

2333 W. Huron J. ©
Chicago, IL 60612 Marselle
February 11, 1992
(312) 489-1832

Robert Fabric
3626 Morrie Drive
San Jose, CA 95127-9990

works @
IWAFF?

Dear Robert,

I finally was able to make a tape of the programs which I wrote to do line drawing and moving, scaling, and rotation. If you recall, I sent you a listing which was produced on a different computer, since I didn't have access to a dally printer. I think the xerox I enclose is probably as readable.

Let me recap what all the programs (actually subprograms; they all RETURN) do. The idea is that we store (X, Y, P) coordinates, *i.e.* LINE X, Y, P , ~~by~~ packing them into one integer @ array element.

This packing is done by PACK (entry line 900) and unpacked by UNPK (500). $@(0)$ contains the number of points to follow in $@(1), @(2), \dots, @(@(0))$.

The array can be loaded from an algorithm or by using LORAN (1500), which is a line drawing program.

$JX(1)$ and $JY(1)$ control the (x, y) coordinates of the rubber band line. $TR(1)$ is used to save the current (x, y) coordinates. If $KN(1) < 0$ when $TR(1)$ is pulled, $p = 0$. Otherwise $p = 1$. Press a key in column 1 of the keypad to delete the last drawn point, a key in column 2 to exit.

The picture may then be transformed by translation in the x & y direction, scaling up (can be modified to scale down but can't do both together) and various rotations. If rotation is to be done, sines & cosines are required. `COSSLO (1000)` was written by Dan Sandin when I took a course with him and uses `@(0) → @(90)` to hold the cosines of $0^\circ * 250$ to $90^\circ * 250$ (~~the~~ $* 250$ is for accuracy since no fractions in BB). Thus the vectors will need to use `@(91)`, ... and `LORAW` will need to be changed (in fact that's the way it is now). `COSSIN (700)` calculates the sines and cosines of its arg (any value in degrees, its normalized to $0 \rightarrow 360^\circ$).

Needless to say if any program is modified, the
@() array needs to be reloaded. ②

SETMOV(2500) interacts with the user to move a
picture and calls MOV(2000) to do the transformation.

SETSCALE(2200) does scaling and calls SCALE(2100).

ROT(2300) does rotation and calls XROT(1200),

YROT(1300), ZROT(1400), or XYROT(2400) to perform
rotations. Other combinations are possible (XZ, YZ, XYZ, XZY...)

The transformations are obtained from Newman &
Sproull's Principles of Interactive Computer Graphics,
2nd edition.

Now for the tape layout.

SIDE 1: Lines 497 (leader) - 2050 2 copies

SIDE 2: Lines 2097 - 2590 2 copies

Following are clean (almost no comments
or PRINT stmts) versions of UNPK, PACK, LDRAW,
SCALE, SETSCALE, MOV, SETMOV, COSSIN, COSLD,
XROT, YROT, ZROT, ROT and a
truncated XYROT (tape ran out).

Maybe some folks might make use of these programs and in the process learn about elementary computer graphics transformations.

Paul fan!

James P. Marselle

The transformations are obtained from Marselle's Principles of Interactive Computer Graphics, 2nd edition.

- Now for the tape layout.
- STEP 1: James POT (leader) - 2000 2 copies
 - STEP 2: James POT - 8250 2 copies
- Following are one clean (about no comments) or PRINT (about) versions of UNP, PAK, LPRN, SCALE, SETSCALE, MOV, SETMOV, GOSIN, GOSD, XROT, YROT, SPOT, ROT and a handwritten XYPOT (tape run out).