:K
EY$="":FLAG=TRUE
1180     IF KEY$=CHR$(137) THEN SKIP=TRUE

```

```

:VDU 9:KEY$="":FLAG=TRUE
1190     UNTIL FLAG
1200 *FX 4
1210 =KEY$
1220 DEF FN_INSERT(varin$,addto$,posn)
1230 IF addTo$="" THEN =varin$
1240 =LEFT$(varin$,posn-1)+addTo$+MID$(
varin$,posn+1)
1250 *FX 4
1260 @%=10
1270 VDU 3,12,10,10
1280 IF ERR<>17 THEN REPORT:PRINT" at l
ine ";ERL

```