

Graphics Workshop

The Shape

by Bill Fortenberry
2906 36 Ave.
Meridian, MS 39301

One of the things that separates the Apple II from other micro computers is its powerful graphics, and at the core of these capabilities is the shape table. The shape table provides the user with the ability to define a HIRES graphic image, identify it by an index number, and then rotate, scale, and draw the shape in any of the Apple's HIRES colors. Creating a shape table is comparable to doing time for armed robbery. It's a long, tedious process where one coding or typing error can put you right back at the beginning.

THE SHAPE to the rescue! THE SHAPE is an Applesoft program that allows you to create, view, edit, and save shape tables. The shapes built by the program are in the standard hex format and can be used in any Applesoft, Integer or assembly language program.

HOW TO BUILD A SHAPE TABLE

The complete instructions for building a shape table are found in the Applesoft reference manual (Programmer's Aid #1 manual for Integer), but to get everyone going, I will give a summary of them here. The first step is to decide on a shape and draw it on graph paper. Now pick a starting point and begin drawing arrows that point from one dot to the next, and only turn on 90 degree angles (these arrows are usually called the **vectors** of the shape). The arrows must then be coded as a 3 digit binary number (which must then be coded as an 8 bit byte). One of the problems with shapes is the way you cram these 3 bit numbers down to 8 bits (THE SHAPE does this for you).

After each byte is set, you must build the **shape table index**. Like the index in a book, the shape table index tells where each shape is located. When you finish the index, you are ready to call the monitor, enter the table, exit the monitor, test the shape, see that it's wrong, look at your arrows again, look at your numbers again, check each byte again, find one mistake, enter the monitor, type the table, exit the monitor, test the table, see that it's wrong, look at your arrows again, look at your numbers again, check the bytes again, sell your Apple, buy a Northstar. If doing all of the above does not appeal to you, read on.

LO-RES TO HI-RES

Late one night, after I had been trying for about 4 hours to set up a shape table for a new game, I decided to write a program to do the actual coding and typing for me. Since I wasn't going to code the vectors myself, I needed a way to get the shape some place where the computer could "see" it. I chose the LO-RES screen mainly because of the availability of the **SCRN** function. This enabled me to "see" each dot on the screen.

Next I had to get the shape on the screen. Out of this need came a simple horizontal and vertical line drawing routine. (I always draw a shape on graph paper so the lines are easy to

see.) Now, how would I code the shape? Since Apple shapes are picky about moving up two consecutive (no plot) spaces in a row, and since I had the problem of trying to put 9 bits of information in an 8 bit word, I decided to **scan the shape right to left, and top to bottom. This totally eliminates any vectors with a value of zero** (the signal for the end of the shape table), and when poking the vectors in memory I only put two vectors in a byte.

My goal with THE SHAPE was to write a program that would give the user a completed shape table (index + definitions) while requiring minimum input.

INSTRUCTIONS FOR THE SHAPE

OPTION ONE — BUILD A SHAPE TABLE

When you run the program, a menu will appear with 2 options. Let's take option one and build a table (I'll talk about option two later.) First, the text page will disappear and the blank graphics page will be displayed. At the bottom of the screen, you will be asked if you want the **turtle** or **line mode**. Here is how the **line mode** works.

After you pick **line mode**, the screen will display:

HLIN FROM **PLOT**

This means that you are in the **horizontal line mode** with the **plot on**. Type a **5** and press return. The Apple replies with **TO**. Now type a **9** and press return. Next the screen displays **ON**. Type **7** and press return. This will draw a **horizontal line from 5,7 to 9,7** (it's the big red line toward the top of the screen).

Now type a **V** and hit return. The screen should say:

VLIN FROM **PLOT**

You are now in the **vertical line mode** (these modes act just like **HLIN** and **VLIN** in BASIC). Try the same sequence of numbers as above. Did you get a vertical line?

Type an **E** and return. The screen will display:

HLIN FROM **ERASE**

Any lines you draw now will be black (or erased). Notice that you returned to the **HLIN mode**.

Now try an **F** and press return. Where did that white square come from? This is the **free cursor mode**. Type some **I's**, **J's**, **K's** and **M's** and the cursor should move around the screen. Press the **P** key. Now when you move the cursor it should leave a red dot. Now put the cursor in the middle of one of your lines, hit the **E** key, and move the cursor. It should have left a black dot. When you press **P** or **E** in the **free mode**, it will **plot** or **erase** the point the cursor is on. By the way, the **coordinates of the cursor** are displayed at the bottom of the screen. The free mode is designed for touching up shapes and finding coordinates for the line statements. To exit from the free mode, press **Q**. In any mode, you may press the question mark key to get a list of available commands.

When you are in the **HLIN** or **VLIN** mode, pressing **Q** will cause the program to start coding your shape.

The screen will now display:

LOCATING SHAPE EDGES

The program is now finding the top, bottom and sides of the shape. After a while, it will display:

CODING PLOT VECTORS

The program is now looking at each dot on the screen and coding it as a number. Then the screen will show:

POKING SHAPE DEFINITION

The numbers that represent your shape are being combined and poked into memory. When the screen shows:

THIS DEFINITION STARTS ON THE WHITE DOT. PRESS RETURN TO PLOT SHAPE

You should find the white dot and mark its position on your graph paper. This is where the Apple will begin to draw your shape (the coordinates in the **DRAW** statement).

SHAPE DISPLAY

We are now ready to see the shape. Press Return and enter the **scale** and **rotation** just as you would in a BASIC program. Choose the color selection with the **first letter** of the desired color and press Return. You should now see your shape in glorious HI-RES graphics. When you press Return, the program will ask if you want to draw the shape again. If you press **Y**, you can enter new coordinates, etc. and redraw it. If you answer **N**, the computer will ask if you want to edit the shape. If you didn't like the way it looked, answer **Y** and you will go back to the edit mode. If you are satisfied with the shape type an **N**.

Now the Apple will ask if you want to keep this shape as a **base** for the next shape. If, for example, you are building a series of shapes that will show a man walking. Since only the legs will move, why should you have to redraw the head and torso for each figure? With THE SHAPE you don't have to. Just answer **Y** to the question and you will be put right back into the Enter mode. (Except that the last shape is now stored safely away) and you are looking at the next shape in line. Now just change the position of the legs, and proceed to the next shape in the sequence.

THE TURTLE MODE

If you select the **Turtle mode** instead of the **Line mode**, the screen will display:

MOVE TO START AND PRESS 'P'

Use the **I**, **J**, **K** and **M** keys to move the cursor to where you want to start your shape, and then press **P**. The screen will display:

PLOT ON
X=xx Y=yy

This tells you that the **plotting** is turned ON and gives you the values for the **x** and **y** coordinates. Just keep pressing the cursor keys to draw your shape. To turn the plot OFF, just press **E**. If you make a mistake while drawing in the turtle mode, the **ESC** key will cause the cursor to **back up** along your shape and erase the errors.

LINE VS. TURTLE

In the **line mode**, the computer starts to code your shape when you push the "Q" key. In the **turtle mode**, the computer follows **every plot as you make it**. Because of this, the turtle mode will code a shape faster (if you draw it with a minimum of moves).

With the **turtle mode**, you **don't** get to use one shape as a base for another. Also, if you choose to edit a shape, you must start over with a blank screen.

The beginner will feel safer with the line mode, while an experienced programmer will want to use the turtle mode to draw more efficient shapes. No matter which mode you use, the computer will generate an index for you. After you have finished your table, the computer will give you the commands to save your shape Table.

OPTION TWO — THE POKE CALCULATOR

When you build a Shape Table with this program, the Shape Table is stored in HI-RES page one. When you load the table back into memory to use with your own program, you must put it in another memory location.

Remember that when you use a Shape Table with Applesoft, you must **poke the address of the table into locations 232 and 233**. The POKE calculator will make this easier for you. When you select Option Two from the main menu and enter the address where you want to put the table, the program will show you what POKES to use in your program.

MEMORY USAGE BY THE SHAPE

In order for this program to work properly you must have a 48K machine. The program resides from LOMEM up to HI-RES page one. Page one of HI-RES is reserved for the shape table as it is being built. Page Two is used to show the shape, and all memory above page Two is set aside for variable storage. This program does **not** contain an **ONERR GOTO** section (to conserve space), but input is checked and most bad input is ignored. You should become familiar with the program before building a large shape Table.

VARIABLES USED IN THE PROGRAM

Most of the variables serve double duty to conserve memory, but here are some of the primary variables:

- AS — input string
- SN — number of shapes in the table
- SS — Shape Table starting location
- G — array to hold shape vectors
- SD — array that holds the starting location of each definition
- SC — loop counter (counts up to # of shapes in table)
- QQ — throw away variable
- B & C — input in line mode
- X & Y — screen coordinates
- ZC — old screen color
- CZ — current screen color
- X,Y & Z — misc. pointers

THE PROGRAM LOGIC

The program does not have many comments because of memory limits. The table below should help you decipher the program section and their logic.

```

350-380 These lines display the title page
200-270 Display the menu and accept the user's input
290-390 Find out how many shapes are in the table
410-460 The main program loop
480-710 Draw a horizontal line or branch to Vlin or Free Cursor
730-960 Same as above for vertical lines
980-1420 Free cursor + turning ON or OFF individual points
1190-1360 Make an imaginary box around the figure
1380-1630 Scan shape and fill array G with shape vectors
1650-1830 Poke the shape vectors into memory
1850-1980 Let the user view the shape and set up the index
2000-2180 Let the user edit and use the shape again
2200 Foot of the main loop
2220-2310 Print details about the table and BSAVE command
2320-2400 Plot and erase subroutines for the line mode
2420 General purpose text centering subroutine
2440-2520 Calculate POKE values for the user
2540-2610 A short hex to decimal converter
2630-2970 The code for the Turtle mode

```

The Shape is by no means complete. Many things were intentionally omitted because they had already been done or would have required program chaining. The shape coding algorithm could also be improved, although it works fine in the present form. Hopefully, THE SHAPE will give other programmers a basis for developing better shape table utilities.

```

10 REM *****
20 REM * THE SHAPE *
30 REM * BILL FORTENBERRY *
40 REM * COPYRIGHT (C) 1982 *
50 REM * BY MICRO-SPARC INC *
60 REM * LINCOLN, MA, 01773 *
70 REM *****
80 LOMEM: 24576
90 TEXT: HOME: PRINT "*****"
100 FOR I = 2 TO 12: VTAB I: PRINT "": TAB(40): "": NEXT
110 VTAB 1
120 PRINT "*****"
130 INVERSE :AS = " " : VTAB 3: GOSUB 2420: VTAB 5 : GOSUB 2420
140 VTAB 4: INVERSE :AS = " " : THE SHAPE ": GOSUB 2420: NORMAL
150 VTAB 2:AS = "APPLE 11 SHAPE TABLE UTILITY": GOSUB 2420
160 VTAB 9:AS = "BY: BILL FORTENBERRY": GOSUB 2420
170 AS = "(C) 1982 BY MICRO-SPARC, INC.": VTAB 11: GOSUB 2420
180 POKE 34,13
190 REM *****
200 VTAB 15: PRINT "WOULD YOU LIKE TO": PRINT : PRINT " 1. BUILD A SHAPE TABLE: PRINT " 2. CALCULATE POKE VALUES: PRINT " 3. INSTRUCTIONS & DEMO"
210 PRINT : INPUT "TYPE THE NUMBER (1-3) OF YOUR CHOICE:
=3:AS
220 IF AS = "" THEN 200
230 A = VAL (AS)
240 IF A < 1 OR A > 3 THEN 200
250 HOME
260 IF A = 3 THEN PRINT CHR$(4):"RUN SHAPE INSTRUCTIONS"
270 IF A = 2 THEN 2430
280 REM *****
290 VTAB 17: INPUT "HOW MANY SHAPES IN THIS TABLE ";SN
300 IF SN = "" THEN 290
310 SN = VAL (SN)
320 IF SN < = 0 THEN 290
330 SS = 13000
340 REM POKE SHAPE LOCATION
350 POKE 232,SS - INT (SS / 256) * 256
360 POKE 233, INT (SS / 256)
370 DIM G(1445),SD(SN + 2)
380 SD(1) = SS + 2 + SN * 2
390 G@: POKE 34,20
400 REM *****

```

```

410 FOR SC = 1 TO SN
420 C@ = FRE (C@)
430 G@: HOME : POKE 34,20: HOME
440 HOME : PRINT "SHAPE #":SC: PRINT "T)":LRTLE OR L:LINE MODE (T/L) ": INPUT "":AS: IF AS = "" THEN 2620
450 COLOR@:1:F = 0
460 HOME : PRINT TAB(33)"PLOT": POKE 33,50
470 REM *****
480 HOME : PRINT "HLIN FROM ": INPUT "":AS
490 IF AS = "" THEN 480
500 IF AS = "V" THEN HOME : GOTO 730
510 IF AS = "H" THEN 980
520 IF AS = "M" THEN 2330
530 IF AS = "E" THEN 2370
540 IF AS = "Q" THEN 1190
550 IF AS = "Z" THEN 2980
560 PRINT "TO ": INPUT "":BS
570 IF BS = "" THEN 560
580 IF BS = "Z" THEN 2980
590 B = VAL (BS)
600 IF B > 38 THEN B = 38
610 IF B < 1 THEN B = 1
620 PRINT "ON ": INPUT "":CS
630 IF CS = "Z" THEN 2980
640 IF CS = "" THEN 620
650 C = VAL (CS)
660 IF C > 38 THEN C = 38
670 IF C < 1 THEN C = 1
680 A = VAL (AS)
690 IF A < 1 THEN A = 1
700 IF A > 38 THEN A = 38
710 HLIN A,B AT C: PRINT : HOME : GOTO 480
720 REM *****
730 INPUT "VLIN FROM ":AS
740 IF AS = "" THEN 730
750 IF AS = "H" THEN HOME : GOTO 480
760 IF AS = "V" THEN 980
770 IF AS = "M" THEN 2330
780 IF AS = "E" THEN 2370
790 IF AS = "Q" THEN 1190
800 IF AS = "Z" THEN 2980
810 INPUT "TO ":BS
820 IF BS = "Z" THEN 2980
830 IF BS = "" THEN 810
840 B = VAL (BS)
850 IF B > 38 THEN B = 38
860 IF B < 1 THEN B = 1
870 INPUT "AT ":C
880 IF CS = "Z" THEN 2980
890 IF CS = "" THEN 870
900 C = VAL (CS)
910 IF C > 38 THEN C = 38
920 IF C < 1 THEN C = 1
930 A = VAL (AS)
940 IF A < 1 THEN A = 1
950 IF A > 38 THEN A = 38
960 VLIN A,B AT C: PRINT : HOME : GOTO 730
970 REM *****
980 X = 20:Y = 20
990 ZC = SCRN(X,Y): COLOR@:15: PLOT X,Y
1000 POKE 33,40
1010 HOME : PRINT "X=";X;Y=";Y
1020 GET AS
1030 IF AS = "Q" THEN COLOR@:ZC: PLOT X,Y: GOTO 2330
1040 IF AS = "M" THEN ZC = 1: GOTO 1020
1050 IF AS = "E" THEN ZC = 0: GOTO 1020
1060 IF AS = "Z" THEN 3040
1070 XX = X:YY = Y
1080 IF AS = "H" AND X > 1 THEN YY = Y - 1: GOTO 1130
1090 IF AS = "M" AND Y < 38 THEN YY = Y + 1: GOTO 1130
1100 IF AS = "V" AND X > 1 THEN XX = X - 1: GOTO 1130
1110 IF AS = "M" AND X < 38 THEN XX = X + 1: GOTO 1130
1120 GOTO 1020
1130 CZ = SCRN(XX,YY)
1140 COLOR@:15: PLOT XX,YY: COLOR@:ZC: PLOT X,Y:XX = XX:Y = Y
1150 ZC = CZ
1160 HOME : PRINT "X=";X;Y=";Y
1170 GOTO 1020
1180 REM *****
1190 L = 0:Z = 0
1200 POKE 33,40: HOME
1210 PRINT "LOCATING SHAPE DEFINITION"
1220 FOR D = 1 TO 39: IF SCRN(L + 1,D) = 1 THEN Z = 1
1230 NEXT
1240 IF Z = 0 THEN L = L + 1: GOTO 1220
1250 J = 39:Z = 0
1260 FOR D = 1 TO 39: IF SCRN(J - 1,D) = 1 THEN Z = 1
1270 NEXT
1280 IF Z = 0 THEN J = J - 1: GOTO 1260
1290 Z = 0:K = 39
1300 FOR D = 1 TO 39: IF SCRN(D,K - 1) = 1 THEN Z = 1
1310 NEXT
1320 IF Z = 0 THEN K = K - 1: GOTO 1300
1330 Z = 0:J = 0
1340 FOR D = 1 TO 39: IF SCRN(D,J + 1) = 1 THEN Z = 1
1350 NEXT
1360 IF Z = 0 THEN J = J + 1: GOTO 1340
1370 REM *****
1380 Z = 1:D = 1:X = 1:L:Y = 1 + 1
1390 PRINT "CODING PLOT VECTORS"
1400 X = X + 1:SX = X:SY = Y: IF SCRN(X,Y) = 0 THEN 1400
1410 BX = X:BY = Y
1420 GOTO 1440
1430 X = X + D
1440 IF D = 1 AND X = J - 1 AND SCRN(X,Y) = 1 THEN G(Z) = 6*Y = Y + 1:D = -D:Z = Z + 1: GOTO 1490
1450 IF D = 1 AND X = J - 1 AND SCRN(X,Y) = 0 THEN G(Z) = 2*Y = Y + 1:D = -D:Z = Z + 1: GOTO 1490
1460 IF D = -1 AND X = L + 1 AND SCRN(X,Y) = 0 THEN G(Z) = 2*Y = Y + 1:D = -D:Z = Z + 1: GOTO 1490
1470 IF D = -1 AND X = L + 1 AND SCRN(X,Y) = 1 THEN G(Z) = 6*Y = Y + 1:D = -D:Z = Z + 1: GOTO 1490
1480 GOTO 1500
1490 IF Y > = K THEN 1630
1500 F = 0
1510 FOR B = L TO J
1520 IF SCRN(B,Y) = 1 THEN F = 1

```

```

1530 NEXT B
1540 IF F = 0 THEN G(Z) = 2:Y = Y + 1:Z = Z + 1: GOTO 1490
1550 F = 0
1560 IF Y > = K THEN 1630
1570 GOTO 1440
1580 IF SCRN(X,Y) = 1 AND D = 1 THEN G(Z) = 5:Z = Z + 1
1590 IF SCRN(X,Y) = -1 AND D = -1 THEN G(Z) = 7:Z = Z + 1
1600 IF SCRN(X,Y) = 0 AND D = 1 THEN G(Z) = 1:Z = Z + 1
1610 IF SCRN(X,Y) = 0 AND D = -1 THEN G(Z) = 3:Z = Z + 1
1620 GOTO 1430
1630 G(Z) = 10
1640 REM

POKE SHAPE
1650 A = SD(SC)
1660 T = A
1670 Z = 0
1680 PRINT "POKING SHAPE DEFINITION"
1690 Z = Z + 1
1700 IF G(Z) = 10 THEN 1790
1710 K = G(Z):Z = Z + 1
1720 IF G(Z) = 10 THEN 1780
1730 K = K + G(Z) * 8:Z = Z + 1
1740 IF G(Z) = 10 THEN 1780
1750 IF G(Z) < 4 AND G(Z) > 0 THEN K = K + G(Z) * 64:Z = Z + 1
1760 Z = Z - 1
1770 POKE A,K:A = A + 1: GOTO 1690
1780 POKE A,K:A = A + 1
1790 POKE A,0
1800 PRINT "": REM CTRL-G
1810 COLOR= 15: PLOT SX,SY
1820 HOME: PRINT "THIS DEFINITION STARTS ON THE WHITE DOT."
1830 VTAB 24: INPUT "PRESS RETURN TO PLOT SHAPE ";AS
1840 REM

DRAW SHAPE
1850 HOME
1860 INPUT "SCALE = ";G$:G = VAL(G$): IF G < 1 THEN 1860
1870 SCALE= G: INPUT "ROT = ";G$:G = VAL(G$): ROT= G
1880 HOME: PRINT "G)REEN P)URPLE O)RANGE B)LUE W)HITE"
: PRINT: INPUT "ENTER FIRST LETTER OF COLOR ";AS
1890 IF AS < > "G" AND AS < > "M" AND AS < > "B" AND AS < > "P" AND AS < > "O" THEN 1880
1900 REM POKE THE DIRECTORY
1910 POKE SS,SN: POKE SS + 1,0
1920 X = SS + 1
1930 FOR I = 1 TO SC
1940 B = SD(I) - SS
1950 X = X + 1:TA = INT(B / 256)
1960 POKE X,B - TA * 256
1970 X = X + 1: POKE X,TA
1980 NEXT I
1990 REM

EDIT THE SHAPE
2000 IF AS = "W" THEN HCOLOR= 3
2010 IF AS = "G" THEN HCOLOR= 1
2020 IF AS = "P" THEN HCOLOR= 2
2030 IF AS = "O" THEN HCOLOR= 5
2040 IF AS = "B" THEN HCOLOR= 6
2050 HGR2: DRAW SC AT 140,40
2060 INPUT AS: POKE - 16298,0: POKE - 16300,0: POKE - 16301,0
2070 SD(SC + 1) = A + 1
2080 IF TM = 0 THEN COLOR= 1: PLOT BX,BY
2090 HOME
2100 INPUT "DRAW THIS SHAPE AGAIN (Y/N) ";AS: IF AS = "" THEN EN 2090
2110 IF AS = "Y" THEN 1850
2120 HOME
2130 INPUT "EDIT THIS SHAPE (Y/N) ";AS: IF AS = "" THEN 2130
2140 IF AS = "Y" AND TM = 1 THEN 2620
2150 IF AS = "Y" THEN 450
2160 IF TM = 1 THEN 2200
2170 HOME: INPUT "USE THIS SHAPE AS A BASE FOR ANOTHER (Y/N)";AS
2180 IF AS = "Y" AND SC < SN THEN SC = SC + 1: GOTO 450
2190 REM

NEXT SHAPE
2200 NEXT SC
2210 REM

TABLE DATA
2220 TEXT
2230 HOME: PRINT "SHAPE TABLE DATA"
2240 PRINT: PRINT: PRINT "SHAPE TABLE STARTS AT ";SD(SC)
2250 PRINT: PRINT "AND ENDS AT ";SD(SC)
2260 PRINT: PRINT "TABLE LENGTH IS ";SD(SC) - SS + 1
2270 REM SHAPE LOCATION
2280 PRINT: PRINT "SAVE SHAPE WITH THIS COMMAND -"
2290 PRINT
2300 PRINT "BSAVE "; INVERSE: PRINT "FILE NAME"; NORMAL: PRINT ",A";SS";",L";SD(SC) - SS + 1
2310 END
2320 REM

PLOT & ERASE SUBS
2330 POKE 33,40: HOME:
2340 PRINT TAB(33)"PLOT": POKE 33,30
2350 COLOR= 1
2360 HOME: GOTO 480
2370 POKE 33,40: HOME
2380 PRINT TAB(33)"ERASE": POKE 33,30
2390 COLOR= 0
2400 HOME: GOTO 480

```

```

2410 REM

CENTER SUB
2420 HTAB 21 - LEN(AS) / 2: PRINT AS: RETURN
2430 REM

POKE CALCS
2440 VTAB 19: PRINT "START HEX NUMBERS WITH A '$!'"
2450 VTAB 17: INPUT "WHERE WILL YOU PUT THE TABLE ";X$
2460 Y = VAL(X$)
2470 IF LEFT$(X$,1) = "$" THEN X$ = RIGHT$(X$, LEN(X$) - 1): GOSUB 2540
2480 X = Y - INT(Y / 256) * 256:Z = INT(Y / 256)
2490 HOME: PRINT "USE THESE POKES IN YOUR PROGRAM BEFORE YOU DRAW YOUR FIRST SHAPE"
2500 PRINT: PRINT "POKE 232,";X: PRINT "POKE 233,";Z
2510 PRINT
2520 END
2530 REM

HEX CONVERTER
2540 Y = 0:Z = 0
2550 FOR I = LEN(X$) - 1 TO 0 STEP - 1
2560 Z = Z + 1
2570 YI = ASC(MID$(X$,Z,1)) - 48
2580 IF YI > 16 THEN YI = YI - 7
2590 Y = Y + YI * 16 ^ I
2600 NEXT
2610 RETURN
2620 REM

TURTLE MODE
2630 OR: POKE 34,20: HOME: PRINT "MOVE DOT TO START & PRESS 'P'": POKE 34,21
2640 Z = 1:X = 20:Y = X:XX = Y:YY = Y:TM = 1
2650 COLOR= 0: PLOT XX,YY: COLOR= 15: PLOT X,Y:XX = X:YY = Y: HOME: PRINT "X=";X;" Y=";Y: GET AS
2660 IF AS = "P" THEN SX = X:SY = Y: GOTO 2720
2670 IF AS = "I" AND Y > 0 THEN Y = Y - 1
2680 IF AS = "M" AND Y < 39 THEN Y = Y + 1
2690 IF AS = "K" AND X < 39 THEN X = X + 1
2700 IF AS = "J" AND X > 0 THEN X = X - 1
2710 GOTO 2650
2720 P = 1:ZC = 0: POKE 34,20
2730 XX = X:YY = Y: HOME: PRINT "X=";X;" Y=";Y
2740 IF P = 1 THEN PRINT "PLOT ON": GOTO 2760
2750 PRINT "PLOT OFF"
2760 GET AS
2770 IF AS = "M" AND Y < 39 THEN G(Z) = 2:Z = Z + 1:F = 0:Y = Y + 1: GOTO 2870
2780 IF AS = "J" AND X > 0 THEN G(Z) = 3:Z = Z + 1:F = 0:X = X - 1: GOTO 2870
2790 IF AS = "K" AND X < 39 THEN G(Z) = 1:Z = Z + 1:F = 0:X = X + 1: GOTO 2870
2800 IF AS = "I" AND Y > 0 AND (F = 0 OR P = 1) THEN G(Z) = 0:F = 1:Z = Z + 1:Y = Y - 1: GOTO 2870
2810 IF AS = "?" THEN 3110
2820 IF AS = "E" THEN P = 0: GOTO 2730
2830 IF AS = "P" THEN P = 1: GOTO 2730
2840 IF AS = "Q" THEN G(Z) = 10: COLOR= ZC: PLOT X,Y: GOTO 1650
2850 IF AS = CHR$(27) THEN 2900: REM ESC KEY
2860 GOTO 2730
2870 CZ = SCRN(X,Y): COLOR= ZC: PLOT XX,YY: COLOR= 15: PLOT X,Y:ZC = CZ
2880 IF P = 1 THEN G(Z - 1) = G(Z - 1) + 4: COLOR= 1: PLOT XX,YY
2890 GOTO 2730
2900 IF Z > 0 THEN Z = Z - 1
2910 IF Z < = 0 THEN 2730
2920 ON G(Z) + 1 GOSUB 2940,2950,2960,2970,2940,2950,2960,2970
2930 COLOR= 15: PLOT X,Y: COLOR= 0: PLOT XX,YY: GOTO 2730
2940 Y = Y + 1: RETURN
2950 X = X - 1: RETURN
2960 Y = Y - 1: RETURN
2970 X = X + 1: RETURN
2980 REM

LINE MODE HELP
2990 POKE 33,40: HOME
3000 PRINT "Y-> DRAW VERTICAL LINES": PRINT "H-> DRAW HORIZONTAL LINES": PRINT "F-> ENTER FREE CURSOR MODE": INPUT "PRESS RETURN";AS: HOME
3010 PRINT "DRAW LINES BY GIVING THE ENDPOINTS AND": PRINT "LEVEL TO DRAW ON": INPUT "PRESS RETURN";AS: HOME
3020 PRINT "F-> TURNS PLOT ON (DRAW LINES)": PRINT "E-> TURNS PLOT OFF (ERASE LINES)": INPUT "PRESS RETURN ";AS: HOME
3030 PRINT "Q-> QUIT DRAWING AND CODE THE SHAPE": INPUT "PRESS RETURN";AS: HOME: GOTO 2330
3040 REM

FREE CUR HELP
3050 HOME
3060 PRINT "I,J,K,M KEYS MOVE THE CURSOR": PRINT "P-> PLOT A POINT": PRINT "E-> ERASE A POINT": INPUT "PRESS RETURN";AS
3070 HOME
3080 PRINT "P & E ONLY EFFECT THE POINT THE CURSOR": PRINT "IS RESTING ON, NO OTHERS!": INPUT "PRESS RETURN";AS: HOME
3090 PRINT "Q-> RETURNS YOU TO THE HLIN MODE WITH": PRINT "THE PLOT ON": INPUT "PRESS RETURN";AS: HOME
3100 GOTO 1000
3110 REM

TURTLE MODE HELP
3120 HOME: PRINT "I,J,K,M KEYS MOVE THE CURSOR": PRINT "P-> TURN PLOTTING ON": PRINT "E-> TURN PLOTTING OFF": INPUT "PRESS RETURN";AS: HOME
3130 PRINT "THE ESC KEY WILL ALLOW YOU TO 'BACK UP'": PRINT "AND ERASE ERRORS": INPUT "PRESS RETURN";AS: HOME
3140 PRINT "Q-> QUIT DRAWING AND CODE THE SHAPE": INPUT "PRESS RETURN";AS: HOME: GOTO 2730

```