

SUPERGRAPHICS GS

Use these powerful ampersand commands in your Applesoft programs and take control of the IIGS's graphics screen in Super Hi-Res mode. Use complete palette control to achieve animation and other dazzling effects.

The Apple IIGS has features that fulfill many users' dreams: Super Hi-Res graphics, advanced sound generation, and a 16-bit central processor chip. But unless you're an assembly language programmer, much of the IIGS's power is locked away in its ROMs. Applesoft cannot access the IIGS Toolbox, which contains the code for the IIGS's advanced features.

SuperGR allows you to use the power of the IIGS's Super Hi-Res graphics from your own Applesoft programs, using ampersand (&) commands for full machine-language speed. The graphics primitives in SuperGR bring to you full control of the 320 × 200 screen, its palettes, soft switches, and viewports. You'll find that many of SuperGR's commands are similar to Applesoft's; if you've worked with Hi-Res graphics in BASIC before, you'll learn SuperGR quickly.

USING SUPERGR

SuperGR has 15 ampersand commands; those not based on existing Applesoft commands are equivalent to Toolbox commands or come from using other computers. Here's a description of each command and its syntax.

&HGR

This command displays the Super Hi-Res screen using 320 × 200 coordinates, turns on shadowing, clears the screen to color 0 (black), and initializes the standard IIGS color table. It also sets up a viewport the same size as the screen, 320 × 200.

&TEXT

Turn off Super Hi-Res graphics with this command. But there's no guarantee you'll be returned to text, since an additional soft switch is involved (see &DRAW and &XDRAW).

&RECALL

You can display the Super Hi-Res screen without erasing it with this command.

&CLEAR color

You can clear the *entire* screen to a specified color. The color parameter must be numbered in the range 0-15. This command is not affected by the viewport command.

&HCOLOR = color

This command sets the current drawing color. The color parameter specified must be numbered in the range 0-15. It'll be used by the following commands: HPLLOT, FILL, CIRCLE, RECTANGLE. The default colors are shown in Table 1.

TABLE 1: Default Color Table

Number	Value	Name
0	000	Black
1	777	Dark Gray
2	841	Brown
3	72C	Purple
4	00F	Blue
5	080	Dark Green
6	F70	Orange
7	D00	Red
8	FA9	Flesh
9	FF0	Yellow
A	0E0	Green
B	4DF	Light Blue
C	DAF	Lilac
D	78F	Periwinkle Blue
E	CCC	Light Gray
F	FFF	White

&P. palette

Sets the current palette number with this command. *Palette* must fall in the range 0-15. All other commands work only with the current setting. For example, &HCOLOR = selects a color from the current palette and DEF defines colors in the current palette. You must be

very careful when using different palettes, because only one palette can be used per line on the screen. All drawing commands affect the palette setting for the appropriate lines. For example, a vertical line drawn from row 10 to row 15 will change the palette number on those six screen lines to the current palette. If different palettes have been used on these lines, they will be changed and the old colors will be replaced by colors contained in the current palette. I recommend that you wait to use this command until you become familiar with the program.

&DEF color,red,green,blue

Here, you can set the red, green, and blue parameters of *color*. *Color*, *red*, *green*, and *blue* must be in the range 0-15. The lowest value (0) is the darkest and gives you no intensity for that color, whereas the highest value (15) is a very high intensity. Setting all three colors (red, green, blue) to high intensity (15) will create white. Also, if all three colors are set to the same level, you will get a gray scale. Keep in mind that mixing colors using light is different from mixing paints. For example, yellow and orange are both obtained by mixing red and green, but using different values. Since there are 16 values (intensities) of each color, you can choose from a total of 4,096 different color combinations.

&V(x1,y1),(x2,y2)

This command lets you define the current viewport clipping rectangle. Any drawing outside of this rectangle will not be displayed. The upper left corner is defined by the coordinates *x1,y1* and the lower right corner by *x2,y2*. The viewport must be on the screen, and the values of *x2,y2* must be greater than *x1,y1*.

&HPOINT [TO] x1,y1 [TO x2,y2]

If a coordinate is given by itself, HPOINT will simply plot a dot in the current color. If it is then followed by a TO *x,y*, a line will be drawn to the next point. The only limit to the number of TO's that can appear is the length of the program line. A TO may also immediately follow the HPOINT statement. In this case, a line is drawn from the last plotted point. Remember that this point is modified by FILL, CIRCLE, RECTANGLE, READ, and VIEWPORT in addition to HPOINT. Lines will only be drawn within the viewport, but you may plot the coordinates to lie off the screen if you wish.

&F(x,y),border

This command fills an area of the screen with the current color. The fill starts at the point *x,y*; the region to be filled is bordered by the viewport and the color *border*. For example, if the point 100,100 lies inside a circle drawn when the drawing color was 12, the command &F(100,100),12 would fill the circle with the current drawing color. The fill coordinate must fall within the viewport, or no fill will be made.

&C(x,y),radius

Draws a circle in the current color centered at *x,y* with a size of *radius*, where *radius* is equal to half the diameter you want the circle to be. The circle can lie outside of the screen or viewport.

&R(x1,y1),(x2,y2)

Draw a rectangle in the current color with this command, using *x1,y1* as the upper-left coordinates and *x2,y2* as the lower-right coordinates. The rectangle may lie outside of the viewport.

&READ (x,y),color%

This command allows you to read the color located at coordinates *x,y* and pass it back to *color%*. *Color%* must be a variable of type integer. The coordinates must be on the screen, but they are not affected by the viewport. The palette number is not returned.

&XDRAW

This command lets you turn off memory shadowing of Super Hi-Res graphics. When shadowing is turned off, the current screen is

locked and any drawing command following it will not be displayed. This allows a semi-page flipping mode, in conjunction with the &DRAW command.

&DRAW

Turn shadowing back on with this command, and display any changes made since XDRAW. Shadowing is also disabled when Super Hi-Res is not being displayed. If you wish to draw while displaying a text message (see demo), be sure to execute a DRAW after a RECALL.

ENTERING THE PROGRAM

SuperGR was written using the Merlin Pro assembler and may require some changes for other assemblers. If you do use another assembler, be sure it allows 65816 or 65802 opcodes or has a macro library to emulate them. The SuperGR source code can be found in Listing 1.

If you do not have an assembler, you'll be required to enter the monitor and type in the hexadecimal codes for the program from Listing 2. To save SuperGR enter the following:

```
BSAVE SUPERGR,AS8E00,LS6E0
```

In lines 142-157 and elsewhere throughout Listing 1, the hex opcodes and operands were entered manually into the object file with the HEX directive, because this version of the assembler could not handle 24-bit addresses.

Listings 3 and 4 contain Applesoft programs that demonstrate the use of SuperGR commands. Type in the program in Listing 3 and save it with the command:

```
SAVE SUPERGR.DEMO1
```

Type in the Applesoft program in Listing 4 and save it with the command:

```
SAVE SUPERGR.DEMO2
```

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

To use SuperGR, simply place the following program line at the beginning of your program:

```
PRINT CHR$(4);"BRUN SUPERGR"; HIMEM: 35328
```

This line will set up the & vectors only. You must execute an &HGR to initialize the screen. The HIMEM will protect SuperGR from any variables used in your Applesoft program and leaves room for ProDOS buffers.

SUPERGR TECHNIQUES

Many special effects can be created by changing the color definitions. Animation and fade-outs can be created by changing the intensity. For example, a detailed drawing using several colors will be rendered invisible if all colors are defined to have the same value. If the color intensities are slowly adjusted, the picture will magically appear (see Listing 3). Animation can be performed by drawing each animation frame in a separate color and then "cycling" them one by one, as in this short program:

```
10 & HGR: FOR I = 1 TO 15: & DEF 1,0,0,0: & HCOLOR=
  I: & C(160,100): I * 11: NEXT
20 FOR J = 1 TO 15: & DEF 1,7,8,15: FOR K = 1 TO 80:
  NEXT: & DEF 1,0,0,0: NEXT: GOTO 20
```

Many special effects can also be created by changing the viewport. For example, to create a semicircle, try the following:

```
& HGR: & HCOLOR= 15: & V(0,0),(160,100): & C(160,
  100) 99
```

Careful study of the demonstration in Listings 3 and 4 should help you in becoming familiar with the numerous commands.

HOW THE PROGRAM WORKS

The pixel data for Super Hi-Res graphics begins at location \$2000

in bank SE1. When memory shadowing is on, you must write to \$2000 in bank \$01, which contains the fast RAM. Every time a value is stored in bank \$01 in Super Hi-Res memory, the same value is stored in bank SE1. When the &XDRAW command is invoked, this shadowing is stopped, which means that the data is stored only in bank \$01, not in bank SE1; therefore, changes are not displayed on the screen.

The Apple IIGS's Super Hi-Res graphics use a linear-mapping method for storing the scanlines (a scanline is simply a single row on the graphics display). Linear mapping means that each line follows the previous one in memory. Each line consists of 160 (\$A0) bytes, with 2 pixels per byte in 320 mode and 4 pixels per byte in 640 mode. We can now find the location of any scanline memory. For example, line 0 begins at \$2000, line 1 begins at \$20A0, and so forth. The last line, line 199, is located from \$9C60 to \$9CFF.

Following the pixel data, from \$9D00 to \$9DC7, are the Scanline Control Bytes (SCBs). There is one byte per scanline and these bytes contain information concerning all pixels of the entire line. Bits 0-3 of the SCB byte contain the number of that line's color table. There are 16 different color tables (palettes) to choose from. Bit 4 is always 0. Bit 5 determines whether the fill mode is on or off; a value of one is on and zero is off. Bit 6 determines whether an interrupt will be generated when the scanline is refreshed by the CRT, and bit 7 determines the number of pixels in the scanline (0=320, 1=640). For standard graphics work in 320 mode, the SCB should be set to zero.

Following the SCBs in memory (\$9E00-\$9FFF) are the color tables. Each table is 32 (\$20) bytes long. Each color is represented by 2 bytes, which are stored in reverse order, typical for assembly language storage of 16-bit values. Bits 0-3 contain the blue intensity, 4-7 contain green, and 8-11 contains red. All other bits are set to zero.

Now that we know how data is stored, how do we display it? Location \$C029 (banks \$00, \$01, \$E0, and \$E1) controls the display attributes of the Super Hi-Res graphics. Bit 7 is set to enable Super Hi-Res and bit 6 is set to allow linear mapping. It is also important to note that \$C029 overrides all other video modes. Thus, when Super Hi-Res is turned off, the computer will return to the state specified by the other soft switches, such as text, Lo-Res, or Hi-Res.

REFERENCES

- Adams, L. *High-Speed Animation and Simulation for Microcomputers*. Pennsylvania: Tab Books Inc., 1987.
- Bishop, B. "Modular Assembly Language Part V: The Apple IIGS." *Call - A.P.P.L.E.*, May, 1987, pp. 28-36.
- Eyes, D., and R. Lichty. *Programming the 65816 Including the 6502, 65C02 and 65802*. New York: Prentice-Hall Press, 1986.
- Fischer, M. *65816/65802 Assembly Language Programming*. California: Osborne McGraw-Hill, Inc., 1986.
- Fischer, M. *Apple IIGS Technical Reference*. California: Osborne McGraw-Hill Inc., 1986.
- Lichty, R., and T. Crosley. "Making the IIGS Sales Demo, Part II." *A+*, July, 1987, pp. 69-72.
- Meuse, S. "AmperDHR." *Nibble*, March, 1985, pp. 58-70.
- Peterson, D. C. "Universal Applesoft." *Nibble*, September, 1986, pp. 15-55.
- Rogers, D. F. *Procedural Elements for Computer Graphics*. New York: McGraw-Hill Book Company, 1985.
- Van Aken, J., and M. Novak. "Curve-Drawing Algorithms for Raster Displays." *ACM Transactions on Graphics* 4, 2, April, 1985, pp. 147-169.
- Ward, J. B. "Turtle Basic." *Nibble*, January, 1986, pp. 14-39.

LISTING 1: SUPERGR

```

1 .....
2 -
3 - SUPERGR S
4 - Super Hi-res Amperand Package
5 - Written by Tim Meekins
6 - Copyright (C) 1988
7 - MicroSPARC, Inc
8 - Concord, MA 01742
9 - Merlin Pro assembler V2.58
10 -
11 .....
12 -
13 - SYNTAX:
14 - HGR
15 - CLEAR <col>
16 - TEXT
17 - DEF <col>,<cr>,<e>,<ob>
18 - PLOT [to] <x>,<y> [to <x>,<y>]
19 - HCOLOR= <col>
20 - RECALL
21 - F(<x>,<y>),<col>
22 - C(<x>,<y>),<rad>
23 - DRAW
24 - XDRAW
25 - V(<x>,<y>),(<x>,<y>)
26 - HEAD(<x>,<y>),<col>
27 - R(<x>,<y>),(<x>,<y>)
28 - P <tbl>
29 -
30 - KC ;Turn on 65C02 opcodes
31 - KC ;Turn on 65C802 opcodes
32 - ORG $E000
33 -
34 LITNUM EQU $50
35 LASTVAR EQU $81
36 VARPNT EQU $82
37 CHRGET EQU $81
38 CHRGET EQU $87
39 AMPER EQU $1F5
40 TYPERR EQU $3076
41 PRMVL EQU $307B
42 CHMCLS EQU $0E8B
43 CHMDFN EQU $0E8E
44 CHMCOF EQU $0E8F
45 SYNERR EQU $0E99
46 PTRGET EQU $0F13
47 ILLERR EQU $E199
48 GETBYT EQU $E4F8
49 GETADR EQU $E752
50 -
51 - Global Variables
52 - Replaces Applesoft's Hi-Res loc's so be careful using Hi-res.
53 -
54 PLOTEX EQU $E0
55 PLOTFY EQU $E3
56 VWLEFT EQU $E5
57 VWRIGHT EQU $E7
58 VWTOP EQU $E9
59 VWBOT EQU $EB
60 -
61 - MACROS
62 -
63 NEG MAC
64 SEC SEC
65 LDA #E
66 SBC #J
67 <<<
68 -
69 - Start of program. Set up & vector
70 -
71 INIT LDA #FAC
72 AMPER AMPER
73 CLC
74 XCF #0
75 REP #0
76 LDA #START
77 STA AMPER1
78 SEC
79 XCF #E
80 RTS
81 -
82 - Interpret & command
83 -
84 START LDX #J4
85 ; CMP #TOKENS_X
86 ; BEQ FOUND
87 BEQ
88 BPL ;
89 JMP SYNERR
90 ; TOKENS DF 145,139,137,184,147,145,147
91 ; DF # ' ',148,149, ' ',135, ' ', ' '
92 FOUND JSR CHRGET
93 TAX
94 ASL
95 TAX

```

```

96 JMP (ADR5,X) 209
97 ADRS DN NGR 210
98 DN CLEAR 211
99 DN TEXT 212 - Text
100 DEF 213
101 DN WRLOT 214
102 DN HCOLOR 215 TEXT
103 DN RECALL 216 STA SCB29
104 DN FILL 217 RTS
105 DN CIRCLE 218
106 DN DRAW 219 - Recall
107 DN KDRAW 220
108 DN VIEWPOINT 221 RECALL LDA #RC1
109 DN READ 222 STA SCB29
110 DN RECT 223 RTS
111 224
112 - Draw - allow shadowing
113 - Hgr 226
114 - 227 DRAW LDA SCB35
115 ME 11 228 AND #11100111
116 CLC 229 STA SCB35
117 XCE 230 PWD
118 PWD 231 CLC
119 LDA #RC1 232 XCE
120 STA SCB29 233 REP 130
121 LDA SCB35 234 LDX #S2000
122 AND #11100111 235 LDY #S2000
123 STA SCB35 236 LDA #S7FFF
124 REP 537 237 SEC 1,1
125 LDA #0 238
126 HEX #F002001 sta 5012000 239 XCE
127 LDX #S2000 240 PLB
128 LDY #S2001 241 RTS
129 LDA #S7FFF 242
130 MWN 1,1 243 - XDraw - Disable shadowing
131 PLB 244
132 LDA #319 245 KDRAW LDA SCB35
133 STA VMLGHT 246 ORA #S0001000
134 STA VMLFT 247 STA SCB35
135 LDA #199 248 RTS
136 STA VMBOT 249
137 STA VMTOP 250 - Define a new palette color
138 STA PAL 251
139 SEP 530 252 ME 11
140 LDX #X1F 253 DEF
141 LDA PAL,130,X 254 JSR EVAL16
142 HEX #F009C01 STA 5019C00,X 255 JSR DEFCOL
143 HEX #F209C01 STA 5019E20,X 256 STA RED
144 HEX #F409C01 STA 5019E40,X 257 JSR CEVAL16
145 HEX #F609C01 STA 5019E60,X 258 STA GREEN
146 HEX #F809C01 STA 5019E80,X 259 JSR CEVAL16
147 HEX #FA09C01 STA 5019EA0,X 260 STA BLUE
148 HEX #FC09C01 STA 5019EC0,X 261 CLC
149 HEX #FE09C01 STA 5019EE0,X 262 XCE
150 HEX #F009F01 STA 5019F00,X 263 LDA DEFCOL
151 HEX #F209F01 STA 5019F20,X 264 ASL
152 HEX #F409F01 STA 5019F40,X 265 STA DEFCOL
153 HEX #F609F01 STA 5019F60,X 266 STA DEFCOL-1
154 HEX #F809F01 STA 5019F80,X 267 REP 530
155 HEX #FA09F01 STA 5019FA0,X 268 LDA PAL
156 HEX #FC09F01 STA 5019FC0,X 269 ASL
157 HEX #FE09F01 STA 5019FE0,X 270 ORA
158 CEX 271 ASL
159 BPL : 272 ASL
160 SEC 273 ASL DEFCOL
161 XCE 274 XCE
162 RTS 275 TAX
276 SEP 520
277 LDA GREEN
163 PAL320 DW 5000 Black Gray
164 DW 5777 Dark Gray
165 DW 5841 Brown
166 DW 572C Purple
167 DW 500F Blue
168 DW 5080 dark green
169 DW 5F70 Orange
170 DW 5000 Red
171 DW 5F4F Fresh Yellow
172 DW 5F4F Yellow
173 DW 5E09 Green
174 DW 54DF Light Blue
175 DW 5DAF Lilac
176 DW 57BF Periwinkle Blue
177 DW 5CCC Gray
178 DW 5FFF White
179 DW 5FFF White
180 - Clear screen to passed color
181 - 293 VIEWPORT CLC
182 - 294 XCE
183 NX 11 295 JSR PARCOORD
184 CLEAR JSR EVAL16 296 REP 520
185 STX CIRCUL 297 LDA PLOTX
186 TXA 298 CMP #S200
187 ASL 299 BCS :VIEWERR
188 ASL 300 STA VMLFT
189 ASL 301 LDA PLOTY
190 ASL 302 CMP #S200
191 ORA CLRCOL 303 BCS :VIEWERR
192 HEX #F002001 sta 5012000 304 STA VMTOP
193 CLC 305 SEP 520
194 XCE 306 JSR CHKOWM
195 PWD 307 JSR PARCOORD
196 REP 530 308 REP 130
197 LDX #S2000 309 LDA PLOTX
198 LDY #S2001 310 CMP VMLFT
199 LDA #17CPE 311 BCC :VIEWERR
200 MWN 1,1 312 CMP #S200
201 PLB 313 BCS :VIEWERR
202 SEC 314 STA VMLGHT
203 XCE 315 LDA PLOTY
204 LDA #199 316 CMP VMTOP
205 LDA PAL 317 BCC :VIEWERR
206 : HEX #F009D01 STA 5019000,X 318
207 DF1 319 BCS :VIEWERR
208 BPL : 320 STA VMBOT
321 SEC
322 XCE
323 RTS
324 :VIEWERR SEC
325 XCE
326 JMP ILLERR
327 +
328 + Read screen color
329 +
330 READ CLC
331 XCE
332 JSR PARCOORD
333 JSR CHKOWM
334 JSR PYRGET
335 BIT LASTVAR
336 BMI :
337 :VIEWERR
338 1 JSR #30
339 1 JSR SCREEN
340 SEP #30
341 LDY 141
342 STA (VARPNT),Y
343 DEY
344 TYA
345 STA (VARPNT),Y
346 SEC
347 XCE
348 RTS
349 -
350 - Rectangle
351 - 352 RECT CLC
353 XCE
354 JSR PARCOORD
355 REP 530
356 LDA PLOTX
357 STA RY1
358 LDA PLOTY
359 STA RY2
360 SEP 530
361 JSR CHKOWM
362 JSR PARCOORD
363 REP 530
364 LDA PLOTX
365 STA RX2
366 LDA PLOTY
367 STA RX1
368 STA Y1
369 LDA RX1
370 STA X1
371 STA Y2
372 LDA RX1
373 STA PLOTX
374 LDA RY1
375 STA Y1
376 STA PLOTY
377 LDA RX2
378 STA X1
379 JSR LINE
380 LDA RX1
381 STA X1
382 STA PLOTX
383 LDA RY1
384 STA PLOTY
385 LDA RY2
386 STA Y1
387 JSR LINE
388 LDA RX2
389 STA PLOTX
390 STA X1
391 LDA RY3
392 STA PLOTY
393 LDA RY2
394 STA Y1
395 JSR LINE
396 SEC
397 XCE
398 RTS
399 - Hplot
400 - 401 -
402 HPLLOT CLC
403 XCE
404 JSR CHKOWM
405 CMP #193 :to
406 BCC :TOLOOP8
407 JSR GETCOORD
408 JSR PLOT
409 SEP 530
410 :TOLOOP JSR CHKOWM
411 :TOLOOP JSR CHKOWM #193 :to
412 CMP #193
413 BNE QUIT
414 :TOLOOP JSR CHKOWM
415 STA X1
416 LDA PLOTX
417 STA X1
418 LDA PLOTY
419 STA Y1
420 JSR GETCOORD
421 JSR LINE
422 LDA OIDX
423 STA PLOTX
424 LDA OLYD
425 STA PLOTY
426 SEP 530
427 BRA :TOLOOP
428 QUIT SEC
429 XCE
430 RTS
431 -
432 - Evaluate AppleSoft type expression
433 -
434 CEVAL SEP 530

```

```

431 JSR CHKCOM
436 EVAL SEC
437 XCE
438 JSR FRMEVL
439 JSR GETADR
440 CLC
441 XCE
442 REP #30
443 LDA LINNUM
444 RTS
445 +
446 * Evaluate and make sure it's < 16.
447 +
448 MX #1
449 CEVAL16 JSR CHKCOM
450 EVAL16 JSR GETBYT
451 CPX #16
452 BGE EVALERR
453 RTS
454 EVALERR JMP ILLERR
455 +
456 * Get coordinate from TXPTX
457 +
458 GETCOORD JSR EVAL
459 STA PLOTX
460 STA OLDX
461 JSR CEVAL
462 STA PLOTY
463 STA OLDY
464 RTS
465 +
466 * Get coordinate between parenthesis
467 +
468 PARCOORD JSR CHKOPN
469 JSR GETCOORD
470 SEP #10
471 JMP CHKCLS
472 +
473 * Draw a line from plots.ploty to xl,y1
474 +
475 MX #0
476 LINE LDA X1
477 SEC
478 SBC PLOTX
479 BEQ #1
480 BVS :INV
481 BMI :SNAP
482 BRA #1
483 :INV BMI #1
484 :SNAP LDX X1
485 LDY PLOTX
486 STY X1
487 STX PLOTX
488 LDX Y1
489 LDY PLOTY
490 STY Y1
491 STX PLOTY
492 #1 SEC
493 LDA X1
494 SBC PLOTX
495 STA DX
496 SEC
497 LDA Y1
498 SBC PLOTY
499 STA DY
500 BMI #2
501 CMP DX
502 BCC :OCT10
503 BEQ :OCT10
504 BRA :OCT20
505 #2 NEG DY
506 CMP DX
507 BCC #7
508 BEQ #7
509 JMP :OCT70
510 #7 JMP :OCT80
511
512 :OCT10 LDX DX
513 STX CNT
514 LDA DX
515 BMI #3
516 LSR
517 STA ERR
518 NEG ERR
519 STA ERR
520 BRA :OCT11
521 #3 NEG DX
522 LSR
523 STA ERR
524 JSR PLOT
525 DEC CNT
526 BPL :OCT12
527 RTS
528 :OCT12 INC PLOTX
529 CLC
530 LDA ERR
531 ADC DY
532 STA ERR
533 BMI :OCT11
534 INC PLOTY
535 SEC
536 LDA ERR
537 SBC DX
538 STA ERR
539 BRA :OCT11
540
541 :OCT20 LDX DY
542 STX CNT
543 LDA DY
544 BMI #4
545 LSR
546 STA ERR
547 NEG ERR
548 STA ERR
549 BRA :OCT21
550
551 #4 LSR
552 STA ERR
553 BPL :OCT22
554 CLC
555 LDA ERR
556 ADC DX
557 STA ERR
558 BRA :OCT21
559
560 #5 NEG DY
561 STX CNT
562 LDA DY
563 BMI #5
564 LSR
565 STA ERR
566 ADC DX
567 STA ERR
568 BRA :OCT21
569
570 #6 NEG DX
571 STX CNT
572 LDA DX
573 BMI #5
574 LSR
575 STA ERR
576 ADC DY
577 BMI :OCT71
578 NEG DY
579 LSR
580 STA ERR
581 NEG ERR
582 :OCT71 JSR PLOT
583 DEC CNT
584 BPL :OCT72
585 RTS
586 :OCT72 PLOTY
587 CLC
588 LDA ERR
589 ADC DX
590 STA ERR
591 BMI :OCT71
592 INC PLOTX
593 CLC
594 LDA ERR
595 ADC DY
596 STA ERR
597 BRA :OCT71
598
599 :OCT80 LDX DX
600 STX CNT
601 LDA DX
602 BMI #6
603 LSR
604 STA ERR
605 NEG ERR
606 STA ERR
607 BRA :OCT81
608 #6 NEG DX
609 LSR
610 STA ERR
611 :OCT81 JSR PLOT
612 DEC CNT
613 BPL :OCT82
614 STA ERR
615 :OCT82 INC PLOTX
616 SEC
617 LDA ERR
618 SBC DY
619 STA ERR
620 BMI :OCT81
621 DEC PLOTY
622 SEC
623 LDA ERR
624 SBC DX
625 STA ERR
626 BRA :OCT81
627 +
628 * Hcolor
629 +
630 MX #1
631 HCOLOR JSR EVAL16
632 STX COLOR
633 RTS
634 +
635 * Palette
636 +
637 PALETTE JSR CEVAL16
638 STX PAL
639 RTS
640 +
641 * Fill
642 +
643 FILL CLC
644 XCE
645 JSR PARCOORD
646 JSR CEVAL16
647 STX BASECOL
648 STZ BASECOL+1
649 REP #30
650 LDX PLOTX
651 LDY PLOTY
652 CPX XRIGHT
653 BEQ #3
654 BCS :FQUIT
655 #3 PHX
656 LDA #2
657 STA STACK
658 #2 STA STACK
659 #2 DEC STACK
660 #1 LDA #1
661 FQUIT SEC
662 XCE
663 RTS
664 MI #0
665 #1 PLA
666 STA PLOTY
667 PLA
668 STA PLOTX
669 JSR
670 LDA PLOTX
671 STA SAVEX
672 #2 LOOP INC PLOTX
673 LDA PLOTX
674 XRIGHT
675 BEQ #1
676 STA SCREEN
677 #1 CMP BASECOL
678 BCC #2
679 JSR COLOR
680 CMP #2
681 BEQ #2
682 JSR PLOT2
683 BRA #2
684 #2 LOOP2 INC PLOTX
685 STA PLOTX
686 STA PLOTX
687 STA PLOTX
688 STA PLOTX
689 #2 LOOP DEC PLOTX
690 LDA PLOTX
691 CMP XRIGHT
692 BCC #3
693 STA SCREEN
694 #3 BEQ #3
695 JSR SCANLINE
696 #3 BEQ #3
697 #3 BEQ #3
698 #3 BEQ #3
699 #3 BEQ #3
700 #3 BEQ #3
701 #3 BEQ #3
702 #3 BEQ #3
703 #3 BEQ #3
704 #3 BEQ #3
705 #3 BEQ #3
706 #3 BEQ #3
707 #3 BEQ #3
708 #3 BEQ #3
709 #3 BEQ #3
710 #3 BEQ #3
711 #3 BEQ #3
712 #3 BEQ #3
713 #3 BEQ #3
714 #3 BEQ #3
715 #3 BEQ #3
716 #3 BEQ #3
717 #3 BEQ #3
718 #3 BEQ #3
719 #3 BEQ #3
720 #3 BEQ #3
721 #3 BEQ #3
722 #3 BEQ #3
723 #3 BEQ #3
724 #3 BEQ #3
725 #3 BEQ #3
726 #3 BEQ #3
727 #3 BEQ #3
728 #3 BEQ #3
729 #3 BEQ #3
730 #3 BEQ #3
731 #3 BEQ #3
732 #3 BEQ #3
733 #3 BEQ #3
734 #3 BEQ #3
735 #3 BEQ #3
736 #3 BEQ #3
737 #3 BEQ #3
738 #3 BEQ #3
739 #3 BEQ #3
740 #3 BEQ #3
741 #3 BEQ #3
742 #3 BEQ #3
743 #3 BEQ #3
744 #3 BEQ #3
745 #3 BEQ #3
746 #3 BEQ #3
747 #3 BEQ #3
748 #3 BEQ #3
749 #3 BEQ #3
750 #3 BEQ #3
751 #3 BEQ #3
752 #3 BEQ #3
753 #3 BEQ #3
754 #3 BEQ #3
755 #3 BEQ #3
756 #3 BEQ #3
757 #3 BEQ #3
758 #3 BEQ #3
759 #3 BEQ #3
760 #3 BEQ #3
761 #3 BEQ #3
762 #3 BEQ #3
763 #3 BEQ #3
764 #3 BEQ #3
765 #3 BEQ #3
766 #3 BEQ #3
767 #3 BEQ #3
768 #3 BEQ #3
769 #3 BEQ #3
770 #3 BEQ #3
771 #3 BEQ #3
772 #3 BEQ #3
773 #3 BEQ #3

```

```

774 SRC RADJUS
775 STA DELTA
776 ASL LOOP
777 INC LOOP
778 LOOP JSR PLOTXIRC
779 LDA DELTA
780 BHI 1
781 DEC CIRCX
782 LDA CIRCX
783 ASL
784 STA TEMP
785 SEC
786 LDA DELTA
787 SRC TEMP
788 STA DELTA
789 BPL
790 INC CIRCY
791 LDA CIRCY
792 ASL
793 CLC
794 ADC DELTA
795 STA DELTA
796 2 DEC LOOP
797 BPL
798 SEC
799 XCE
800 RTS
801
802 WX 00
803 PLOTXIRC JSR CALCCIRC
804 LDA Y0
805 STA PLOTY
806 LDA X0
807 STA PLOTX
808 JSR PLOT
809 LDA XZ
810 STA PLOTE
811 JSR PLOT
812 LDA YZ
813 STA PLOTY
814 JSR PLOT
815 LDA X0
816 STA PLOTA
817 JMP PLOT
818
819 CALCCIRC CLC
820 LDA CENTERX
821 DEC CIRCX
822 STA X0
823 SEC
824 LDA CENTERX
825 SRC CIRCX
826 STA XZ
827 CLC
828 LDA CENTERX
829 ADC CIRCY
830 STA Y0
831 SEC
832 LDA CENTERX
833 SRC CIRCX
834 STA YZ
835 RTS
836 -
837 - Plot x dot in 320 mode
838
839 WX 00
840 PLOT LDA PLOTA
841 CMP VWRIGHT
842 BEQ PC1
843 BCS PEK1T
844 PC1 CMP VLEFT
845 BHI PEK1T
846 BCC PEK1T
847 LDA PLOTY
848 CMP VNBOT
849 BEQ PC2
850 BCS PEK1T
851 PC2 CMP VTOP
852 BCC PEK1T
853 LDA PLOTY-1
854 LSR
855 LDA
856 ADC PLOTY-1
857 BEQ ADOP
858 LSR
859 TAX
860 SEP $20
861 BCS P00D
862 PEVEN LDA COLOR
863 ASL
864 ASL
865 ASL
866 STA PCOL
867 AND BF002001 :lda $012000,x
868
869 ORA PCOL
870 LDX BF002001 :lda $012000,x
871 LDI PLOTY
872 LDA
873 PAL
874 BEZ BF009001 :sta $019000,x
875 REP $30
876 RTS
877
878 P00D HEX BF002001 :lda $012000,x
879 AND #F8
880 ORA COLOR
881 AND BF002001 :lda $012000,x
882 LDI PLOTY
883 LDA
884 HEX BF009001 :sta $019000,x
885 PEK1T $30
886 NIS
887 PLOTF SEP $20
888 STY :GOPLDT,1

```

```

889 GOPLDT JMP PEVEN
890 +
891 + Get color at plots, ploty
892 +
893 SCREEN LDA PLOTY-1
894 LSR
895 LSR
896 ADC PLOTY-1
897 ADC PLOTY
898 LSR
899 TAX
900 BCS
901 WX 00
902 EVEN LDA #0
903 SEP $20
904 HEX BF002001 :lda $012000,x
905 LSR
906 LSR
907 LSR
908 LSR
909 REP $30
910 LDI #PEVEN
911 RTS
912 WX 00
913 ODD LDA #0
914 SEP $20
915 HEX BF002001 :lda $012000,x
916 AND #0F
917 REP $30
918 LDI #PC0D
919 RTS
920 +
921 + Global variables
922 +
923 COLOR DS 1
924 DF HEX 00
925 PCOL DS 2
926 PAL DS 2
927 +
928 + Local Variables
929 +
930 LOCAL EQU +
931
932 CLRCOL EQU LOCAL
933
934 DEFCOL EQU LOCAL
935 RED EQU DEFCOL+2
936 BLE EQU RED+1
937 GREEN EQU BLUE+1
938
939 DK EQU LOCAL
940 DF EQU DK+2
941 CNT EQU CNT+2
942 X1 EQU CNT+2
943 Y1 EQU X1+2
944 OLR EQU Y1+2
945 OLF EQU OLR+2
946 ERR EQU OLF+2
947 RX1 EQU ERR+2
948 RX2 EQU RX1+2
949 RY1 EQU RX2+2
950 RY2 EQU RY1+2
951
952 XRIGHT EQU LOCAL
953 XLEFT EQU XRIGHT-2
954 SAVEX EQU XLEFT-2

```

```

955 SAVEX EQU SAVEX+2
956 BASECOL EQU SAVEY+2
957 ADDR EQU BASECOL+2
958 STACK EQU ADDR+2
959
960 X0 EQU LOCAL
961 Y0 EQU X0+2
962 TEMP EQU Y0+2
963 CENTERX EQU TEMP+2
964 CENTERY EQU CENTERX+2
965 RADJUS EQU CENTERY+2
966 CIRCX EQU RADJUS+2
967 CIRCY EQU CIRCX+2
968 DELTA EQU CIRCY+2
969 LOOP EQU DELTA+2
970 XZ EQU X0+2
971 YZ EQU XZ+2
972 +
973 FIN EQU +

```

END OF LISTING 1

KEY EXECUTION 5.0
RUN ON SUPERGR

CODE-5.0	ADDR#	ADDR#	CODE-4.0
3A993AAE	8E0E	BE4F	2870
C9940353	8E36	BE9F	2548
C060CAAF	8E46	BEFF	2781
90431315	8EFD	BF3F	2627
00A033C0	8F40	BF8F	2675
E996B8EC	8F90	BFDF	287A
68E9BEF8	8FE0	902F	265B
03919436	9030	907F	263A
ASAFECFA	9080	90CF	28F2
3C090399	9090	911F	27C8
B662AD7A	9120	916F	2626
S190BE2F	9170	91BF	256A
A60345E7	9210	925F	2812
5D705EC6	9260	92AF	2A95
SDBB4816	92B0	92FF	2659
18A420A0	9300	934F	2705
D25F0B11	9350	939F	290F
CODE493B	93A0	93EF	2A8B
FCBC47BF	93F0	943F	262A
5431A0B9	9440	948F	2358
3E99BF19	9490	94DF	28F9
ACBC390E	= PROGRAM TOTAL =		0660

LISTING 2: SUPERGR

START: 8E00 LENGTH: 6E0

```

BE BE00:A9 4C 8D F5 03 18 FB C2
7F BE08:20 A9 12 BE 8D F6 03 38
95 BE10:F6 60 A2 0E DD 1F BE F0
D3 BE18:15 CA 10 8D 4C C9 DE 91
2R BE20:8D 89 BE 93 92 A7 46 43
00 BE28:94 95 56 87 52 50 20 81
F5 BE30:00 8A 0A 7A C7 3E 55
80 BE38:BE F9 8E 2C 8F 5E 8F 8A
05 BE40:90 88 92 32 8F 99 92 8D
84 BE48:93 38 8F 55 8F AB 8F F0
62 BE50:8F 14 90 92 92 18 FB 8B
50 BE58:A9 C1 8D 29 C0 AD 35 C0
2R BE60:29 E7 8D 35 C0 C2 30 A9
8B BE68:00 80 8F 00 20 01 A2 00
43 BE70:20 A0 01 20 A9 FE 7F 34
8A BE78:01 01 AB A9 3F 01 85 E7
BE BE80:64 E5 A9 C7 00 85 E6 64
95 BE88:99 9C DE 94 E2 30 A2 1F
F6 BE90:BD D0 8E 9F 00 9E 01 9F
BF BE98:20 9E 01 9F 40 9E 01 9F
3F BEA0:60 9E 01 9F 80 9E 01 9F
4C BEA8:0A 9E 01 9F C0 9E 01 9F
DA BEB0:F0 9E 01 9F 00 9E 01 9F
02 BEC0:60 9F 01 9F 80 9F 01 9F
EB BECS:A0 9F 01 9F C0 9F 01 9F
59 BED0:E0 9F 01 CA 10 8A 38 F8
63 BED8:60 00 00 77 07 41 08 2C

```

```

68 BEE0:07 0F 00 80 00 70 0F 00
58 BEE8:0D A9 0F 0F 0F E0 00 DF
3E BEF0:04 0F 0D 8F 07 CC 0F FF
0E BEF8:0F 20 DF 90 BE E0 94 8A
CA BF00:0A 0A 0A 0A 00 00 94 8F
88 BF08:00 20 01 18 FB 8R C2 30
2D BF10:A2 00 20 A0 20 A9 FE
CF BF18:7C 54 01 01 AB 38 FA A2
E3 BF20:C7 AD DE 94 9F 00 90 01
CE BF28:CA 10 F9 60 A9 01 8D 29
7A BF30:C0 60 A9 C1 8D 29 C0 60
67 BF38:AD 35 C0 29 0F 87 8D 35 C0
33 BF40:8B 18 FB C2 30 A2 00 20
8C BF48:AD 00 20 A9 FF 7F 54 01
43 BF50:01 38 FB AB 60 AD 35 C0
F8 BF58:09 18 8D 35 C0 60 20 DF
96 BF60:90 BE 8A 20 0C 8E 94 8E
05 BF68:E2 94 20 DC 00 8E E4 94
C9 BF70:2D C0 DC 90 BE E3 94 18 FB
C9 BF78:AD E0 9A 0A 80 E0 94 9C
3F BF80:E1 94 C2 30 AD DE 94 8A
E4 BF88:0A 0A 0A 0A 00 00 94 8A
74 BF90:E2 20 AD E4 0A 0A 0A 0A
D8 BF98:0A E0 E3 94 9F 00 9E 01
A6 BFA0:E8 AD E2 94 9F 00 9E 01
D7 BFA8:38 FB 60 18 FB 20 FB 90
7R BFB0:C2 20 A5 E0 C9 40 01 80
E1 BFB8:32 85 E5 A5 E3 C9 C8 00
69 BFC0:BD 29 85 E5 E2 20 20 BE
04 BFC8:DE 20 FB 90 C2 30 A5 E0

```

```

CA 8FD0 C5 E5 90 17 C9 40 01 B0
DE 8FD8 12 85 E7 A5 E3 C5 E9 90
43 8FE0 0A C9 C8 00 FB 4C 85 EB
07 8FE8 38 FB 60 38 FB 4C 99 E1
80 8FF0 18 FB 20 FB 90 20 BE DE
C1 8FF8 20 E3 DF 24 81 30 03 AC
51 9000 76 DD C2 30 20 AA 94 E2
0E 9008 30 A0 01 91 83 88 98 91
40 9010 83 38 FB 60 18 FB 20 FB
A1 9018 90 C2 30 A5 E0 80 F0 94
18 9020 A5 E3 8D F4 94 E2 30 20
49 9028 BE DE 20 F8 90 C2 30 A5
55 9030 E0 8D F2 94 A5 E3 8D F6
28 9038 94 8D E8 94 AD F0 94 8D
9C 9040 E6 94 20 06 91 AD F0 94
92 9048 85 E0 AD F4 94 8D E8 94
08 9050 85 E3 AD F2 94 8D E6 94
09 9058 20 06 91 AD F0 94 8D E6
FE 9060 94 85 E0 AD F4 94 85 E3
79 9068 AD F6 94 8D E8 94 20 06
33 9070 91 AD F2 94 85 E0 8D E6
F6 9078 94 AD F4 94 85 E3 AD F6
F4 9080 94 8D E8 94 20 06 91 AD
DE 9088 FB 18 FB 20 07 80 C9
58 9090 C1 F0 0F 20 EA 90 20 3E
43 9098 94 E2 30 20 E7 80 C9 C1
47 90A0 D0 23 20 B1 60 C2 30 A5
3A 90A8 E0 8D E6 94 A5 E3 8D E8
AA 90B0 94 20 EA 90 20 06 91 AD
AE 90B8 EA 94 85 E0 AD EC 94 85
00 90C0 E3 E2 30 8D 06 38 FB 60
AD 90C8 F2 30 20 BF DE 38 FB 20
91 90D0 7B DD 20 52 E7 18 FB C2
93 90D8 30 A5 50 60 20 8E DE 20
D8 90E0 FB E6 E0 10 80 01 4C AC
BE 90E8 99 E1 20 CD 90 85 E0 8D
CA 90F0 EA 94 20 C8 90 85 E3 8D
58 90F8 EC 94 60 20 88 DE 20 EA
92 9100 90 F2 30 4C 8B DE AD F6
5A 9108 94 38 E5 E0 F0 1C 70 94
52 9110 30 04 80 16 30 14 AE E5
72 9118 94 A4 E0 8C E6 94 86 E0
C4 9120 AE E8 94 A4 E3 8C E8 94
98 9128 8E E3 38 AD E6 94 E5 E0
2C 9130 8D E0 94 38 AD E8 94 E5
94 9138 E3 8D F2 94 30 09 CD E0
38 9140 94 90 18 F0 16 80 5F 38
87 9148 A9 00 00 ED E2 94 CD E0
88 9150 94 90 05 F0 03 4C F1 91
46 9158 AC 40 92 AE E0 94 8E 4A
50 9160 94 AD E0 94 30 10 4A 8D
32 9168 EE 94 38 A9 00 00 ED EE
27 9170 94 8D EE 94 80 08 38 AD
91 9178 00 00 ED E0 94 4A 8E
56 9180 94 20 3E 94 CE EA 94 10
02 9188 01 60 E6 E0 18 AD EE 94
05 9190 6D E2 94 8D EE 94 30 E9
37 9198 E6 E3 38 AD EE 94 ED E0
14 91A8 94 8D EE 94 8D DE AE E2
F1 91B8 94 8E E4 94 AD E2 94 30
E7 91C0 10 4A 8D EE 94 3E A9 00
4C 91C8 00 ED EE 94 8D EE 94 8D
BC 91D0 38 38 A9 00 00 ED EE 94
FB 91D8 4A 8D EE 94 20 3E 94 CE
A1 91E0 EA 94 10 01 60 EE E3 8D
78 91E8 AD EE 94 60 E0 94 8D EE
59 91F8 94 E0 E2 94 8D EE 94 8D
55 91F0 DB 38 A9 00 00 ED F2 94
85 91F8 8D EA 94 AD E2 94 30 06
E8 9200 4A 8D EE 94 8D 15 38 A9
CC 9208 00 00 ED E2 94 4A 8D EE
22 9210 94 38 A9 00 00 ED EE 94
47 9218 8D EE 94 20 3E 94 CE EA
CA 9220 94 10 01 60 C6 E3 18 AD
08 9228 EE 94 6D E0 94 8D FF 94
A3 9230 30 E9 EG E0 18 AD EE 94
52 9238 6D E2 94 8D EE 94 8D DB
09 9240 AE E0 94 8E EA 94 AD E0
43 9248 94 30 10 4A 8D EE 94 38
05 9250 A9 00 00 ED EE 94 8D EE
3E 9258 94 80 08 38 A9 00 00 ED
88 9260 E0 94 4A 8D EE 94 20 3E
03 9268 EC E4 94 10 01 60 E6
06 9270 E0 38 AD EE 94 FD F2 94
06 9278 8D EE 94 30 E9 C6 E3 38
FE 9280 AD EE 94 ED E0 94 8D EE
87 9288 94 80 DB 20 DF 90 8E DA
90 9290 94 60 20 DC 90 8E DE 94
73 9298 60 18 FB 20 FB 90 20 DC
C6 92A0 90 8E E8 94 9C E9 94 C2
70 92A8 30 A6 E0 A4 E3 E4 E7 80
86 92B0 02 80 DA 5A 93 02 80
87 92B8 8D EC 94 CE EC 94 8D 83
40 92C0 38 FB 60 68 85 E3 68 85
30 92C8 E0 20 3E 94 A0 8D 8D EA
CE 92D0 94 E5 E0 A5 E0 C5 E7 F0
FE 92D8 02 80 12 20 AA 94 CD E8
88 92E0 94 F0 0A CD DA 94 F0 05
55 92E8 20 A2 94 8D EA A6 E0 CA
50 92F0 8E F0 94 AD E4 94 85 E0
71 92F8 C6 E0 A5 E0 E5 C5 30 16
31 9300 90 12 20 AA 94 CD E8 94
9A 9308 F0 0C DA DA 94 F0 07 20
97 9310 A2 94 8D E4 80 A5 A6 E0
1E 9318 E8 BE E2 94 A6 E3 BE E6
A5 9320 94 E8 86 E3 EA EB F0 02
31 9328 80 EA 20 3F 93 AE E6 94
68 9330 CA 86 E3 30 DF EA E9 90
C8 9338 20 20 3F 93 AC BB 92 68
BE 9340 8D EA 94 AD E2 94 85 E0
CF 9348 20 AA 94 CD E8 94 F0 05
3C 9350 CD DA 94 08 00 E6 E0 A5
65 9358 E0 CD EA 94 90 EA F0 E8
85 9360 80 26 20 00 94 CD E8 94
CF 9368 F0 BE CD DA 94 F0 09 E6
47 9370 E8 A5 E0 CD E0 94 90 E8
23 9378 A5 E0 CA DA A5 E3 48 E2
30 9380 EC 94 E8 EC E0 94 90 CD
68 9388 AD EA 94 48 60 20 FB 90
26 9390 20 C8 90 8D EA 94 8D EC
78 9398 94 8D F2 94 A5 E0 8D E6
13 93A0 94 A5 E3 8D E8 94 9C EE
20 93A8 94 38 A9 01 08 ED EA 94
47 93B0 8D F0 94 E0 F2 94 EE F2
03 93B8 94 20 ED 93 AD F0 94 30
01 93C0 16 EC EC 94 AD EC 94 0A
A6 93C8 8D E4 94 38 AD F0 94 ED
A2 93D0 EA 94 8D F0 94 80 EE FE
AC 93D8 EE 94 AD EE 94 0A 18 6D
16 93E0 FB 94 8D F0 94 C6 F2 94
85 93E8 10 CF 38 FB 60 20 15 94
8A 93F0 AD E2 94 85 E3 AD E0 94
18 93F8 85 E0 20 3E 94 AD F4 94
84 9400 85 E0 20 3E 94 AD F6 94
23 9408 85 E3 20 3E 94 AD E0 94
C8 9410 85 E0 4C 3E 94 18 AD E6
F2 9418 94 60 EC 94 8D E0 94 38
29 9420 AD E6 94 ED EC 94 8D F4
60 9428 94 18 AD E8 94 6D EE 94
3A 9430 8D E2 94 38 AD E8 94 ED
A6 9438 EE 94 8D F6 94 60 A5 E0
27 9440 C5 E7 F0 02 80 59 C5 E5
7A 9448 30 20 90 53 A5 E3 C5 EB
EB 9450 F0 02 80 48 C5 E9 80 47
40 9458 A5 E2 4A 4A 65 E2 65 E0
9C 9460 AA AA E2 20 80 23 AD DA
71 9468 94 0A 0A 8A 8D DC 94
87 9470 8F 00 00 21 29 8F 0D C8
34 9478 94 9F 00 20 01 A6 E3 AD
68 9480 DE 94 9F 00 9D 01 C2 30
A7 9488 60 8F 00 20 01 29 F0 8D
64 9490 DA 94 9F 00 00 01 A6 E3
10 9498 AD DE 94 9F 00 00 91 C2
E6 94A0 30 6D 92 20 8C A6 94 AC
DF 94A8 66 94 A5 E2 4A 4A 65 E2
6E 94B0 65 E0 4A AA 80 13 A9 E0
6D 94B8 00 E2 20 8F 00 20 01 4A
83 94C0 4A 4A AC C2 30 A0 66 94
8E 94C8 60 A9 00 00 E2 20 BF 00
E4 94D0 20 01 29 0F C2 30 A0 89
40 94D8 94 60 80 00 00 00 00 00
TOTAL: 5892
END OF LISTING 2

```

LISTING 3: SUPERGR.DEMO1

```

C7 10 REM *****
30 20 REM = SUPERGR.DEMO1 =
A9 30 REM = SUPERGRAPHICS GS =
AE 40 REM = BY TIM MEEKINS =
CB 50 REM = COPYRIGHT (C) 1988 =
24 60 REM = MICROSPARC, INC =
45 70 REM = CONCORD, MA 01742 =
3A 80 REM *****
06 90 PRINT CHR$(4) "BRUN SUPERGR"
58 100 ONERR GOTO 600
1D 110 HIMEM: 35328
96 120 & HGR : HOME : VTAB 11: HTAB 20: PRINT "P
lease wait..." : & TEXT
8F 130 FOR I = 1 TO 15: & DEF I,0,0,0
1C 140 & HCOLOR= I: & C(160,100),I - 12: & C(160
,100),(I + 1) * 12
00 150 X = I * 12 + 2: Y = 0: IF X > 159 THEN X = I
58: Y = 98 * (I = 15) + 80 * (I = 14)
08 160 & F(160 - X,100 - Y),15: & F(160 + X,100 -
Y),15: & F(160 + X,100 + Y),15: & F(160 -
X,100 + Y),15
4C 170 NEXT

```

```

DC 180 VTAB 11: HTAB 18: PRINT "Press any key to
start... GET Z$: & RECALL : & DRAW
68 190 FOR I = 0 TO 15: FOR J = 0 TO I: & DEF J,
I - J, I, J,0: NEXT: FOR K = 1 TO 100: NEXT
I : NEXT
7D 200 FOR I = 1 TO 15: FOR J = 0 TO 15: P = 15 -
ABS (I - J): & DEF J,P,0: NEXT: FOR K =
1 TO 100: NEXT: NEXT
F6 210 FOR I = 1 TO 15: FOR J = I TO 15: P = J - I
: & DEF J,P,0: NEXT: FOR K = 1 TO 100:
NEXT: NEXT
D8 220 FOR I = 0 TO 15: & DEF I,1,3,7,3: FOR J =
1 TO 200: NEXT: NEXT
47 230 & CLEAR 0
55 240 & HCOLOR= 1: & HPLLOT 90,160 TO 158,160 T
0 124,92 TO 90,160: & F(124,100),1
0F 250 & HPLLOT 158,24 TO 226,160 TO 162,160 TO 1
26,88 TO 158,24: & F(160,100),1
D5 260 & HCOLOR= 2: & HPLLOT 150,60 TO 190,140 T
0 110,140 TO 150,60: & F(160,100),2
36 270 & HPLLOT 166,60 TO 206,140 TO 194,140 TO 1
60,72 TO 166,60: & F(166,70),2
F4 280 & HCOLOR= 4: & HPLLOT 142,96 TO 154,120 T
0 130,120 TO 142,96

```

```

CC 290 & HPL0T 150,80 TO 170,120 TO 158,120 TO 1
    44,92 TO 150,80
26 300 & HPL0T 166,80 TO 186,120 TO 174,120 TO 1
    60,92 TO 166,80
F6 310 & HCOLOR= 3: & F(140,110),4: & F(150,100)
    ,4: & F(170,100),4
E7 320 FOR I = 0 TO 3: & DEF 1,15,14,15: NEXT:
    GOSUB 610: FOR J = 14 TO 8 STEP - 1: FOR
    I = 0 TO 3: & DEF 1,J,14,15: NEXT: GOSUB
    610: NEXT: FOR J = 14 TO 4 STEP - 1: FOR
    I = 0 TO 3: & DEF 1,8,14,J: NEXT: GOSUB 6
    10: NEXT
CA 330 FOR I = 13 TO 10 STEP - 1: FOR J = 0 TO 2
    : & DEF J,8,1,4: NEXT: GOSUB 610: NEXT:
    FOR I = 7 TO 1 STEP - 1: FOR J = 0 TO 2:
    & DEF J,I,10,4: NEXT: GOSUB 610: NEXT
33 340 & DEF 0,0,10,4: & DEF 1,0,10,4: GOSUB 61
    0: FOR I = 9 TO 6 STEP - 1: & DEF 0,0,I,4
    : & DEF 1,0,1,4: GOSUB 610: NEXT
C9 350 FOR I = 5 TO 3 STEP - 1: & DEF 0,0,I,4:
    GOSUB 610: NEXT
42 360 FOR I = 1 TO 2000: NEXT: & DEF 5,0,2,3
AB 370 & V(90,24),(226,160): & HCOLOR= 5: FOR I
    = 266 TO 0 STEP - 1: & C(160,100),I: NEXT
    & CLEAR 0
CF 380 & DEF 1,15,0,15: & HCOLOR= 1: & V(0,0),(
    319,100)
6D 390 R = 100: FOR THETA = 0 TO 6.283 STEP .1258:
    X = (100 - R) * COS (THETA) + 160: Y = (100
    - R) * SIN (THETA) + 100: & C(X,Y),R:R =
    R - 2: NEXT
35 400 & V(0,101),(319,199): FOR I = 100 TO 0 STE
    P - 4: & C(160,100),I: NEXT
C1 410 & DEF 2,3,3,15: & DEF 3,0,0,12
CF 420 & V(60,0),(260,199)
73 430 FOR I = 0 TO 200 STEP 4: & HCOLOR= 2:
    & HPL0T 0,100 TO 319,I: & HCOLOR= 3:
    & HPL0T 0,I TO 319,100: NEXT
A9 440 & HCOLOR= 0: FOR I = 1 TO 1000: NEXT
FD 450 FOR I = 100 TO 0 STEP - 1: & R(160 - I,10
    0 - I),(160 + I,100 + I): NEXT
4C 460 & CLEAR 0
0B 470 FOR X = 0 TO 15: FOR Y = 0 TO 15: & P,Y:
    & DEF X,X,Y,(X + Y) / 2: NEXT: NEXT
4F 480 P = 0: FOR Y = 0 TO 199
09 490 IF P > 15 THEN & P,31 - P: GOTO 510
60 500 & P,P
76 510 C = 0: FOR X = 0 TO 319
3F 520 IF C > 15 THEN & HCOLOR= 31 - C: GOTO 54
    0
D5 530 & HCOLOR= C
AA 540 & HPL0T X,Y
03 550 C = C + 1: IF C = 32 THEN C = 0
77 560 NEXT
00 570 P = P + 1: IF P = 32 THEN P = 0
FF 580 NEXT
D1 590 GET Z$
12 600 & TEXT: & HOME: PRINT "SuperGR Demo Termi
    nated.": END
D1 610 FOR Z = 1 TO 300: NEXT: RETURN

```

TOTAL: BC8E

END OF LISTING 3

KEY PERFECT 5.0
RUN ON
SUPERGR.DEMO1

```

=====
CODE-5.0  LINE# - LINE#  CODE-4.0
-----
8626B57C   10 - 100   5F7A
20238955   110 - 200   8401
8D4AD1EF   210 - 300   C82C
AD9F96A7   310 - 400   81A08
26062CFE   410 - 500   7968
50412E97   510 - 600   40C7
0548B74A   610 - 610   0786
4DFAB5DE = PROGRAM TOTAL = 0812

```

LISTING 4: SUPERGR.DEMO2

```

37 10 REM *****
C0 20 REM = SUPERGR.DEMO2 *
B9 30 REM = BY TIM MEEKINS *
AE 40 REM = COPYRIGHT (C) 1988 *
CB 50 REM = MICROSPARC, INC *
24 60 REM = CONCORD, MA 01742 *
45 70 REM *****
7C 80 PRINT CHR$(4);"BRUN SUPERGR": HIMEM= 3532
    8: 8HGR
25 90 FOR Y = 0 TO 15: FOR X = 0 TO 15: & P,Y:
    & DEF X,X,Y,0: & HCOLOR= X
9E 100 X1 = X + 20: X2 = X1 + 19: Y1 = 12 + Y: Y2 = Y
    1 + 11: & V(X1,Y1),(X2,Y2): & F(X1,Y1),15
5F 110 NEXT: NEXT
ED 120 & HCOLOR= 0
16 130 GET Z$
64 140 FOR I = 0 TO 15: & XDRAW
28 150 FOR X = 0 TO 15: FOR Y = 0 TO 15: & P,Y:
    & DEF X,X,Y,I: NEXT: & HPL0T 0,12 - X:
    NEXT
CF 160 & DRAW
4C 170 NEXT
C3 180 GET Z$: & TEXT

```

TOTAL: 64A9

END OF LISTING 4

KEY PERFECT 5.0
RUN ON
SUPERGR.DEMO2

```

=====
CODE-5.0  LINE# - LINE#  CODE-4.0
-----
23EBC994   10 - 100   8842
C5E22F9B   110 - 180   2F0F
9AABD003 = PROGRAM TOTAL = 0194

```


Supergraphics GS (Vol. 9/No. 2):
Sometimes SuperGR will erroneously
jump back to Applesoft while still in the
65816's native mode.

If you have an assembler, insert the
two lines

```
SEC  
XCE
```

before **line 337** in **Listing 1**, and change
line 454 to

```
:EVALERR SEC
```

Also, insert the lines

```
XCE  
JMP ILLERR
```

before **line 455**.

If you don't have an assembler,
BLOAD the SUPERGR file and enter
the following commands from the
monitor

```
8FFF:4C E8 94  
90E7:4C E5 94  
94E8:38 FB 4C 76 DD 38 FB 4C  
94E8:E1 99  
BSAVE SUPERGR,A$8E00,L$6EA
```