DEPARTMENT OF THE INTERIOR

U.S. GEOLOGICAL SURVEY

SPIKE: A Program to Plot Three-Dimensional Data

by

Carl A. Carlson[1]

Open-File Report 85-105

[1] Menlo Park, California

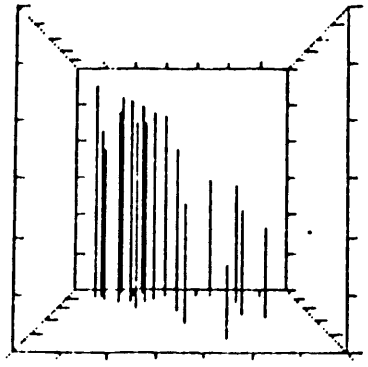
1984

Introduction

This report describes the computer program SPIKE which plots three dimensional data in perspective view for up to 150 data points. Figure 1 shows two kinds of three-dimensional plots generated by SPIKE. The top of each vertical line (or spike) marks the position of a data point in three-dimensional cartesian coordinates. The x-axis increases toward the viewer; the y-axis increases from left to right; and the z-axis increases from bottom to top. The program can rotate the data, plot a hard copy, plot a stereo-pair, store the data, and edit the data. SPIKE and its data requires a HP-85 with a total of 32 K of RAM. The program has the following features:

1) It runs on a HP-85 desk top computer.
2) The data is plotted on the system CRT. As the data is entered it is automatically scaled to fit.
3) Scaling information is automatically printed on the system printer.
4) The plot may be readily rotated up, down, clockwise, counterclockwise, to the left, or to the right in increments of 10 degrees.
5) The plot may be printed on the system printer.
6) A stereo-pair may be printed on the system printer.
7) The data may be readily edited.
8) The data and the plot may be readily saved or retrieved to or from the HP-85 system mass storage device.
9) A help screen can be printed on request

SPIKE permits a user to visualize and interpret spatial data such as geochemical surveys by displaying the data from multiple perspectives.



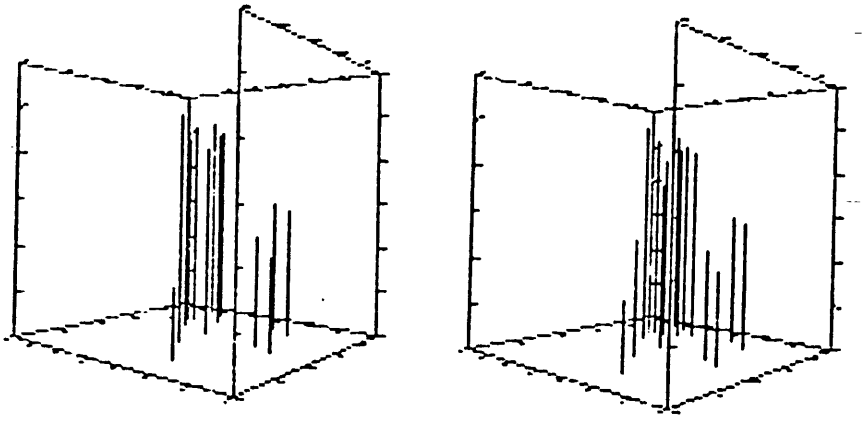


Figure 1. Examples of plots produced by SPIKE. The bottom pair of plots is a stereo-pair.

Program Methods

The following discussion describes some of the subroutines used by SPIKE. This information will be useful to programmers who want to revise the program or to use these subroutines in different programs. It is not necessary to read the following to use SPIKE.

SPIKE uses a package of three-dimensional plotting subroutines written in BASIC. These subroutines are general purpose and could easily be applied to other uses. The subroutines are based on the Hewlett Packard BASIC two-dimensional plotting commands:

```
PLOT X, Y
SCALE XMIN, XMAX, YMIN, YMAX
XAXIS Y-INTERCEPT, TIC, XMIN, XMAX
YAXIS X-INTERCEPT, TIC, YMIN, YMAX
etc.
```

Among others, there are subroutines analogous to:

```
PLOT X, Y, Z
SCALE XMIN, XMAX, YMIN, YMAX, ZMIN, ZMAX
XAXIS Y-INTERCEPT, Z-INTERCEPT, TIC, XMIN, XMAX
ROTATE X, Y, Z, ANGLE, AXIS NUMBER
MOVE X, Y, Z
etc.
```

The following conventions are used in the subroutines:

1) They reserve all variable names begining with the letters O through Z.
2) They reserve all line numbers between 1 and 40 and those greater than 7999.
3) A subroutine is called by first assigning a user's values to the appropriate subroutine variable names and then executing a GOSUB command to a subroutine line number in the 8000 to 8050 range.
4) Subroutine variable names can be: X, Y, Z, X1, X2, Y1, Y2, Z1, Z2, V1, V2, and V3.
5) For example: to initialize the plot volume; scale the plot volume to a cube 10 units on a side; move the "pen" to the point (5,5,5); plot a line to the point (7,7,7); rotate the cube 20 degrees about an x-axis passing through the point (2,2,2), and replot the line execute the following commands.

```
10 X1,Y1=0 @ X2,Z=255 @ Y2=191 @ GOSUB 8000 !Initialize
20 X1,Y1,Z1=0 @ X2,Y2,Z2=10 @ GOSUB 8001     ! Scale
30 X,Y,Z=5 @ GOSUB 8022                      ! Move to 5,5,5
40 X,Y,Z=7 @ GOSUB 8020                      ! Plot to 7,7,7
50 X,Y,Z=2 @ V1=20 @ @ V2=1 GOSUB 8003       ! Rotate 20
60 GOSUB 8750                                ! Replot
```

Using the Program

Follow these steps to use SPIKE:

1) Turn on the HP-85.
2) Insert the Data Cartridge containing SPIKE and type: LOAD "SPIKE". After SPIKE is loaded, you can change the cartridge if you want to store your data elsewhere.
3) Press [END LINE] and [RUN].
4) The HP-85 screen will clear and display a message prompting you to enter data.
5) Type H [END LINE] to get a help listing.
6) Enter the first data as a three number x,y,z triplet with asterisks seperating the numbers, e.g. 1.2*3.4*5.6 [END LINE]. This example would represent a point at the cartesian coordinate location (1.2, 3.4, 5.6).
7) Repeat step 6) for the next two points. SPIKE uses the first three points to select an initial scale for the plot. To avoid frequent rescaling and replotting, choose extreme points for the first three data points. Thereafter, the data will be plotted as it is entered.
8) To rotate the plot or execute any of the commands, type any combination of command letters in the place of a data triplet. For example RRUUUC would rotate the plot 20 degrees to the right, then rotate the plot 30 degrees upward, and then rotate the plot 10 degrees clockwise. The help listing (step 5 above) gives additional examples and explainations.
9) Enter data points in the same format as in step 6) or enter command sequences as in step 8). You can enter data lines or a command lines in any order you wish.

```
 10 DIM S(4,4),S1(3,6),S6(3,2),W
    (4,4),W1(3,3),W2(3),W3(4,4),
    W4(4,4),W5(2,3),X2$[300]
 20 SHORT K(3,300)
 30 INTEGER K1(300),T,T1,T2,S4(4
    ),S5(3),S2(3)
 40 DIM W6(3)
 50 REM 888
 99 REM  *** SPIKE VER 2.1 ***
100 GCLEAR @ CLEAR @ DISP "
      SPIKE DATA PLOTTER" @ DISP
     "              VERSION 2.1"
101 DIM A$[64],E$[64],E1$[32],E2
    $[19],E3$[21],E4$[15],E5$[13
    ],E6$[28],E7$[16],L(3,3)
102 DISP @ DISP "        -PLEASE
    STAND BY-" @ DISP
103 E2$="TOO MANY PTS. MAX= " @
    E3$="BAD DATA CHARACTER ->"
    @ E4$="BAD DATA LINE. "
104 C1,E9=0 @ B$="LRUDCKPS, *EMH
    " @ C2=14 @ C3=10
105 E5$="CHECK: +-. 'S" @ E6$="B
    AD CHAR. IN COMMAND LINE ->"
     @ E7$=" COMMAS (OR *'S)"
108 X1=0 @ Y1=10 @ X2,Z=255 @ Y2
    =191 @ GOSUB 8000
110 X1,Z1=0 @ Y1=37/255 @ X2,Z2=
    1 @ Y2=218/255 @ GOSUB 8001
112 X,Y=.5 @ Z=-(90.5/255) @ V1=
    -90 @ V2=1 @ GOSUB 8003
114 X,Y=.5 @ Z=-(90.5/255) @ V1=
    -90 @ V2=2 @ GOSUB 8003
116 X=127.5 @ Y=100.5 @ Z=200 @
    GOSUB 8008
122 DISP
124 DISP "YOU MAY ENTER EITHER A
     DATA"
126 DISP "LINE OR A COMMAND LINE
    "
128 DISP "THE PLOT WILL FIRST AP
    PEAR"
130 DISP "AFTER YOU ENTER YOUR T
    HIRD"
132 DISP "DATA LINE.  TYPE H FOR
     HELP." @ DISP
133 ON ERROR GOSUB 200
134 IF C1<=2 OR E9=1 THEN GOSUB
    190 ELSE GOSUB 210
135 IF E9=1 THEN GOTO 146 ELSE O
    FF ERROR
136 CLEAR @ GCLEAR 10 @ E9=0
138 DISP USING "8/,12X,7A" ; "WO
    RKING"
139 IF LEN(A$)=0 THEN E1$=E4$ @
    E$=" " @ GOSUB 220 @ GOTO 14
    6
140 E$=A$
142 IF FNA(A$[1,1]) THEN GOSUB 2
    30 ELSE GOSUB 430
```

```
144 IF E9=0 THEN CLEAR @ DISP US
    ING "8/"
146 GOTO 133
149 REM * FNA   IS C$ NUMERIC *
150 DEF FNA(C$)
152 IF C$[1,1]="." OR C$[1,1]="+
    " OR C$[1,1]="-" THEN FNA=1
    @ GOTO 158
154 A=NUM(C$[1,1])
156 IF A<48 OR A>57 THEN FNA=0 E
    LSE FNA=1
158 FN END
169 REM * CHECK FOR BAD +-. 'S *
170 DEF FNB(D$[32])
172 IF D$[1,1]="." THEN D=1 ELSE
     D=0
174 FOR I=2 TO LEN(D$)
176 IF D$[I,I]="+" THEN FNB=1 @
    GOTO 188
178 IF D$[I,I]="-" THEN FNB=1 @
    GOTO 188
180 IF D$[I,I]="." AND D=1 THEN
    FNB=1 @ GOTO 188
182 IF D$[I,I]="." THEN D=1
184 NEXT I
186 FNB=0
188 FN END
189 REM * ALPHA INPUT *
190 E9=0 @ IF C1<3 THEN GOTO 194
191 MOVE 0,0
192 LABEL "ENTER:"
193 MOVE 45,0
194 DISP "ENTER DATA OR COMMAND
    LINE:"
196 INPUT A$
198 RETURN
200 OFF ERROR
202 E1$=E4$ @ E$=" " @ GOSUB 220
204 RETURN
209 REM * GRAPH INPUT *
210 MOVE 0,0
214 LABEL "ENTER:"
215 MOVE 45,0
216 INPUT A$
218 RETURN
219 REM * ERROR PROCESSOR *
220 BEEP
222 CLEAR
224 DISP USING "8/"
225 DISP E1$,E$
226 E9=1
228 RETURN
229 REM * CHECK DATA LINE *
230 IF T>T1-3 THEN E1$=E2$&VAL$(
    C1) @ GOSUB 220 @ RETURN
232 A1=0
234 A2=LEN(A$)
236 FOR I=1 TO A2
238 IF A$[I,I]="*" THEN A$[I,I]=
    ","
```

```
240 IF A$[I,I]="," THEN A1=A1+1
241 IF A1=3 THEN GOTO 246
242 IF A$[I,I]<>"," AND NOT FNA(
    A$[I,I]) THEN E1$=E3$&A$[I,I
    ] @ GOSUB 220 @ RETURN
244 NEXT I
246 IF A1=2 OR A1=3 THEN GOSUB 2
    50 ELSE E1$=E4$&VAL$(A1)&E7$
     @ GOSUB 220 @ RETURN
248 RETURN
249 REM *CONVERT DATA LINE *
250 A3=POS(A$[1,A2],",") @ A4=PO
    S(A$[A3+1,A2],",")+A3 @ A5=P
    OS(A$[A4+1,A2],",")+A4
251 IF A3<=1 OR A4<=A3+1 OR A4>=
    A2 OR A1=3 AND A5<=A4+1 OR A
    1=3 AND A5>=A2 THEN GOTO 267
252 IF FNB(A$[1,A3-1]) THEN GOTO
     266 ELSE B1=VAL(A$[1,A3-1])
254 IF FNB(A$[A3+1,A4-1]) THEN G
    OTO 266 ELSE B2=VAL(A$[A3+1,
    A4-1])
256 IF A1=2 THEN A6=A2 ELSE A6=A
    5-1
258 IF FNB(A$[A4+1,A6]) THEN GOT
    O 266
260 B3=VAL(A$[A4+1,A6])
261 C1=C1+1
262 IF C1<=3 THEN GOSUB 270 ELSE
     GOSUB 290
264 RETURN
266 E1$=E4$&E5$ @ GOSUB 220 @ RE
    TURN
267 E1$=E4$&E7$ @ GOSUB 220 @ RE
    TURN
269 REM * 1ST 3 PTS & RE-SCALE *
270 X=B1 @ Y=B2 @ Z=0 @ GOSUB 85
    00
272 X1(T)=2
274 X=B1 @ Y=B2 @ Z=B3 @ GOSUB 8
    500
276 X1(T)=1
278 IF A1=3 THEN X=B1 @ Y=B2 @ Z
    =B3 @ GOSUB 8500 @ X2$[T,T]=
    A$[A5+1,A5+1] @ X1(T)=4
280 IF C1=3 THEN GOSUB 310
282 RETURN
289 REM * DRAW SPIKE *
290 L1=B1<L(1,1) @ L2=B1>L(1,2)
292 L3=B2<L(2,1) @ L4=B2>L(2,2)
294 L5=B3<L(3,1) @ L6=B3>L(3,2)
296 IF L1 OR L2 OR L3 OR L4 OR L
    5 OR L6 THEN GOSUB 270 @ GOS
    UB 310 @ RETURN
298 X=B1 @ Y=B2 @ Z=L(3,1) @ GOS
    UB 8022
300 X=B1 @ Y=B2 @ Z=B3 @ GOSUB 8
    026
302 IF A1=2 THEN RETURN
304 X=B1 @ Y=B2 @ Z=B3 @ V$=A$[A
    5+1,A5+1] @ GOSUB 8028
```

```
306 RETURN
309 REM * RE-SCALE *
310 M1,M2,M3=-(99^99) @ N1,N2,N3
    =99^99
312 FOR I=1 TO T
314 IF X1(I)<>1 THEN GOTO 320
316 M1=MAX(M1,X(1,I)) @ M2=MAX(M
    2,X(2,I)) @ M3=MAX(M3,X(3,I)
    )
318 N1=MIN(N1,X(1,I)) @ N2=MIN(N
    2,X(2,I)) @ N3=MIN(N3,X(3,I)
    )
320 NEXT I
322 D1=(M1-N1)/10 @ D2=(M2-N2)/1
    0 @ D3=(M3-N3)/10
324 L(1,1)=N1-D1 @ L(1,2)=M1+D1
    @ L(2,1)=N2-D2 @ L(2,2)=M2+D
    2 @ L(3,1)=N3-D3 @ L(3,2)=M3
    +D3
326 FOR I=1 TO 3
328 GOSUB 600
330 GOSUB 620
332 NEXT I
334 D4=.2
336 D5=D4*(L(1,2)-L(1,1)) @ D6=3
    7*(L(1,2)-L(1,1)+2*D5)/181 @
     X1=L(1,1)-D5-D6 @ X2=L(1,2)
    +D5+D6
338 D5=D4*(L(2,2)-L(2,1)) @ Y1=L
    (2,1)-D5 @ Y2=L(2,2)+D5
340 D5=D4*(L(3,2)-L(3,1)) @ Z1=L
    (3,2)+D5 @ Z2=Z1+255*(L(3,2)
    -L(3,1)+2*D5)/181
342 GOSUB 8001
343 GOSUB 360
344 GOSUB 380
346 PRINT "XSCALE:" @ PRINT "XMI
    N=";L(1,1) @ PRINT "XMAX=";L
    (1,2) @ PRINT "X-TIC=";L(1,3
    )
347 PRINT
348 PRINT "YSCALE:" @ PRINT "YMI
    N=";L(2,1) @ PRINT "YMAX=";L
    (2,2) @ PRINT "Y-TIC=";L(2,3
    )
349 PRINT
350 PRINT "ZSCALE:" @ PRINT "ZMI
    N=";L(3,1) @ PRINT "ZMAX=";L
    (3,2) @ PRINT "Z-TIC=";L(3,3
    )
351 PRINT
352 RETURN
359 REM * ADJUST BASE OF SPIKE *
360 FOR I=1 TO T
362 IF X1(I)=2 THEN X(3,I)=L(3,1
    )
364 NEXT I
366 RETURN
379 REM * PLOT THE AXIS ETC *
380 X=L(1,1) @ Z=L(3,1) @ V1=L(2
    ,3) @ V2=L(2,1) @ V3=L(2,2)
    @ GOSUB 8006
```

```
382 X=L(1,1) @ Z=L(3,1) @ V1=L(2
    ,3) @ V2=L(2,1) @ V3=L(2,2)
    @ GOSUB 8006
384 X=L(1,1) @ Y=L(2,1) @ V1=L(3
    ,3) @ V2=L(3,1) @ V3=L(3,2)
    @ GOSUB 8007
386 Z=L(3,1) @ Y=L(2,1) @ V1=L(1
    ,3) @ V2=L(1,1) @ V3=L(1,2)
    @ GOSUB 8005
387 GCLEAR
388 GOSUB 8010
390 X=L(1,1) @ Z=L(3,2) @ V1=L(2
    ,3) @ V2=L(2,1) @ V3=L(2,2)
    @ GOSUB 8006
392 X=L(1,1) @ Y=L(2,2) @ V1=L(3
    ,3) @ V2=L(3,1) @ V3=L(3,2)
    @ GOSUB 8007
394 Z=L(3,1) @ Y=L(2,2) @ V1=L(1
    ,3) @ V2=L(1,1) @ V3=L(1,2)
    @ GOSUB 8005
396 GOSUB 8010
398 X=L(1,2) @ Z=L(3,1) @ V1=L(2
    ,3) @ V2=L(2,1) @ V3=L(2,2)
    @ GOSUB 8006
400 X=L(1,2) @ Y=L(2,1) @ V1=L(3
    ,3) @ V2=L(3,1) @ V3=L(3,2)
    @ GOSUB 8007
402 Z=L(3,2) @ Y=L(2,1) @ V1=L(1
    ,3) @ V2=L(1,1) @ V3=L(1,2)
    @ GOSUB 8005
404 GOSUB 8010
406 Z=L(3,2) @ Y=L(2,2) @ V1=L(1
    ,3) @ V2=L(1,1) @ V3=L(1,2)
    @ GOSUB 8005
408 X=L(1,2) @ Y=L(2,2) @ V1=L(3
    ,3) @ V2=L(3,1) @ V3=L(3,2)
    @ GOSUB 8007
410 S1(2,1),S1(2,2),S1(2,3),S1(2
    ,4),S1(2,5),S1(2,6)=0
412 GOSUB 8010
414 GOSUB 8011 @ GOSUB 8012
416 RETURN
429 REM * CHECK COMMAND LINE *
430 B1=LEN(A$)
432 FOR I9=1 TO B1
434 GOSUB 450
436 IF I1=C2+1 THEN E1$=E6$&A$[I
    9,I9] @ GOSUB 220 @ RETURN
438 NEXT I9
440 GOSUB 470
442 IF I1<>7 AND I1<>12 AND C1>2
     AND I1<>13 THEN GOSUB 500
444 RETURN
449 REM * ID'FY COMMAND *
450 FOR I1=1 TO C2
452 IF A$[I9,I9]=B$[I1,I1] THEN
    RETURN
454 NEXT I1
458 RETURN
469 REM * EXEC COMMAND *
```

```
470 FOR I9=1 TO B1
472 GOSUB 450
474 IF I1=C2+1 THEN PAUSE
476 ON I1 GOSUB 490,490,520,520,
    530,530,540,630,570,570,570,
    660,730,770
478 NEXT I9
480 RETURN
489 REM * R L *
490 V1=C3
492 IF I1=1 THEN V1=-V1
494 V2=2 @ X=127.5 @ Y=1 @ Z=-90
    .5 @ GOSUB 8033
496 RETURN
499 REM * PLOT COMMAND RESULTS *
500 GOSUB 380
502 RETURN
519 REM * U D *
520 V1=C3
522 IF I1=3 THEN V1=-C3
524 V2=1 @ X=1 @ Y=100.5 @ Z=-90
    .5 @ GOSUB 8033
526 RETURN
529 REM * C K *
530 V1=C3
532 IF I1=5 THEN V1=-V1
534 V2=3 @ X=127.5 @ Y=100.5 @ Z
    =1 @ GOSUB 8033
536 RETURN
539 REM * P *
540 CLEAR
542 DISP USING 543
543 IMAGE 4/,"1: USE PRINTER/PLO
    TTER",/,"2: USE PEN PLOTTER"
    ,2/,"ENTER THE NUMBER 1 OR 2
    "
544 INPUT H1$
545 IF H1$<>"2" AND H1$<>"1" THE
    N BEEP @ CLEAR @ GOTO 542
546 REM IF H1$="2" THEN PLOTER I
    S X ELSE PLOTER IS 1
548 REM IF H1$="2" THEN GOTO 556
550 GRAPH @ IF I9<>1 THEN GOSUB
    500
552 COPY
554 RETURN
556 REM GOSUB 310
558 REM PLOTER IS 1
559 REM GOSUB 500
560 REM RETURN
569 REM * ,* OR blank *
570 A=1
572 RETURN
599 REM * COMPUTE TICS *
600 F=LGT(ABS(L(I,2)-L(I,1)))
602 IF F>=0 THEN F1=INT(F) @ F2=
    F-F1 @ F2=10^F2
603 IF F<0 THEN F1=ABS(IP(F))+1
    @ F=F+F1 @ F1=-F1 @ F2=10^F
604 F3=IP(F2/2)*.5
```

```
606 IF F3=0 THEN F3=.25
612 L(I,3)=F3*10^F1
614 RETURN
619 REM * COMPUTE AXIS LIMITS *
620 L(I,1)=INT(L(I,1)/L(I,3))*L(
    I,3)
622 L(I,2)=(1+INT(L(I,2)/L(I,3))
    )*L(I,3)
623 H3=N1
624 IF I=2 THEN H3=N2
625 IF I=3 THEN H3=N3
626 IF L(I,1)<0 AND H3>=0 THEN L
    (I,1)=0
627 IF L(I,1)>0 AND H3<=3*L(I,3)
     THEN L(I,1)=0
628 RETURN
629 REM * STEREO *
630 CLEAR @ DISP USING 543 @ INP
    UT H1$@ REM IF H1$="2" THEN
    PLOTTER IS X ELSE PLOTTER IS
     1
631 W6(1)=S2(1) @ W6(2)=S2(2) @
    W6(3)=S2(3) @ H1=1
632 X=-(W6(3)*TAN(3))+W6(1) @ Y=
    S2(2) @ Z=S2(3) @ GOSUB 8008
634 GOSUB 380
636 IF H1=1 THEN COPY
638 GCLEAR
640 X=W6(3)*TAN(3)+W6(1) @ Y=S2(
    2) @ Z=S2(3) @ GOSUB 8008
642 IF H1=2 THEN DISP "PUSH CONT
     WHEN PLOTTER IS READY" @ PA
    USE
644 GOSUB 380
646 IF H1=1 THEN COPY
648 GCLEAR
650 X=W6(1) @ Y=W6(2) @ Z=W6(3)
    @ GOSUB 8008
651 REM IF H1=2 THEN PLOTTER IS
    1
652 RETURN
659 REM * E(DIT) *
660 H1=T @ H0=C1 @ X2$[T1,T1]="
    " @ H5=1
662 FOR H2=1 TO H1
664 IF X1(H2)=2 THEN GOSUB 680 E
    LSE BEEP @ PAUSE
666 IF H4=3 THEN GOTO 672
668 H2=H2+H3 @ H5=H5+1
670 NEXT H2
672 T=H1 @ C1=H0
673 IF C1>2 THEN GOSUB 310
674 RETURN
679 REM * QUERY & CHANGE *
680 ALPHA @ CLEAR
682 DISP "DATA POINT NO.:  ";H5
684 DISP USING "D.2DE,X,D.2DE,X,
    D.2DE,X,A" ; X(1,H2+1),X(2,H
    2+1),X(3,H2+1),X2$[H2+2,H2+2
    ]
```

```
686 DISP USING "/,8A,/,15A,/,21A
    ,/" ; "  1:  OK","  2:  CHAN
    GE IT","  3:  NO MORE CHANGE
    S"
687 DISP @ DISP "ENTER THE NUMBE
    R:  1, 2, OR 3"
688 INPUT H4$
690 IF H4$<>"1" AND H4$<>"2" AND
     H4$<>"3" THEN DISP "BAD INP
    UT" @ GOTO 682
691 H4=VAL(H4$)
692 IF H4=3 THEN RETURN
694 IF H4=1 THEN GOTO 722
696 C1=0 @ T=H2-1
698 DISP "ENTER DATA LINE"
700 INPUT A$
702 H6,H8=0
704 FOR H7=1 TO LEN(A$)-1
706 IF A$[H7,H7]="*" THEN A$[H7,
    H7]=","
708 IF A$[H7,H7]="," THEN H6=H6+
    1 @ H8=H7
710 NEXT H7
712 IF H6=3 AND X1(H2+2)<>4 THEN
     A$=A$[1,H8-1]
714 IF H6=2 AND X1(H2+2)=4 THEN
    A$=A$&", "
716 E$=A$ @ CLEAR @ E9=0
718 GOSUB 230
720 IF E9=1 THEN GOTO 682
722 IF X1(H2+2)=4 THEN H3=2 ELSE
     H3=1
724 RETURN
729 REM * M(EMORY) *
730 CLEAR @ DISP USING "4/,16A,/
    ,16A,2/,23A" ; "1:  INPUT DA
    TA","2:  OUTPUT DATA","ENTER
     1 OR 2"
732 INPUT H1$@ IF H1$<>"1" AND H
    1$<>"2" THEN BEEP @ GOTO 730
734 CLEAR @ DISP "ENTER THE NAME
     OF THE DATA FILE" @ INPUT A
    $@ H1=VAL(H1$)
736 IF H1=2 THEN GOTO 748
738 ON ERROR GOTO 746
740 ASSIGN# 5 TO A$
742 READ# 5 ; S(,),S1(,),S6(,),X
    2$,X(,),X1(),T,S4(),S5(),S2(
    ),L(,),C1
743 ASSIGN# 5 TO *
744 OFF ERROR @ GOSUB 310 @ RETU
    RN
746 OFF ERROR @ ASSIGN# 5 TO * @
     BEEP @ CLEAR @ DISP "CAN'T
    READ ";A$ @ WAIT 3000 @ RETU
    RN
748 ON ERROR GOTO 758
750 CREATE A$,60
752 ASSIGN# 5 TO A$
754 PRINT# 5 ; S(,),S1(,),S6(,),
    X2$,X(,),X1(),T,S4(),S5(),S2
    (),L(,),C1
```

```
756 OFF ERROR @ ASSIGN# 5 TO * @
     RETURN
758 OFF ERROR @ ASSIGN# 5 TO * @
     BEEP @ DISP "CAN'T CREATE O
    R WRITE ";A$ @ WAIT 3000 @ R
    ETURN
769 REM * H(ELP) *
770 PRINT @ PRINT @ PRINT
772 PRINT "TO ENTER DATA USE EIT
    HER OF     THESE FORMATS:" @
     PRINT "X*Y*Z" @ PRINT "X*Y*
    Z*L"
774 PRINT "WHERE X,Y,&Z ARE COOR
    DINATES" @ PRINT "AND L IS A
     LABEL.  FOR EXAMPLE:"
776 PRINT "-1.29*45.923*4.2*+" @
     PRINT
778 PRINT "TO ENTER A COMMAND US
    E ANY" @ PRINT "COMBINATION
    OF THESE LETTERS:" @ PRINT
780 PRINT "          TO ROTATE 10
     DEGREES:" @ PRINT "L OR R:
     LEFT OR RIGHT"
782 PRINT "U OR D:  UP OR DOWN"
    @ PRINT "C OR K:  CLOCKWISE
    OR"
783 PRINT "          KOUNTERCLOCK
    WISE"
784 PRINT @ PRINT "E:       EDIT
     DATA"
786 PRINT "P:       PLOT HARD CO
    PY"
788 PRINT "S:       PLOT STEREO
    PAIR"
790 PRINT "M:       STORE OR RET
    RIEVE" @ PRINT "         DAT
    A FROM MASS MEMORY"
792 PRINT "H:       PRINT THIS Q
    UICK GUIDE"
794 PRINT @ PRINT "FOR EXAMPLE:"
     @ PRINT "RUUKKKPDDDS"
796 PRINT
798 RETURN
8000 GOTO 8050
8001 GOTO 8110
8002 GOTO 8170
8003 GOTO 8200
8005 GOTO 8300
8006 GOTO 8310
8007 GOTO 8320
8008 GOTO 8520
8009 GOTO 8013
8010 GOTO 8350
8011 GOTO 8750
8012 GOTO 8770
8013 BEEP @ DISP "E" @ PAUSE
8014 GOTO 8013
8020 GOTO 8470
8021 GOTO 8013
8022 GOTO 8630
```

```
8023 GOTO 8013
8024 GOTO 8013
8025 GOTO 8013
8026 GOTO 8610
8027 GOTO 8013
8028 GOTO 8700
8031 GOTO 8510
8032 GOTO 8240
8033 GOTO 8290
8034 GOTO 8220
8049 REM  *** INITIATE:8000 ***
8050 T=0
8051 T1=300
8052 X2$[T1,T1]=" "
8053 S5(1),S5(2),S5(3)=0
8054 S3(1),S3(2)=0 @ S3(3)=0
8055 DEG
8056 GCLEAR @ ALPHA
8057 SCALE 0,255,0,191
8058 GOSUB 8100
8066 FOR R=0 TO T1
8067 X1(R)=1
8068 NEXT R
8069 FOR R=1 TO 6
8070 S1(1,R),S1(2,R),S1(3,R)=0
8071 NEXT R
8072 S4(1)=X1 @ S4(2)=X2 @ S4(3)
     =Y1 @ S4(4)=Y2
8073 FOR R=1 TO 4
8074 IF S4(R)<0 THEN S4(R)=0
8075 NEXT R
8076 IF X1>255 THEN S4(1)=255
8077 IF X2>255 THEN S4(2)=255
8078 IF Y1>191 THEN S4(3)=191
8079 IF Y2>191 THEN S4(4)=191
8080 S(4,1)=S4(1) @ S(4,2)=S4(3)
      @ S(4,3)=0
8081 S(1,1)=S4(2)-S4(1) @ S(2,2)
     =S4(4)-S4(3) @ S(3,3)=Z
8086 S6(1,1),S6(2,1),S6(3,1)=0 @
      S6(1,2),S6(2,2),S6(3,2)=1
8087 S2(1)=(S4(2)-S4(1))/2+S4(1)
      @ S2(2)=(S4(4)-S4(3))/2+S4
     (3) @ S2(3)=10*Z
8090 RETURN
8099 REM *** SET S TO UNITY ***
8100 FOR R=1 TO 4
8101 S(R,1),S(R,2),S(R,3),S(R,4)
     =0
8102 S(R,R)=1
8103 NEXT R
8104 RETURN
8109 REM *** SCALE: 8001 ***
8110 GOSUB 8130
8112 W(1,1)=(S6(1,2)-S6(1,1))/(X
     2-X1)
8114 W(2,2)=(S6(2,2)-S6(2,1))/(Y
     2-Y1)
8116 W(3,3)=(S6(3,2)-S6(3,1))/(Z
     2-Z1)
```

```
8118 W(4,1)=S6(1,1)-X1*W(1,1)
8120 W(4,2)=S6(2,1)-Y1*W(2,2)
8122 W(4,3)=S6(3,1)-Z1*W(3,3)
8124 S6(1,1)=X1 @ S6(1,2)=X2 @ S
     6(2,1)=Y1 @ S6(2,2)=Y2 @ S6
     (3,1)=Z1 @ S6(3,2)=Z2
8126 GOSUB 8570
8127 U1=0
8128 RETURN
8129 REM  *SET W TO UNITY*
8130 FOR R=1 TO 4
8132 W(R,1),W(R,2),W(R,3),W(R,4)
     =0
8134 W(R,R)=1
8136 NEXT R
8138 RETURN
8139 REM   * S x W *
8140 FOR R=1 TO 4
8142 FOR R1=1 TO 4
8144 W3(R,R1)=S(R,1)*W(1,R1)+S(R
     ,2)*W(2,R1)+S(R,3)*W(3,R1)+
     S(R,4)*W(4,R1)
8146 NEXT R1
8154 NEXT R
8156 GOSUB 8600
8157 S5(3)=1
8158 RETURN
8169 REM *** TRANSL_UU: 8002 ***
8170 GOSUB 8130
8172 W(4,1)=X
8174 W(4,2)=Y
8176 W(4,3)=Z
8192 GOSUB 8570
8194 RETURN
8199 REM *** ROTATE_UU: 8003 ***
8200 GOSUB 8220
8201 X=-X
8202 Y=-Y
8204 Z=-Z
8206 GOSUB 8240
8208 GOSUB 8260
8210 X=-X
8212 Y=-Y
8214 Z=-Z
8216 GOSUB 8240
8218 RETURN
8219 REM *SCALE XYZ TO GU*
8220 U1=(S(1,1)*X+S(2,1)*Y+S(3,1
     )*Z+S(4,1))/S(4,4)
8222 U2=(S(1,2)*X+S(2,2)*Y+S(3,2
     )*Z+S(4,2))/S(4,4)
8224 U3=(S(1,3)*X+S(2,3)*Y+S(3,3
     )*Z+S(4,3))/S(4,4)
8226 X=U1
8228 Y=U2
8230 Z=U3
8232 RETURN
8239 REM *** TRANSL_GU: 8032 ***
8240 GOSUB 8130
8242 W(4,1)=X
```

```
8244 W(4,2)=Y
8246 W(4,3)=Z
8248 GOSUB 8140
8250 RETURN
8259 REM * DOES ROTATION *
8260 U1,U2,U3=0
8262 IF V2=1 THEN U1=1
8264 IF V2=2 THEN U2=1
8266 IF V2=3 THEN U3=1
8267 GOSUB 8130
8268 W(1,1)=U1+(1-U1)*COS(V1)
8270 W(1,2)=U3*SIN(V1)
8272 W(1,3)=-(U2*SIN(V1))
8274 W(2,1)=-(U3*SIN(V1))
8276 W(2,2)=U2+(1-U2)*COS(V1)
8278 W(2,3)=U1*SIN(V1)
8280 W(3,1)=U2*SIN(V1)
8282 W(3,2)=-(U1*SIN(V1))
8284 W(3,3)=U3+(1-U3)*COS(V1)
8286 GOSUB 8140
8288 RETURN
8289 REM *** ROTATE_GU: 8033 ***
8290 X=-X @ Y=-Y @ Z=-Z
8291 GOSUB 8240
8292 GOSUB 8260
8293 X=-X @ Y=-Y @ Z=-Z
8294 GOSUB 8240
8295 RETURN
8299 REM  *** X-AXIS:8005 ***
8300 X=0
8302 U1=1
8304 GOSUB 8330
8306 RETURN
8309 REM  *** Y-AXIS: 8006 ***
8310 Y=0
8312 U1=2
8314 GOSUB 8330
8316 RETURN
8319 REM  *** Z-AXIS: 8007 ***
8320 Z=0
8322 U1=3
8324 GOSUB 8330
8326 RETURN
8329 REM  * AXIS *
8330 S1(U1,1)=X
8332 S1(U1,2)=Y
8334 S1(U1,3)=Z
8336 S1(U1,4)=V1
8338 S1(U1,5)=V2
8340 S1(U1,6)=V3
8342 RETURN
8349 REM  *** AXISPLOT: 8010 ***
8350 FOR R=1 TO 3
8353 IF S1(R,5)=S1(R,6) THEN GOT
     O 8424
8354 U1=0
8356 GOSUB 8550
8360 GOSUB 8510
8362 W5(1,1)=X
8364 W5(1,2)=Y @ W5(1,3)=R5
```

```
8366 U1=1
8368 GOSUB 8550
8372 GOSUB 8510
8374 W5(2,1)=X
8376 W5(2,2)=Y @ W5(2,3)=R5
8377 R8=1
8378 GOSUB 8430
8379 IF R8=0 THEN GOTO 8424
8380 MOVE W5(1,1),W5(1,2)
8382 DRAW W5(2,1),W5(2,2)
8383 GOSUB 8860
8384 U1=S1(R,5)
8386 U2=S1(R,6)
8388 IF S1(R,4)>0 THEN GOTO 8394
8390 U1=U2
8392 U2=S1(R,5)
8394 FOR R1=U1 TO U2 STEP S1(R,4
     )
8395 X=S1(R,1) @ Y=S1(R,2) @ Z=S
     1(R,3)
8396 IF R=1 THEN X=R1
8398 IF R=2 THEN Y=R1
8399 IF R=3 THEN Z=R1
8400 GOSUB 8510
8401 W5(1,1)=X @ W5(1,2)=Y @ W5(
     1,3)=R5
8402 X=S1(R,1) @ Y=S1(R,2) @ Z=S
     1(R,3)
8403 IF R=1 THEN X=R1 @ Y=Y+W7
8404 IF R=2 THEN Y=R1 @ X=X+W7
8405 IF R=3 THEN Z=R1 @ Y=Y+W7
8406 GOSUB 8510
8407 W5(2,1)=X @ W5(2,2)=Y @ W5(
     2,3)=R5
8408 GOSUB 8430
8409 IF R8=0 THEN GOTO 8420
8412 MOVE W5(1,1),W5(1,2)
8414 DRAW W5(2,1),W5(2,2)
8420 NEXT R1
8424 NEXT R
8426 RETURN
8429 REM * CLIP *
8430 IF W5(1,3) OR W5(2,3) THEN
     GOSUB 8730
8431 R5,R6=0 @ R8=1
8432 IF S4(1)<=MIN(W5(1,1),W5(2,
     1)) AND S4(2)>=MAX(W5(1,1),
     W5(2,1)) THEN R5=1
8433 IF S4(3)<=MIN(W5(1,2),W5(2,
     2)) AND S4(4)>=MAX(W5(1,2),
     W5(2,2)) THEN R6=1
8434 IF R5 AND R6 THEN RETURN
8436 R5,R6=1
8438 IF W5(1,1)>W5(2,1) THEN R5=
     2
8440 IF W5(2,2)>W5(1,2) THEN R6=
     2
8444 IF S4(1)>=W5(ABS(R5-3),1) T
     HEN R8=0 @ RETURN
8446 IF S4(2)<=W5(R5,1) THEN R8=
     0 @ RETURN
```

```
8448 IF S4(3)>=W5(R6,2) THEN R8=
     0 @ RETURN
8450 IF S4(4)<=W5(ABS(R6-3),2) T
     HEN R8=0 @ RETURN
8452 IF S4(1)>W5(R5,1) THEN W5(R
     5,2)=FNX(1) @ W5(R5,1)=S4(1
     )
8453 R5=ABS(R5-3)
8454 IF S4(2)<W5(R5,1) THEN W5(R
     5,2)=FNX(2) @ W5(R5,1)=S4(2
     )
8456 IF S4(3)>W5(ABS(R6-3),2) TH
     EN W5(ABS(R6-3),1)=FNY(3) @
      W5(ABS(R6-3),2)=S4(3)
8460 IF S4(4)<W5(R6,2) THEN W5(R
     6,1)=FNY(4) @ W5(R6,2)=S4(4
     )
8462 RETURN
8463 REM *FNX CLIP LEFT & RIGHT*
8464 DEF FNX(R7) = W5(1,2)+(S4(R
     7)-W5(1,1))*(W5(2,2)-W5(1,2
     ))/(W5(2,1)-W5(1,1))
8465 REM *FNY CLIP TOP & BOTTOM*
8466 DEF FNY(R7) = W5(1,1)+(S4(R
     7)-W5(1,2))*(W5(2,1)-W5(1,1
     ))/(W5(2,2)-W5(1,2))
8469 REM *** PLOT: 8020 ***
8470 IF T>=T1 THEN PRINT "ERROR.
      TOO MANY PTS" @ BEEP @ RET
     URN
8472 GOSUB 8500
8474 X1(T)=0
8476 IF X1(T-1)<2 THEN GOSUB 848
     3 ELSE GOSUB 8530
8477 RETURN
8478 GOSUB 8510
8482 REM * INK LINE *
8483 GOSUB 8510
8484 W5(2,1)=X
8485 W5(2,2)=Y @ W5(2,3)=R5
8487 X=X(1,T-1) @ Y=X(2,T-1) @ Z
     =X(3,T-1)
8488 GOSUB 8510
8489 W5(1,1)=X @ W5(1,2)=Y @ W5(
     1,3)=R5
8490 GOSUB 8430
8491 IF R8=0 THEN RETURN
8492 MOVE W5(1,1),W5(1,2)
8493 DRAW W5(2,1),W5(2,2)
8494 S5(3)=0
8495 RETURN
8499 REM  * ADD PTS INCR T *
8500 T=T+1
8502 X(1,T)=X
8504 X(2,T)=Y
8506 X(3,T)=Z
8508 RETURN
8509 REM *** LOCATION: 8031 ***
8510 U5=(X*S(1,1)+Y*S(2,1)+Z*S(3
     ,1)+S(4,1))/S(4,4)
```

```
8511 U6=(X*S(1,2)+Y*S(2,2)+Z*S(3
     ,2)+S(4,2))/S(4,4)
8512 U7=(X*S(1,3)+Y*S(2,3)+Z*S(3
     ,3)+S(4,3))/S(4,4)
8513 U8=-(U7/S2(3))+1
8514 IF U8=0 THEN U8=.000001
8515 U5=(U5-S2(1))/U8 @ U6=(U6-S
     2(2))/U8 @ U7=U7/U8
8516 X=U5+S2(1) @ Y=U6+S2(2)
8517 IF U8>0 THEN R5=0 ELSE R5=1
8518 RETURN
8519 REM *** PROJPT: 8008 ***
8520 S2(1)=X
8522 S2(2)=Y
8524 S2(3)=Z
8528 RETURN
8529 REM * PLOT SPOT *
8530 GOSUB 8510
8532 W5(1,1),W5(2,1)=X @ W5(1,2)
     ,W5(2,2)=Y @ W5(1,3),W5(2,3
     )=R5
8534 GOSUB 8430
8536 IF R8=0 THEN RETURN
8538 MOVE W5(1,1),W5(1,2) @ PLOT
      W5(1,1),W5(1,2)
8540 RETURN
8549 REM * AXIS COORDS *
8550 X=S1(R,1)
8552 Y=S1(R,2)
8554 Z=S1(R,3)
8556 IF R=1 THEN X=S1(R,5+U1)
8558 IF R=2 THEN Y=S1(R,5+U1)
8560 IF R=3 THEN Z=S1(R,5+U1)
8562 RETURN
8569 REM * W x S *
8570 FOR R=1 TO 4
8572 FOR R1=1 TO 4
8574 W3(R,R1)=W(R,1)*S(1,R1)+W(R
     ,2)*S(2,R1)+W(R,3)*S(3,R1)+
     W(R,4)*S(4,R1)
8576 NEXT R1
8586 NEXT R
8587 S5(3)=1
8588 GOSUB 8600
8590 RETURN
8599 REM * W3 INTO S *
8600 FOR R=1 TO 4
8602 S(R,1)=W3(R,1) @ S(R,2)=W3(
     R,2) @ S(R,3)=W3(R,3) @ S(R
     ,4)=W3(R,4)
8604 NEXT R
8606 RETURN
8609 REM *** DRAW: 8026 ***
8610 IF T>=T1 THEN PRINT "ERROR.
       TOO MANY PTS" @ BEEP @ RE
     TURN
8612 GOSUB 8500
8614 X1(T)=1
8616 GOSUB 8483
8618 RETURN
```

```
8629 REM *** MOVE: 8022 **
8630 IF T>=T1 THEN PRINT "ERROR.
       TOO MANY PTS" @ BEEP @ RE
     TURN
8632 GOSUB 8500
8634 X1(T)=2
8636 RETURN
8648 RETURN
8699 REM *** LABEL: 8028 ***
8700 IF T>=T1 THEN PRINT "ERROR.
       TOO MANY PTS" @ BEEP @ RE
     TURN
8702 GOSUB 8500
8703 X2$[T,T]=V$[1,1]
8704 X1(T)=4
8706 GOSUB 8510
8708 W5(1,1),W5(2,1)=X-2
8710 W5(1,2),W5(2,2)=Y-4
8711 W5(1,3),W5(2,3)=R5
8712 GOSUB 8430
8714 IF R8=0 THEN RETURN
8716 MOVE W5(1,1),W5(1,2)
8718 LABEL X2$[T,T]
8720 RETURN
8729 REM *TRUNC INFINITE LINE*
8730 IF W5(1,3) AND W5(2,3) THEN
      W5(1,1),W5(2,1),W5(1,2),W5
     (2,2)=10000000000 @ RETURN
8732 IF W5(1,3)=1 THEN U5=1 ELSE
      U5=2
8734 IF W5(1,3)=1 THEN U6=2 ELSE
      U6=1
8736 IF W5(2,1)=W5(1,1) THEN GOT
     O 8746
8738 U3=(W5(2,2)-W5(1,2))/(W5(2,
     1)-W5(1,1)) @ U4=W5(1,2)-U3
     *W5(1,1)
8740 IF W5(U5,1)<W5(U6,1) THEN W
     5(U5,1)=10000000000 ELSE W5
     (U5,1)=-10000000000
8742 W5(U5,2)=U3*W5(U5,1)+U4
8744 RETURN
8746 IF W5(U5,2)<W5(U6,2) THEN W
     5(U5,2)=10000000000 ELSE W5
     (U5,2)=-10000000000
8748 RETURN
8749 REM ***LINE_PLOT:     ***
8750 T9=T
8752 FOR T=1 TO T9
8753 X=X(1,T) @ Y=X(2,T) @ Z=X(3
     ,T)
8754 IF X1(T)=0 THEN GOSUB 8476
8756 IF X1(T)=1 THEN GOSUB 8483
8758 IF X1(T)=2 THEN GOTO 8762
8760 IF X1(T)=3 THEN GOSUB 8648
8762 NEXT T
8763 T=T9
8764 RETURN
8769 REM *** LABEL_PLOT:      ***
8770 T9=T
```

```
8772 FOR T=1 TO T9
8773 X=X(1,T) @ Y=X(2,T) @ Z=X(3
     ,T)
8774 IF X1(T)=4 THEN V$=X2$[T,T]
      @ GOSUB 8703
8776 NEXT T
8777 T=T9
8778 RETURN
8859 REM *TIC LENGTH*
8860 U1=0 @ GOSUB 8550
8862 GOSUB 8510 @ W5(1,1)=X @ W5
     (1,2)=Y
8864 U1=0 @ GOSUB 8550
8866 IF R=2 THEN X=X+1 ELSE Y=Y+
     1
8868 GOSUB 8220
8870 IF Z=0 THEN R2=1 ELSE R2=AB
     S(S2(3)/Z*.01)
8872 U1=0 @ GOSUB 8550
8874 IF R=2 THEN X=X+R2 ELSE Y=Y
     +R2
8876 GOSUB 8510
8878 W7=3*R2/((W5(1,1)-X)^2+(W5(
     1,2)-Y)^2)^.5
8880 RETURN
 1066
```