HI-RES SCAN

Scan a shape for your shape table right from the screen

A number of good programs have been published to simplify the creation of Hi-Res shapes and shape tables. After several unsuccessful attempts to duplicate an existing shape using my favorite of these programs, I wrote Hi-Res Scan to supplement it. You'll find this program useful when you have a shape that is not in a shape table and you want an easy way to put it in one, or you have a particular shape in mind that proves easier to generate using standard HPLOT commands. It will not replace your favorite shape creating program, but it can be a very useful supplement to it.

USING THE PROGRAM

When you have a shape or a portion of a Hi-Res picture on Hi-Res screen 1, RUN HIRES.SCAN. After the program gets ready, your shape will reappear and you will be prompted to either enter the name of a shape table which you want to add the shape to or to pick the number of shapes you want to allow in a new shape table. (If you wish to add the new shape to an existing shape table, keep in mind that there must be space in the shape table's directory for additional shapes.) When entering a filename, you may press ? to view a disk catalog and to change the slot and drive.

You'll then be prompted to create a boundary around your shape. You can use the Arrow keys and the I-J-K-M keys. When the boundary is drawn, press P. At this point, the scanning begins, starting at the top left corner of the shape. After the shape has been scanned, you can choose to look at all the shapes in the table, save the shape table, including your new shape, or save the new shape table.

ENTERING THE PROGRAM

Type in the Applesoft program from **Listing 1**. Save it to disk with the command:

SAVE HIRES.SCAN

For help with entering *Nibble* listings, see the Typing Tips section.

HOW IT WORKS

The program begins by setting LOMEM: to protect the Hi-Res screen; the shape table is stored on Hi-Res page 2. Then lines 570-810 display the shape and prompt the user to enter the name of the shape table. An ampersand routine (POKEd into place by lines 1750 to 1800), which duplicates in a limited way the LORES SCRN function, is used in lines 1000 to 1250 to shrink the boundary to the smallest size that will encompass the shape. The ampersand function does not return a color (as SCRN does), it simply returns a value of 1 for C if the spot (X,Y, HPLOT coordinate as in & C = X,Y) is on and a 0 if it is off. To avoid the problem of too many "move up — no plot" vectors during scanning, this program starts at the top left corner of the shape boundary and scans right and then down and then left and then down, using the ampersand routine to determine whether to plot. In essence these lines create up to 1000 plot (or no plot) vectors. These vectors are then converted to shape table entries by the subroutine at line 300 and POKEd into the shape table by lines 1410 to 1470.

MODIFICATIONS

Like most utilities, Hi-Res Scan has some limitations. First, before running it, you must get the desired shape onto Hi-Res screen 1. Significant modifications would be required to scan a shape from Hi-Res screen 2. Second, it uses the standard shape directory/shape table as described in various Apple manuals. If your favorite shape creating/shape table program uses something else, you will have to modify accordingly. Third, as written, Hi-Res Scan is limited to a rectangular shape size of less than 1000 plot coordinates. As a reference, capital letters on the text screen are generally 5 wide by 7 high (or 35 plot coordinates). A shape could be five times that size in each direction. If you try to scan anything larger, speed will become a factor.

LISTING 1: HIRES.SCAN

```
37   10 REM *********************
C0   20 REM * HIRES.SCAN        *
89   30 REM * BY RON SJOLANDER  *
AE   40 REM * COPYRIGHT(C) 1988 *
4A   50 REM * CONCORD, MA 01742 *
80   60 REM *********************
45   70 REM
26   80 TEXT : HOME : PRINT CHR$ (21): SCALE= 1:
        ROT= 0
6F   90 POKE 49232,0: POKE 49235,0: POKE 49236,0:
        POKE 49239,0: REM HGR W/O CLEARING SCREEN
FC  100 LOMEM: 24576: REM ABOVE PAGE 2
29  110 MEM = 16384: REM STORE SHAPE TABLE ON PAGE 2
D7  120 ONERR GOTO 1840
00  130 PRODOS = PEEK (48896) = 76: IF NOT PRODOS
        THEN GOTO 150
5C  140 LF$ = CHR$ (13) + CHR$ (10): CHR$ (21): DN$ =
        CHR$ (10):UP$ = CHR$ (11)
DE  150 DIM W(1000),VEC(100)
AE  160 POKE 232,0: POKE 233,64
DB  170 D$ = D$ + A$
18  180 GOSUB 1750: REM HIRES SCRN FUNCTION
A4  190 GOTO 570
02  200 VTAB 21: HTAB 1: CALL - 958: INVERSE :
        HTAB 13: PRINT "PLEASE STANDBY": NORMAL :
        HTAB 1
A4  210 REM MOVE CURSOR
     220 IF X$ = "J" OR X$ = "j" OR X$ = LF$ THEN
        H = X - 5: IF H < 0 THEN H = X: RETURN
3B  230 IF X$ = "K" OR X$ = "k" OR X$ = RT$ THEN
        H = X + 5: IF H > 279 THEN H = X: RETURN
01  240 IF X$ = "I" OR X$ = "i" OR X$ = UP$ THEN
        V = Y - 5: IF V < 0 THEN V = Y: RETURN
     250 IF X$ = "M" OR X$ = "m" OR X$ = DN$ THEN
        V = Y + 5: IF V > 159 THEN V = Y: RETURN
17  260 XDRAW 1 AT X,Y
     270 XDRAW 1 AT H,V
DC  280 X = H: Y = V: RETURN
     290 REM CREATE SHAPE TABLE ENTRIES
     300 B = 0: Q = 0: FOR V = 1 TO P
     310 IF B = 2 AND (M(V) > 0 AND M(V) < 4 THEN 34
        0
8F  320 IF B = 2 AND (M(V) = 0 OR M(V) > 4) THEN 330
     330 B = 0: Q = Q + 1
59  340 VEC(Q) = VEC(Q) + M(V) * (8 ^ B)
     350 B = B + 1
     360 IF B = 3 THEN B = 0: Q = Q + 1
     370 VEC(Q) = VEC(Q): HTAB 34: PRINT : NEXT V
     380 RETURN
62  390 REM SCAN RIGHT
     400 FOR X = X1 TO X2 - 1
9E  410 A = C = X1
     420 P = P + 1:M(P) = 1: IF C = 1 THEN M(P) = 5
     430 NEXT
     440 X = X2,Y1
     450 P = P + 1:M(P) = 1: IF C = 1 THEN M(P) = 5
     460 IF Y1 = Y2 THEN E = 0
     470 RETURN
     480 REM SCAN LEFT
91  490 FOR I = X2 TO X1 STEP - 1
CD  500 P = I, Y1
C7  510 P = P + 1:M(P) = 3: IF C = 1 THEN M(P) = 7
81  520 NEXT I
D5  530 X1 = X1, Y1
     540 E = 1
     550 Y1 = Y1 + 1: IF Y1 > Y2 THEN E = 0
     560 RETURN
     570   VTAB 21: HTAB 1: CALL - 958: PRIN
        T "DO YOU WANT TO ADD THIS SHAPE TO AN":
        PRINT "EXISTING TABLE? (Y/N) ";: POKE
        6368,0: GET X$
CD  580 IF X$ = "N" OR X$ = "n" THEN 570
E6  590 IF X$ < > "Y" AND X$ < > "y" THEN 580
95  600 VTAB 21: HTAB 1: CALL - 958: PRINT "ENTER
        SHAPE TABLE FILENAME (OR PRESS '?'
        FOR CATALOG): INPUT ">;F$: IF F$ = "?"
        THEN 1920
F3  610 PRINT D$;"BLOAD ";F$;",A";MEM
FA  620 LE = PEEK (43616 + 5224 * PRODOS) + PEEK
        (43617 + 5224 * PRODOS) * 256
82  630 IF PEEK (MEM - 2) + PEEK (MEM - 3) * 256
        < PEEK (MEM) + 1: 2 THEN 790
F1  640 VTAB 21: CALL - 958: PRINT "SORRY - THERE
        NOT ENOUGH ROOM IN YOUR"
     650 PRINT "TABLE DIRECTORY FOR ANOTHER SHAPE."
     660 PRINT "TRY ANOTHER SHAPE TABLE? (Y/N) ";
16  670 GET X$
     680 IF X$ = "Y" OR X$ = "y" THEN 570
     690 IF X$ < > "N" AND X$ < > "n" THEN 670
     700 GOTO 1820
     710 REM NEW SHAPE TABLE
     720 F$ = ""
74  730 VTAB 21: HTAB 1: CALL - 958: PRINT "HOW M
        ANY SHAPES WOULD YOU LIKE TO"
     740 PRINT "LEAVE SPACE FOR IN THE SHAPE DIRECT
        ORY?"
D5  750 INPUT N$: N$ = VAL (N$): IF N$ < 1 OR N$
        > 255 THEN PRINT CHR$ (7); : GOTO 730
     760 X = N$ - 2 + 1:HB = INT (X / 256): LX = X -
        HB * 256
     770 POKE MEM + 1,0: POKE MEM + 2,L
        B: POKE MEM + 3,HB
EF  780 LOC = MEM + 2 - N$ - 2: GOTO 830
33  790 N$ = PEEK (MEM + 1): MEM = MEM + 1
     800 ST = LOC: MEM: MEM = PEEK (MEM) + 1: LOC =
        MEM + 1
     810 POKE MEM (MEM) - 2,LB: POKE MEM +
        1,HB * 256
82  820 REM OUTLINE SHAPE
6D  830 VTAB 21: CALL - 958: PRINT "POSITION THE
        CURSOR"
     840 VTAB 21: INVERSE : PRINT "PRESS 'P' WHEN
        I N POSITION.": NORMAL
04  850 HCOLOR= 3: X = 4:Y = 144: POKE 232,39:
        POKE 233,32: XDRAW 1 AT X,Y
     860 FOR I = 1 TO 3
     870 ON I GOSUB 930,940,950
DB  880 GET X$: VTAB 21: IF X$ = "P" OR X$ = "p"
        THEN H(I) = H:V(I) = V: GOTO 900
     890 GOSUB 220: GOTO 880
     900 NEXT : HCOLOR= 3: XDRAW 1 AT H,V:
        POKE 232,0: POKE 233,64
74  910 HPLOT H(1),V(1) TO H(2),V(1) TO H(2),V(3)
        TO H(1),V(3) TO H(1),V(1)
     920 GOTO 1010
B7  930 VTAB 22: HTAB 1: CALL - 868: PRINT "SLIGH
        TLY ABOVE AND LEFT OF THE SHAPE.": RETURN
BC  940 VTAB 22: HTAB 1: CALL - 868: HTA
        B 1: PRINT
     950 PRINT "MOVE THE CURSOR STRAIGHT ACROSS"
     960 PRINT "TO THE RIGHT OF THE SHAPE.": RETURN
A3  970 VTAB 22: HTAB 1: CALL - 868: HTA
        B 1: PRINT
     980 PRINT "MOVE THE CURSOR STRAIGHT DOWN"
     990 PRINT "TO JUST BELOW THE SHAPE.": RETURN
     1000 REM SHRINK OUTLINE
A8  1010 GOSUB 240: VTAB 23: HTAB 2: PRINT "SHRINK
        ING OUTLINE TO A MINIMUM SIZE."
     1020 FOR I = H(1) + 1 TO V(3) - 1
     1030 FOR II = H(1) + 1 TO H(2) - 1
DA  1040 C = II,I
     1050 IF C = 1 THEN 1070
     1060 NEXT II,I
     1070 X = I
AE  1080 FOR I = H(1) + 1 TO H(2) - 1
A6  1090 FOR II = Y1 TO V(3) - 1
     1100 C = I,II
     1110 IF C = 1 THEN 1130
     1120 NEXT II,I
EE  1130 X1 = I
     1140 FOR I = V(3) - 1 TO Y1 STEP - 1
     1150 FOR II = X1 TO H(2) - 1
     1160 C = I,II
     1170 IF C = 1 THEN 1190
     1180 NEXT II,I
EE  1190 Y2 = I
     1200 FOR I = H(2) - 1 TO X1 STEP - 1
68  1210 FOR II = Y1 TO Y2
     1220 C = I,II
```

LISTING 1: HIRES.SCAN

```
ED    1230  IF C = 1 THEN 1250
14    1240  NEXT II,I
43    1250  R2 = 1
86    1260  HCOLOR= 6 : HPLOT H(1),V(1) TO H(2),V(2) T
            O H(2),V(3) TO H(1),V(3) TO H(1),V(1)
02    1270  HCOLOR= 3 : HPLOT X1 - 3,Y1 - 3 TO X2 - 3,
            Y1 - 3 TO X2 + 3 TO X1 - 3,Y2 + 3 TO X1 -
            3,Y2 + 3 TO X1 - 3,Y1 - 3
78    1280  REM  SCAN SHAPE
B3    1290  GOSUB 200 : HTAB 1
6A    1300  VTAB 21 : PRINT "SCANNING RIGHT - ";Y1
B3    1310  GOSUB 400 : REM  SCAN RIGHT
E1    1320  IF E = 1 THEN 1380
C6    1330  VTAB 21 : HTAB 1 : CALL - 868 : PRINT "SCAN
            NING LEFT - ";Y1
C6    1340  GOSUB 490 : REM  SCAN LEFT
51    1350  IF E = 0 THEN 1280
02    1360  GOTO 1300
4D    1370  REM  CREATE SHAPE TABLE ENTRIES AND POKE
            INTO TABLE
59    1380  HOME : GOSUB 200 : VTAB 23 : HTAB 3 : PRINT "CREATI
            NG SHAPE TABLE ENTRIES -"
37    1390  VTAB 21 : HTAB 33 : PRINT "VECTOR";
80    1400  GOSUB 3000
81    1410  GOSUB 400 : REM  SCAN 4 : POKING
            SHAPE ENTRIES INTO TABLE.
F0    1420  FOR I = 0 TO Q
67    1430  POKE LOC,VEC(I)
43    1440  LOC = LOC + 1
4A    1450  NEXT
A9    1460  POKE LOC,0
1C    1470  LOC = LOC + 1 : POKE LOC,0
EE    1480  HOME : VTAB 21 : HTAB 3 : PRINT "WOULD YOU
            LIKE TO SEE THE SHAPES IN"; : PRINT "THIS SHA
            PE TABLE? (Y OR N) "; : GET X$
A5    1490  IF X$ = "N" AND X$ < > "y" THEN 1610
0D    1500  IF X$ < > "N" AND X$ < > "y" THEN 1480
51    1510  HCOLOR= 3 : SCALE = 1 : ROT= 0
25    1520  FOR I = 1 TO 5
1A    1530  HGR : DRAW I AT 100,100
1E    1540  HOME : VTAB 21 : HTAB 3
53    1550  VTAB 21 : HTAB 1 : CALL - 958 : PRINT "PRES
            S RETURN TO CONTINUE "; : POKE - 16368,0
8F    1560  GET Z$
B0    1570  NEXT I
B0    1570  HOME : VTAB 21 : HTAB 3 : PRINT "SEE SHAPES
            AGAIN? (Y OR N) "; : GET X$
5B    1580  IF X$ = "Y" OR X$ = "y" THEN 1520
A7    1590  IF X$ < > "N" AND X$ < > "n" THEN 1570
50    1600  HOME : VTAB 21 : HTAB 3
A1    1610  IF "5" = "" THEN 1600
84    1620  REM  DO YOU WANT TO ADD THIS SHAPE?
14    1630  PRINT "DO YOU WANT TO ADD THIS SHAPE"; : PRINT TO F5:(Y/N) ";:POKE - 16368,
            0. GET X$
5A    1640  IF X$ = "Y" OR X$ = "y" THEN 1710
2D    1650  IF X$ < > "N" AND X$ < > "n" THEN 1600
5B    1660  HOME : VTAB 21 : HTAB 3 : PRINT "WANT TO SAVE THIS
            SHAPE TABLE? (Y OR N)
6C    1670  GET X$ : IF X$ = "N" OR X$ = "n" THEN 1820
D2    1680  IF X$ = "Y" AND X$ < > "y" THEN 1510
A8    1690  VTAB 21 : HTAB 1 : CALL - 958
70    1700  PRINT "ENTER A NAME FOR THE TABLE": PRINT
            "OR PRESS RETURN TO CANCEL"; : PRINT F5,
            IF F5 < > "" THEN LE - LEM : PRINT D
            5 "BSAVE ";F5;" A";MEM;",L";LE
4E    1710  ON "5" = "" GOTO 1820:LE = LOC - MEM
3B    1720  PRINT D5,"BSAVE ";F5;",A";MEM;",L",LE
A4    1730  GOTO 1820
1D    1740  REM  POKE IN M/L HIRES "SCRN" FUNCTION FO
            LLOWED BY SHIFT TABLE.
F4    1750  FOR I = 768 TO 812: READ J : POKE I,J: NEX
            T I
C6    1760  DATA  32,227,223,133,133,132,134,169,208,
            32
A6    1770  DATA  192,222,165,18,72,165,17,72,32,185
DA    1780  DATA  246,32,17,244,165,48,49,38,240,2
2C    1790  DATA  169,1,168,32,1,227,76,91,218,1,0,4,
            0,57,8
D8    1800  POKE 1013,76 : POKE 1014,0 : POKE 1015,3
F1    1810  RETURN
82    1820  TEXT : HOME : VTAB 12 : PRINT "PRESS RETUR
            N TO CONTINUE" : PRINT "OR ESCAPE TO QUIT";
```

```
                POKE - 16368,0. GET Z$ : IF Z$ = CHR$ (27
                ) THEN  HOME : VTAB 23: END
B7     1830  RUN
       1840  VTAB 21 : HTAB1 : CALL - 958:EN = PEEK (
RA             222):EL = PEEK (218) = 256 :
             REM  ERROR TRAP ROUTINE
E1     1850  IF EN = 6 THEN  PRINT "FILE NOT FOUND" :
             GOTO 1890
23     1860  IF EL = 9 THEN  PRINT "DISK FULL"
FA     1870  IF EN = 8 THEN  PRINT "DISK WRITE-PROTECT
             ED" : GOTO 1890
48     1880  IF EL = 1950 OR EL = 610 OR EL = 1700 OR
             EL = 1720 THEN  PRINT "DISK ERROR"
       1890  HOME : PRINT "PRESS RETURN TO CONTINUE "; : POKE
             - 16368,0: GET Z$
       1900  ON (EL = 610) + 2 - (EL = 1700) + 3 - (EL
             = 1720) + 4 - (EL = 1950) GOTO 570,1660,16
             00,1920
EE     1910  HOME : VTAB 23 : PRINT "ERROR ";EN;" AT L
             INE ";EL : PRINT "PROGRAM TERMINATED" : EN
             D
EC     1920  TEXT : HOME : REM  CATALOG ROUTINE
F0     1930  VTAB 21 : HTAB 3 : HTAB 1 :
             INPUT "SLOT ?";SL$:SL = VAL (SL$): IF SL
             < 1 OR SL > 7 THEN 1930
C4     1940  VTAB 21 : HTAB 1 : HTAB 1 : HTAB SPC( 39) : HTAB 1 :
             INPUT "DRIVE ?";DR$:DR = VAL (DR$): IF DR
             < 1 OR DR > 2 THEN 1940
F6     1950  PRINT D5,"CATALOG", 7 - 4 = PRODOS
             );S";SL;",D";DR : IF PRODOS THEN  PRINT D5
             "PREFIX",S";SL;",D";DR
FE     1960  HOME : PRINT "PRESS RETURN TO CONTINUE ";
             : POKE - 16368,0: GET Z$
10     1970  POKE - 16304,0: GOTO 570

TOTAL: 6DD1

END OF LISTING 1
```

---

the output routines of BASIC, or COUT from assembly language, with Escape Control-O and Escape Control-N.

Note that MouseText characters are only available on the enhanced IIe, the Apple IIc, and the Apple IIGS. If you have an Apple II, II Plus or unenhanced IIe, MouseText characters are not available. Figure 1 shows the complete MouseText character set.

⌚	⯀	↰	⨯	✓	▚	↵
@	A	B	C	D	E	F	G

←	...	↓	↑	⎍	◀	▶
H	I	J	K	L	M	N	O

⊞	◆	⌘	—	∟	◣	▼
P	Q	R	S	T	U	V	W

⊏	⊐	¦	⊒	⊐	⌐
X	Y	Z	[	\	]	^	_

Figure 1: MouseText Characters

Note to GS owners: When you ran the first demonstration program that printed the ASCII values from 0 to 255, you may have noticed your cursor change to an underscore. This is related to a built-in feature of the GS that lets a program change the cursor by printing Control-^ (Control-Shift-6) followed by the ASCII value that you want for the cursor. In the first program, when I = 30, you get the equivalent of PRINT CHR$(30);CHR$(3). The CHR$(30) generates the Control-^, and the following character, Control-__ (Control-underscore) tells the computer to use an