

SOFTWARE FILE

(continued from previous page)

```

THEN LET F$(10)="1"
9150 IF F$(9)="1" AND F$(8)="2"
THEN LET F$(10)="2"
9155 IF F$(8)="2" AND F$(7)="2"
THEN LET F$(10)="3"
9160 IF F$(7)="2" AND F$(6)="2"
THEN LET F$(10)="4"
9170 LET U=3
9180 IF F$(7) <> "0" THEN LET U=U+
1
9190 IF F$(6) <> "0" THEN LET U=U+
1
9200 IF F$(5) <> "0" THEN LET U=U+
1
9210 IF F$(4) <> "0" THEN LET U=U+
1
9215 IF U=5 THEN LET U=5
9220 LET F$(5)=CHR$(U+65)
9230 LET A$(X,Y,Z)=F$
9240 NEXT Z
9250 NEXT Y
9260 NEXT X
9270 GOTO 15
9500 DIM A$(4,4,4,10)
    
```

```

9503 LET COUNT=0
9505 LET H$="0"
9507 LET J=0
9509 LET I=0
9510 DIM F$(10)
9515 DIM O$(4)
9520 LET U1=35
9530 LET U2=35
9540 LET U3=35
9550 LET U4=30
9560 LET U5=15
9570 LET U6=0
9580 LET U7=15
9590 LET U8=2
9595 LET UPLAY=103
9615 LET INSERT=360
9616 LET DRAW=2060
9617 LET FOUND30=2100
9618 LET FOUND3X=2430
9619 LET PRINTWIN=3000
9620 LET SET=6000
9630 LET SEARCH=6200
9640 LET WEIGHT=6400
9650 LET UPDATE=6600
9660 LET UPDATEX=6820
9670 LET UPDATED=6710
9680 LET IPLAY=7000
    
```

```

9690 LET UWIN=7400
9710 LET POINTX=7480
9720 LET POINT0=7520
9725 LET MARKWIN=7560
9730 LET MARK=7600
9740 LET WINLINE=7700
9750 LET SETUP=9000
9800 GOTO SETUP
9900 SAVE "3D43"
9910 RUN
    
```

The Basic routine.

```

7020 LET HIWT=0
7030 FOR X=1 TO 4
7040 FOR Y=1 TO 4
7050 FOR Z=1 TO 4
7060 IF A$(X,Y,Z,4) <> "1" THEN GO
TO 7200
7070 LET WT=CODE A$(X,Y,Z,5)
7080 IF WT < HIWT THEN GOTO 7200
7090 IF WT=HIWT AND RND>.33 THEN
GOTO 7200
7100 LET HIWT=WT
7110 LET C=X
7120 LET D=Y
7130 LET E=Z
7200 NEXT Z
7210 NEXT Y
7220 NEXT X
    
```

Keyword define

J Bradshaw,
Runcorn,
Cheshire.

VIC-20

THIS PROGRAM runs on the unexpanded Vic-20 and allows you to define eight Basic keywords to the function keys. The Basic program sets up a small machine-code program — 139 bytes — at the top of memory. Therefore, you can run or type fairly long programs while this one is running. There is also an error check in line 40 which ensures

you have typed in the data correctly, thereby preventing system crashes.

Four permanent functions are defined:

F1 = Screen colour normal
F = Screen colour black
F5 = Quote mode on
F7 = Quote mode off

And, of course, by using the function keys in conjunction with the Shift and Commodore keys, eight Basic keywords can be printed at the current cursor position on the screen. You can choose any Basic keyword and assign it to its appropriate function key by altering the eight Data values on line 200 respectively.

Some values may give an output of, say, half

a keyword for example,

INT (PRINT)

yet by trial and error you should find the desired keyword. These values on line 200 merely state how far into the ROM keyboard table the Vic is to look before printing out the characters it finds. At present, the function keys are defined as follows:

Shift F1 = PRINT CBM F1=FOR
Shift F3 = POKE CBM F3=NEXT
Shift F5 = PEEK CBM F5=GOTO
Shift F7 = THEN CBM F7=GOSUB

To disable the program press run stop and restore. To enable enter Sys 7541.

```

1 REM DEFINED FUNCTION      KEYS
2 REM BY JOHN BRADSHAW"
10 POKE51,117:POKE52,29:POKE55,117:POKE56,29
20 READA:IFA=-1THENFORC=7672T07679:READA:POKEC,A:NEXT:GOTO40
30 T=T+A:POKE7541+I,A:I=I+1:GOTO10
40 IFT<>17900THENPRINT"NO DATA WRONG":END
50 SYS7541:PRINT"FUNCTION KEYS DEFINED"
60 DATA120,169,130,141,20,3,169,29,141,21,3,88,96,165,197,197,187,240,23,133,18
7,162,1
70 DATA201,39,240,18,232,201,47,240,13,232,201,55,240,8,232,201,63,240,3,76,191
,234,173
80 DATA141,2,201,1,240,50,201,2,240,42,224,1,240,22,224,2,240,26,224,3,240,7,16
9,0,133
90 DATA212,76,191,234,169,1,133,212,76,191,234,169,27,141,15,144,76,191,234,169
,8,141
100 DATA15,144,76,191,234,232,232,232,232,189,247,29,133,188,164,188,185,161,1
92,48,8
110 DATA32,210,255,230,188,76,224,29,56,233,128,32,210,255,76,191,234,-1
200 DATA99,89,214,148,0,3,32,48
    
```

Proc point

John Chalmers,
Godstone,
Surrey.

LYNX

LYNX BASIC provides no instructions for examining the display, and Peeking it is not possible because of the way it is organised. The routine given in the listing provides the Basic programmer with a method of examining any point on the display. It works by calling a ROM routine which looks at the display RAM. The routine is written as a procedure, so to use it you simply type:

```
PROC POINT(X,Y)
```

where X and Y are the co-ordinates of the point you wish to examine. The result will be contained in the variable z on exit from the routine and will be the colour value of the point X,Y. Note how small letter variables are

used within the procedure. This would seem like good practice whenever using procedures as extra "commands" — the small letter names are reserved solely for such routines and thus do not corrupt the capital letter names of the main variables. To test the procedure once you have entered it, enter the following lines

```
INK 4
DOT100,100
PROCPPOINT(100,100)
?
```

The z should of course be 4.

```

99990 DEFPROC POINT(x,y)
99991 LET y=((x+256*y)/8),x=INT(y),
y=2*(7-(FRAC(y)*8)),z=0
99992 CALL @0069,@8000+x
99993 IF (HL BNAND y)>0 THEN LET z=z+1
99994 CALL @0069,@C000+x
99995 IF (HL BNAND y)>0 THEN LET z=z+2
99996 CALL @0070,@C000+x
99997 IF (HL BNAND y)>0 THEN LET z=z+4
99998 ENDPROC
    
```

Key click

P A Street,
Biggleswade,
Bedfordshire.

DRAGON

THIS KEYBOARD click routine for the Dragon 32 will emit a short click whenever a key is depressed.

The machine-code routine generates a short click, by setting up the sound generator and loading the memory location 65312 with a number, between 0 and 255.

Memory locations 363 and 364 contain the address to which the computer jumps when a key is pressed. This address is changed to the beginning of the click routine. The routine is then started by Poking location 362 with the number 126.

(continued on page 175)