VB10/C INTERACTIVE GRAPHICS TERMINAL

Digital Equipment Corporation
July 1970

VB-10/C INTERACTIVE GRAPHICS TERMINAL

The VB-10/C graphics terminal is a highly interactive display system for the PDP-10. The VB-10/C system was designed to allow complete user freedom and flexibility under normal PDP-10 time-sharing. The basic hardware system consists of a 340/C display connected directly to PDP-10 memory through a special memory channel. Several important features included in the VB-10/C display are memory protection and relocation, slave mode operation, raster mode, and subroutining. To allow complete user-display interaction the VB-10/C graphics terminal has a function box, and a Tablet as options. Light Pen is standard, however, the important feature of the VB-10/C system is the extensive software package* available from DECUS to control the display. This package permits a user not familiar with the displays hardware instructions to write display programs in a higher level language (FORTRAN or LISP).

The VB-10/C software package contains all of the basic routines for displaying vectors, points, and text, and for controlling the scale and intensity of the picture. The user can define his own number space or sets of number spaces and the viewpoints on the display where the pictures are to be displayed. Pictures can be generated as a set of subpictures or nested subpictures. Any part of a picture which falls outside of the user's number space or window will not be displayed. Subpictures and pictures can be blanked from the display and unblanked at a later time. Core space used by a picture can be released and refused for generating new pictures. The VB-10/C package also contains all of the routines necessary to create three-dimensional pictures. The user can rotate the picture by changing the pitch, yaw, and roll coefficients of the picture or zoom-in or back off from a picture by changing the viewpoint value.

The VB-10/C graphics terminal provides several levels of user interaction. Simple data inputs can be entered through the function box or the teletype. The identification of lines or points within a picture can be accomplished with the light pen. However, the most important interactive feature of the VB-10/C terminal is that a user can input graphical data through the Tablet. An architect can draw new structures, a mathematician can input new curves, an eletrical engineer can add new connections during circuit design, all with no more difficulty than drawing on a piece of paper. The VB-10/C graphics terminal can solve almost all complex graphics requirements with hardware and software that has already been developed and is running in production enviroments today.

The following description of VB10/C software is followed by an explanation of the Tablet and its interactive capability with the user's program. Pictures illustrating VB10/C output are distributed throughout the text.

*Developed at National Institutes of Health, Division of Computer Research and Technology by Harry R. Lewis

FORTRAN-LISP DISPLAY ROUTINES

BY

HARRY R. LEWIS

****** SEE IMPORTANT NEW INFORMATION ON PAGE 20 *********

TABLE OF CONTENTS
##################

I.	INTRODUCTION	1
II.	PICTURES	2
III.	WINDOWS AND VIEWPORTS	3
IV.	A NOTE ON DATA TYPES	4
V.	INITIALIZATION	5
V.1	FORTRAN INITIALIZATION PROCEDURE	
V.2	LISP INITIALIZATION PROCEDURE	
V.3	ESTIMATING SPACE FOR THE DISPLAY FILE	
VI.	SETTING THE WINDOW AND VIEWPORT	6
VII.	PICTURE GENERATION	7
VII.1	FORTRAN GENERATION	
VII.2	LISP GENERATION	
VIII.	SCALE AND INTENSITY	8
IX.	PLOTTING AND SETTING POINTS	9
X.	PLOTTING VECTORS	10
XI.	TEXT	11
XI.1	FORTRAN TEXT	
XI.2	LISP TEXT	
XII.	THE LIGHT PEN	12
XIII.	MISCELLANEOUS PICTURE-HANDLING ROUTINES	13
XIII.1	BLANK, UNBLANK, AND DESTROY	
XIII.2	SUBPICTURE	
XIII.3	REPLACE	
XIII.4	COMPIC	
XIV.	MISCELLANEOUS ROUTINES	14
XIV.1	SYNC AND NOSYNC	
XIV.2	DISEND	
XV.	SPECIAL FORTRAN ROUTINES	15
XV.1	RASTER MODE	
XV.2	THREE-DIMENSIONAL DISPLAY ROUTINES	
XVI.	ERROR MESSAGES	
XVII.	DYNAMIC TECHNIQUE	
XVIII.	ASSEMBLY AND LOADING	20
XIX.	VERSION-DEPENDENT FEATURES	21
XX.	SAMPLE PROGRAMS	22
APPENDIX:	LISTING OF ERROR MESSAGES	25

I. INTRODUCTION

THIS DOCUMENT DESCRIBES A GRAPHICAL LANGUAGE FOR THE PDP-10/340 COMPUTER DISPLAY SYSTEM. THE LANGUAGE IS IMPLEMENTED AS A SET OF SUBROUTINES WHICH MAY BE CALLED FROM ASSEMBLY CODE, FROM AN INTERPRETED LIST-PROCESSING LANGUAGE (LISP), OR FROM A COMPILED ALGEBRAIC LANGUAGE (FORTRAN). THESE SUBROUTINES GENERATE BINARY PROGRAMS FOR THE 340 DISPLAY FROM DATA SUPPLIED AS ARGUMENTS; THEY ALSO INCLUDE OPERATIONS TO POST, ADD, AND DELETE INFORMATION FROM A DISPLAY FILE.

SEVERAL GOALS DIRECTED THE DESIGN OF THE LANGUAGE. FIRST, SINCE THE 340 IS A COMPUTER, A GREAT DEAL OF PATIENCE AND PROGRAMMING EXPERIENCE IS NEEDED TO WRITE ASSEMBLY OR ABSOLUTE CODE FOR IT; THE DIFFICULTIES ARE PARTICULARLY ACUTE SINCE THE 340 HAS A POOR INSTRUCTION SET AND NO DEBUGGING AIDS ARE AVAILABLE FOR IT.

SECOND, THE DESIGN OF THE 340 AND THE PDP-10/340 INTERFACE MAKE DYNAMIC CHANGING, ADDING, AND DELETING OF DISPLAY INFORMATION A RATHER TRICKY PROCESS; IT IS CONVENIENT TO HAVE A LANGUAGE TO HANDLE THESE ANNOYING DETAILS.

THIRD, EFFICIENT USE OF MEMORY SPACE IN A PROGRAM WHICH DYNAMICALLY CHANGES A DISPLAY FILE REQUIRES THAT THE PROGRAM WHICH GENERATES A NEW DISPLAY FILE MAKE UWE OF SPACE OCCUPIED BY ABANDONED DISPLAY FILES. SINCE A DISPLAY FILE IS A BINARY PROGRAM, IT CANNOT EASILY BE RELOCATED, SO THE GARBAGE COLLECTION PROBLEM IS DIFFICULT AND ANNOYING.

FOURTH, THE FACE OF THE CRT IS OF FINITE SIZE, WHILE EUCLIDEAN GEOMETRICAL SPACE IS INFINITE. HENCE, THE SCOPE MUST ACT AS A WINDOW ON THE ACTUAL DATA SPACE, AND THE CODE GENERATING PROGRAM MUST SELECT FOR DISPLAY ONLY THE INFORMATION WHICH COULD BE "SEEN" THROUGH THIS WINDOW. A RELATED PROBLEM IS THAT THE SCOPE HAS A FIXED, DISCRETE COORDINATE SYSTEM, WHILE AN APPLICATION PROGRAM MAY WISH TO DEAL IN ANGSTROMS, MILES, OR SOME OTHER UNIT. THE CHANGE OF COORDINATES SHOULD BE IMPLICIT.

FINALLY, THE SUBROUTINES WERE MADE COMPATIBLE WITH HIGHER-LEVEL LANGUAGES SO THAT THE PROCESSING POWER OF THESE HOST LANGUAGES COULD BE USED TO DEAL WITH THE NON-GRAPHIC ASPECTS OF THE PROGRAM.

II. PICTURES

THE BASIC UNIT OF DISPLAY INFORMATION IS CALLED A "PICTURE" OR "FRAME." FIVE BASIC OPERATIONS MAY BE PERFORMED ON A PICTURE: GENERATE, BLANK, UNBLANK, DESTROY, AND SUBPICTURE.

GENERATE	IS THE PROCESS WHICH CREATES A PICTURE.
BLANK	REMOVES A PICTURE FROM VIEW, WHILE KEEPING IT AVAILABLE FOR LATER USE.
UNBLANK	RESTORES A PREVIOUSLY BLANKED PICTURE. AFTER GENERATION, A PICTURE IS AUTOMATICALLY UNBLANKED.
DESTROY	REMOVES A PICTURE FROM VIEW AND ANNIHILATES IT.
SUBPICTURE	(USED AS PART OF THE GENERATION PROCESS FOR ANOTHER PICTURE) MAKES A PICTURE PART OF ANOTHER PICTURE. SUBPICTURES CAN BE NESTED TO ANY LEVEL.

PICTURES ARE OF TWO KINDS: TOP-LEVEL OR COMPONENT. FOR TOP-LEVEL PICTURES, THE BLANK AND UNBLANK OPERATIONS AFFECT THE PICTURE'S VISIBILITY DIRECTLY. FOR COMPONENT PICTURES, THESE OPERATIONS AFFECT VISIBILITY ONLY IF A PICTURE OF WHICH THE PICTURE IS A PART IS UNBLANKED. A FEW EXAMPLES WILL MAKE THIS CLEARER. IN EACH CASE, ASSUME THAT NO OTHER PICTURES EXIST.

EXAMPLE 1. PICTURE A IS COMPONENT AND UNBLANKED; OR COMPONENT AND BLANKED; OR TOP-LEVEL AND BLANKED. THEN A WILL BE INVISIBLE.

EXAMPLE 2. PICTURE A IS TOP-LEVEL AND UNBLANKED. BICTURE B IS TOP-LEVEL AND A SUBPICTURE OF A. THEN IF B IS BLANKED, ONLY A WILL BE VISIBLE. IF B IS UNBLANKED, A WILL BE VISIBLE AND TWO SUPERIMPOSED COPIES OF B WILL BE VISIBLE. THAT IS, IN THE LATTER CASE, BOTH A AND B WILL BE VISIBLE, BUT B WILL BE BRIGHTER THAN A.

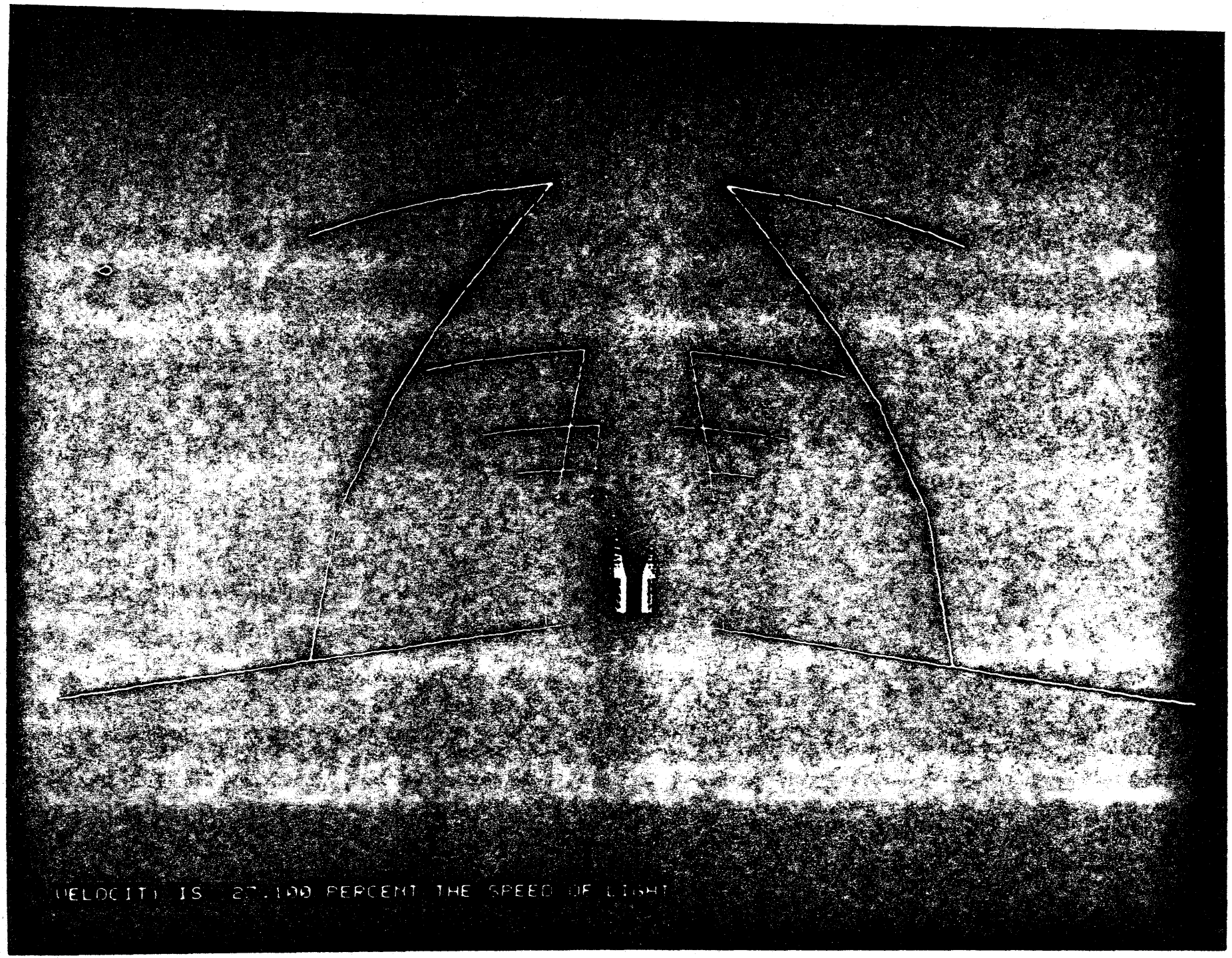
EXAMPLE 3. A IS TOP-LEVEL; B IS A COMPONENT SUBPICTURE OF A; C IS A COMPONENT SUBPICTURE OF B. THEN C WILL BE VISIBLE IF AND ONLY IF A, B, AND C ARE ALL UNBLANKED; B WILL BE VISIBLE IF AND ONLY IF A AND B ARE BOTH UNBLANKED; AND A WILL BE VISIBLE IF AND ONLY IF IT IS UNBLANKED.

III. WINDOWS AND VIEWPORTS

THE USER'S DATA SPACE SHOULD BE EQUIPPED WITH A TWO-DIMENSIONAL CARTESIAN COORDINATE SYSTEM. BEFORE GENERATING THE FIRST PICTURE, HE MUST SPECIFY THE PORTION OF THE PLANE WHICH WILL BE SHOWN ON THE SCOPE. THIS PORTION OF THE DATA PAGE IS THE WINDOW, AND MUST BE A RECTANGLE WHOSE EDGES ARE PARALLEL TO THE USER'S COORDINATE AXES. NO INFORMATION OUTSIDE THE WINDOW WILL BE SHOWN. FOR EXAMPLE, IF THE USER SPECIFIES A LINE WHICH CROSSES SOME EDGE OF THE WINDOW, ONLY THAT PORTION OF THE LINE INSIDE THE WINDOW WILL BE SHOWN.

THE ENTIRE SCOPE FACE NEED NOT BE USED TO SHOW A PICTURE. FOR EXAMPLE, IT MAY BE DESIRABLE TO SHOW FOUR PICTURES, EACH OCCUPYING A QUADRANT OF THE SCREEN. THE PORTION OF THE SCREEN INTO WHICH THE WINDOW WILL BE MAPPED IS CALLED THE VIEWPORT AND MUST BE A RECTANGLE WHOSE EDGES ARE PARALLEL TO THE FIXED CRT COORDINATE AXES. THE TRANSFORMATION WHICH MAPS THE ENTIRE WINDOW AREA INTO THE ENTIRE VIEWPORT AREA IS LINEAR; E.G., A POINT IN THE CENTER OF THE WINDOW WILL BE MAPPED INTO A POINT IN THE CENTER OF THE VIEWPORT.

THE WINDOW AND VIEWPORT MAY BE CHANGED DURING THE GENERATION PROCESS FOR A PICTURE OR BETWEEN PICTURES.


VELOCITY IS 27.100 PERCENT THE SPEED OF LIGHT

IV. A NOTE ON DATA TYPES

MOST SUBROUTINE ARGUMENTS MAY BE EITHER FIXED OR FLOATING POINT NUMBERS; THE SUBROUTINES WILL CORRECT EACH TYPE TO THE OTHER WHEN NECESSARY FOR THEIR OWN INTERNAL PURPOSES. THE ONLY EXCEPTIONS ARE: IN LISP, THE NAMES OF PICTURES MUST BE ATOMS; AND IN FORTRAN, THE NAMES OF PICTURES AND THE DISPLAY FILE MUST BE ARRAYS (OF ANY TYPE).

Here we see a roadway lined with telephone poles. Because the observer is traveling at a rate which is a significant fraction of the speed of light, the poles appear to be bent. The observer's eye is at street level.

V. INITIALIZATION

(FOR LOADING PROCEDURES, SEE SECTION XVIII.)

V.1 FORTRAN INITIALIZATION PROCEDURE

IN FORTRAN, INITIALIZATION REQUIRES THAT FOUR TYPES OF STATEMENTS BE PRESENT IN THE PROGRAM.

V.1.1. AN ARRAY OF SUFFICIENT SIZE TO HOLD THE DISPLAY FILE MUST BE DIMENSION'ED. REFER TO V.3 TO ESTIMATE THE REQUIRED SIZE.

V.1.2. AN ARRAY MUST BE DIMENSION'ED FOR EACH PICTURE TO BE GENERATED. THE LENGTH OF AN ARRAY MUST BE AT LEAST THE LARGER OF 3 AND N/2 WHERE N IS THE NUMBER OF SUBPICTURE REFERENCES TO THE PICTURE. HERE N/2 IS ROUNDED UP (I.E. [N/2]+1 FOR N ODD, N/2 FOR N EVEN). THE NAME OF THE ARRAY IS THE NAME BY WHICH THE PICTURE WILL BE REFERRED TO. THESE ARRAYS MAY NOT BE USED FOR ANY OTHER PURPOSE.

V.1.3. BEFORE CALLING ANY OTHER DISPLAY ROUTINES, THE PROGRAM MUST CALL DISPINI(ARRAY,LENGTH) WHERE "ARRAY" IS THE NAME OF THE ARRAY REFERRED TO IN V.1.1 AND "LENGTH" IS ITS SIZE IN WORDS. THIS ROUTINE SHOULD NOT BE CALLED AGAIN; IT COMPLETELY INITIALIZES THE DISPLAY FILE.

V.1.4. BEFORE GENERATING A PICTURE, THE PROGRAM MUST CALL DCLPIC(NAME,LENGTH), WHERE "NAME" IS THE NAME OF THE PICTURE ARRAY AS DESCRIBED IN V.1.2 AND "LENGTH" IS ITS SIZE IN WORDS. THIS ROUTINE SHOULD BE CALLED ONLY ONCE FOR EACH PICTURE; IT NEED NOT BE CALLED AGAIN IF A PICTURE IS DESTROYED AND REGENERATED.

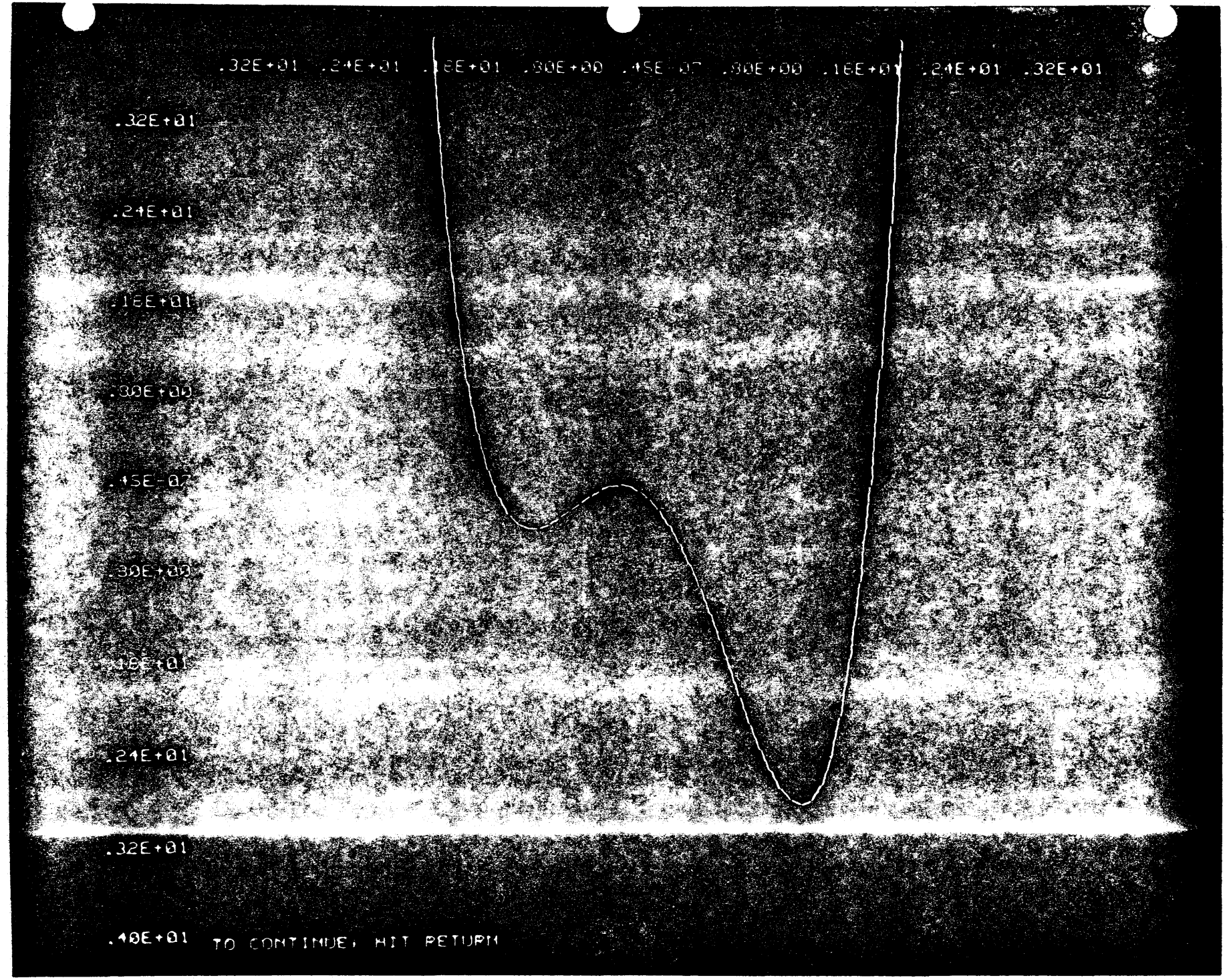
V.2 LISP INITIALIZATION PROCEDURE

BEFORE CALLING ANY OTHER DISPLAY ROUTINE, THE PROGRAM SHOULD CALL (DISPINIT). THIS CALL SHOULD NOT BE REPEATED.

V.3 ESTIMATING SPACE FOR THE DISPLAY FILE

WHEN USED WITH LISP, THE DISPLAY ROUTINES TAKE MEMORY SPACE FOR THE DISPLAY FILE FROM BINARY PROGRAM SPACE, SO THE USER MUST LEAVE SUFFICIENT ROOM AT LOAD AND ALLOC TIME. WITH FORTRAN, ROOM MUST BE ALLOCATED BY A DIMENSION STATEMENT AND A CALL TO DSPINI. TO ESTIMATE THE NEEDED ROOM, ALLOW

1 WORD FOR EACH POINT (PGEN)
1 WORD FOR EACH 2 SHORT VECTORS (VGEN)
1 WORD FOR EACH 6 CHARACTERS
1 WORD FOR EACH 5 RASTER POINTS FOR RASTER MODE DISPLAY
1 WORD FOR EACH TOP-LEVEL PICTURE
AND ABOUT 10% OF THE ABOVE TOTAL FOR OVERHEAD.


.32E+01 .24E+01 .18E+01 .30E+00 .45E-07 .30E+00 .18E+01 .24E+01 .32E+01
.32E+01
.24E+01
.18E+01
.30E+00
.45E-07
.30E+00
.18E+01
.24E+01
.32E+01
.40E+01 TO CONTINUE, HIT RETURN

VI. SETTING THE WINDOW AND VIEWPORT

FORTRAN:
```
CALL SETWIN (XMIN,XMAX,YMIN,YMAX)
CALL SETPOR (XMIN,XMAX,YMIN,YMAX)
```

LISP:
```
(SETWIN XMIN XMAX YMIN YMAX)
(SETPOR XMIN XMAX YMIN YMAX)
```

THE FOUR ARGUMENTS REFER RESPECTIVELY TO THE COORDINATES OF THE LEFT, RIGHT, BOTTOM, AND TOP EDGES OF THE RECTANGLE. IN ALL CASES, "XMIN" MUST BE STRICTLY LESS THAN "XMAX" AND "YMIN" STRICTLY LESS THAN "YMAX." THE WINDOW COORDINATES MAY BE IN ANY NUMERICAL RANGE; THE VIEWPORT COORDINATES MUST BE BETWEEN 0 AND 1023 DECIMAL, INCLUSIVE.

This graph was plotted by a rather interesting demonstration program. The user need only specify the range of X and Y, the degree of the equation to be plotted, and the coefficient of the terms.

VII. PICTURE GENERATION

THIS IS THE PROCESS WHICH CREATES A PICTURE. A PICTURE WHICH HAS BEEN GENERATED CANNOT BE GENERATED AGAIN UNTIL IT HAS BEEN DESTROYED. A SECOND PICTURE CANNOT BE GENERATED DURING THE GENERATION PROCESS FOR A PICTURE. AT THE END OF THE GENERATION PROCESS, THE PICTURE IS ENTERED AS A TOP-LEVEL PICTURE IN THE DISPLAY FILE AND UNBLANKED.

VII.1 FORTRAN GENERATION

THE GENERATION PROCESS IS BEGUN BY

```
CALL GENINI (PICNAME)
```

WHERE "PICNAME" IS THE NAME OF AN ARRAY OF THE TYPE DESCRIBED IN V.1.2. THE PROCESS IS TERMINATED BY

```
CALL GENEND (PICNAME).
```

BETWEEN THESE CALLS ARE THE CODE-GENERATING STATEMENTS WHICH DESCRIBE THE PICTURE. ANY NON-DISPLAY STATEMENT MAY BE EXECUTED DURING THE GENERATION PROCESS.

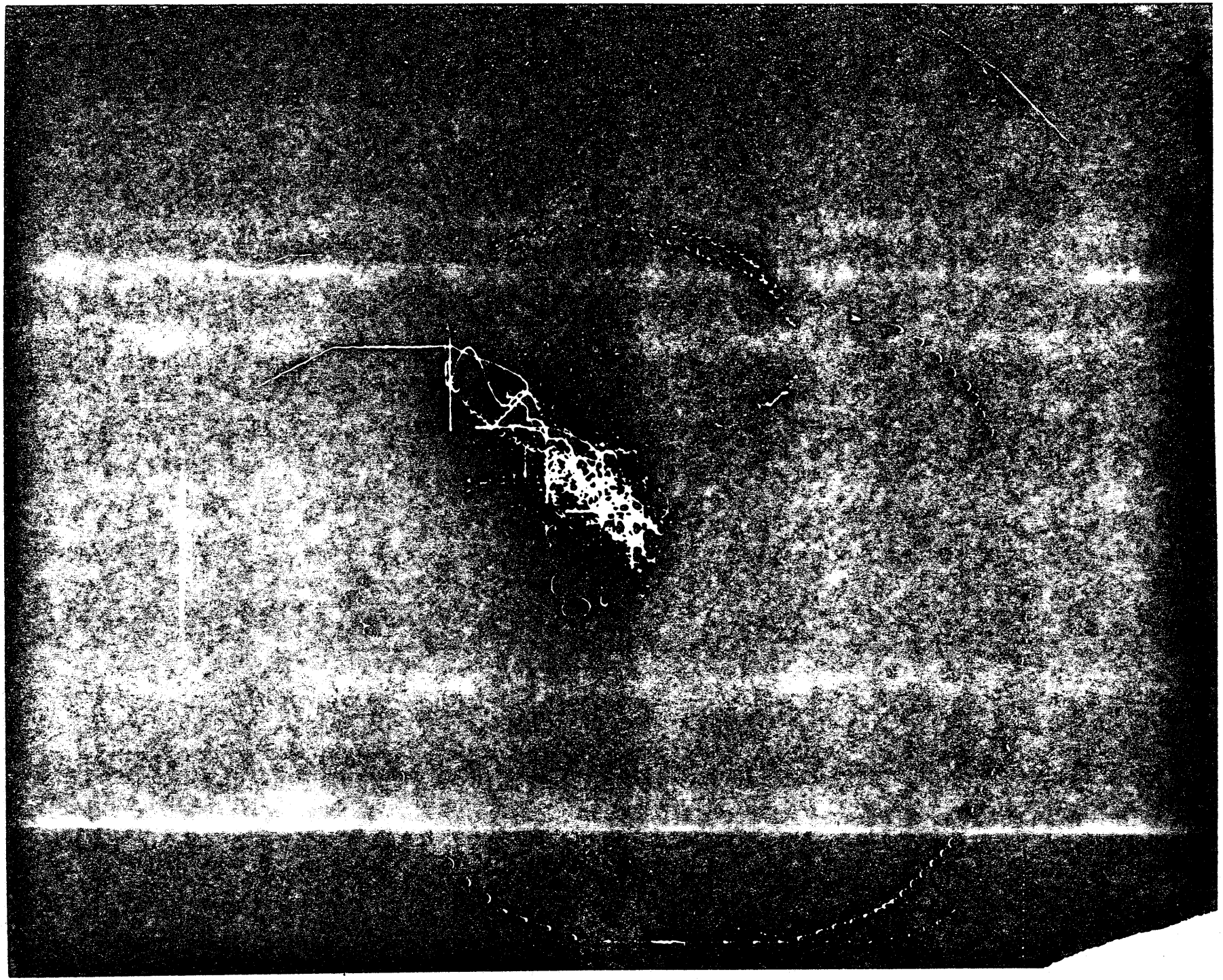
VII.2 LISP GENERATION

THE GENERATION PROCESS OCCURS DURING THE EVALUATION OF A SINGLE S-EXPRESSION WHICH IS OF THE FORM

```
(GENERATE X E1 E2 ... EN).
```

HERE "X" MUST BE AN ATOM AND IS THE NAME BY WHICH THE PICTURE WILL BE REFERRED TO. THE "E" ARGUMENTS ARE S-EXPRESSIONS WHICH ARE SUCCESSIVELY EVALUATES. SOME OF THESE S-EXPRESSIONS ARE THE CODE-GENERATING STATEMENTS WHICH DESCRIBE THE PICTURE; OTHERS MAY BE S-EXPRESSIONS WHICH DO NOT REFER TO DISPLAY ROUTINES.

THE VALUE RETURNED IS "X".



VIII. SCALE AND INTENSITY

THE DISPLAY HARDWARE IS CAPABLE OF DRAWING INCREMENTAL DATA IN FOUR "SCALES," 1, 2, 4, OR 8. THIS VALUE GOVERNS THE SPACING BETWEEN INCREMENTALLY PLOTTED POINTS IN THE DISPLAY OF LINE SEGMENTS AND CHARACTERS; 1 IS THE SMALLEST SCALE, IN SCALE 2 THE POINTS ARE TWICE AS FAR APART, ETC. FOR THE PURPOSES OF THESE DISPLAY ROUTINES, THE SCALE VALUE IS USED ONLY TO CONTROL THE SPACING BETWEEN POINTS WHEN DRAWING LINE SEGMENTS (NOT THEIR SIZE), BUT CHANGES BOTH THE SIZE AND POINT SPACING OF CHARACTERS.

THE DISPLAY HARDWARE ALSO HAS THE FACILITY FOR DISPLAYING DATA AT ANY OF 8 INTENSITIES, 0-7, 0 BEING THE DIMMEST AND 7 THE BRIGHTEST.

BOTH SCALE AND INTENSITY ARE ESTABLISHED DURING GENERATION BY THE PARMS ROUTINE.

```
FORTRAN:        CALL PARMS (SCALE,INTENSITY)

LISP:           (PARMS SCALE INTENSITY)
```

The "Neutron of *Minsky" is a fascinating toy which moves three points with the movement of any one dependent on the motions of the other two.

IX. PLOTTING AND SETTING POINTS

THE POINT-GENERATING ROUTINE SETS AN IMAGINARY "BEAM" AT A SPECIFIED POSITION ON THE DATA PAGE. IF THAT POINT IS WITHIN THE WINDOW IT MAY BE INTENSIFIED, I.E., SHOWN; OTHERWISE, THE ROUTINE MERELY ESTABLISHES A POSITION RELATIVE TO WHICH DATA CAN BE SPECIFIED.

FORTRAN: CALL PGEN (X,Y,INTENSIFY)

LISP: (PGEN X Y INTENSIFY)

THE ARGUMENTS "X" AND "Y" SPECIFY THE COORDINATES OF THE POINT. THE "INTENSIFY" ARGUMENT SHOULD BE Ø IF THE POINT IS NOT TO BE INTENSIFIED, NON-ZERO IF IT IS TO BE INTENSIFIED.

X. PLOTTING VECTORS

A LINE SEGMENT IS SPECIFIED BY GIVING THE POSITION OF ITS END POINT RELATIVE TO ITS STARTING POINT. THE POSITION OF THE STARTING POINT IS IMPLICIT, HAVING BEEN SET UP BY A CALL TO "PGEN" OR "VGEN." FOR EXAMPLE, TO DRAW A LINE FROM (2,3) TO (-4,7), ONE WOULD POSITION THE BEAM AT (2,3) BY A "PGEN" AND THEN DRAW A VECTOR WITH ARGUMENTS (-6,4). VECTORS, LIKE POINTS, MAY BE EITHER INTENSIFIED OR NOT.

FORTRAN: CALL VGEN (DELTAX,DELTAY,INTENSIFY)

LISP: (VGEN DELTAX DELTAY INTENSIFY)

AS WITH "PGEN," "INTENSIFY" IS NON-ZERO TO INTENSIFY, AND ZERO NOT TO INTENSIFY.

XI. TEXT

THE DISPLAY PACKAGE INCLUDES SUBROUTINES TO DISPLAY TEXT AT ANY OF THE FOUR SIZES AT ARBITRARY POSITIONS ON THE SCOPE. THESE ROUTINES ARE GOVERNED BY THE WINDOW AND VIEWPORT PARAMETERS FOR THE PURPOSES OF CLIPPING -- TEXT WILL NOT BE SHOWN OUTSIDE THE SPECIFIED VIEWPORT -- BUT NOT FOR SCALING -- I.E., THE SHAPE OF A CHARACTER WILL NOT BE DIFFERENT IF THE PROPORTIONS OF THE WINDOW RECTANGLE ARE DIFFERENT FROM THOSE OF THE VIEWPORT RECTANGLE. SINCE THESE ROUTINES TAKE ADVANTAGE OF THE TEXT-FORMATTING CAPABILITIES OF THE HOST LANGUAGE, THEY MAY NOT BE USED IF THE ROUTINES ARE USED WITHOUT A HOST LANGUAGE, I.E. IF THEY ARE USED ONLY WITH ASSEMBLY CODE.

THE POSITION OF THE FIRST CHARACTER OF A STRING OF TEXT IS THE LAST POINT SPECIFIED DIRECTLY BY "PGEN" OR IMPLICITLY BY "VGEN." IF THIS POINT IS OUTSIDE THE WINDOW, NO TEXT WILL BE SHOWN. OTHERWISE, THIS POINT IS USED TO DETERMINE THE POSITION OF THE "LEFT MARGIN" IN CASE OF A CARRIAGE RETURN.

XI.1 FORTRAN TEXT

TEXT DISPLAY IS INITIATED BY

```
CALL TEXTP.
```

AFTER THIS CALL, ALL FORTRAN "TYPE ..." STATEMENTS WILL CAUSE THEIR OUTPUT TO BE CHANNELLED TO THE DISPLAY FILE RATHER THAN TO THE TELETYPE, UNTIL ANOTHER GENERATING ROUTINE (PGEN, VGEN, GENEND, ETC.) IS CALLED. AFTER CALLING ONE OF THESE ROUTINES, THE OUTPUT OF "TYPE ..." STATEMENTS WILL AGAIN APPEAR ON THE TELETYPE.

TEXT WILL APPEAR ON THE DISPLAY EXACTLY AS IT WOULD APPEAR ON THE TELETYPE; FORMAT SPECIFIERS ARE INTERPRETED THE SAME WAY, CARRIAGE CONTROL IS RECOGNIZED, ETC. THE ONLY DIFFERENCES ARE THAT LINES TOO LONG TO FIT IN THE WINDOW WILL BE CLIPPED -- THE PART WHICH WOULD BE OUTSIDE THE WINDOW WILL NOT BE SHOWN; AND LINES WHICH WOULD APPEAR BELOW THE BOTTOM OF THE WINDOW WILL NOT BE SHOWN.

XI.2 LISP TEXT

TEXT IS GENERATED BY

```
(TEXT ...).
```

THIS ROUTINE COMPLETELY MIMICS "(PRINT ...)" IN THE WAY IT EVALUATES ITS ARGUMENTS AND FORMATS ITS OUTPUT. THE ROUTINE MAY BE USED FOR DISPLAYING QUOTED TEXT; I.E. "(TEXTQ A B C)" HAS THE SAME EFFECT AS "(TEXT (QUOTE (A B C)))."

THESE ROUTINES ADJUST THE LENGTH OF THE "PRINTING" LINE SO THAT NO TEXT WILL BE LOST OFF THE RIGHT-HAND EDGE OF THE WINDOW; BUT THEY WILL CLIP OFF TEXT WHICH WOULD APPEAR BELOW THE BOTTOM OF THE WINDOW.

XII. THE LIGHT PEN

A LIMITED FACILITY FOR THE USE OF THE LIGHT PEN HAS BEEN PROVIDED. THE LIGHT PEN MUST BE "TURNED OF AND OFF" WITHIN THE DISPLAY FILE IN ORDER TO MAKE CERTAIN PARTS OF THE PICTURE LIGHT-PEN SENSITIVE. ITS INITIAL STATUS, I.E. ITS STATUS AT THE BEGINNING OF THE DISPLAY FILE, IS OFF. SINCE THE USER HAS NO CONTROL OVER THE ORDER IN WHICH PICTURES ARE DISPLAYED, HE SHOULD SPECIFY IN EACH PICTURE THE STATUS OF THE PEN, IF HE USES IT AT ALL.

DURING THE GENERATION PROCESS, THE CODE TO ENABLE THE LIGHT PEN WILL BE GENERATED BY

FORTRAN: CALL LPON

LISP: (LPON)!

THE CODE TO TURN OFF THE LIGHT PEN BY

FORTRAN: CALL LPOFF

LISP: (LPOFF).

AFTER GENERATION, THE PROGRAM CAN GET THE SCOPE COORDINATES OF THE POINT MOST RECENTLY "SEEN" BY THE LIGHT PEN BY CALLING THE ROUTINE "LPIN." IF A LIGHT PEN HIT HAS NOT YET OCCURRED WHEN THIS ROUTINE IS CALLED, THE ROUTINE WILL HANG AND WAIT FOR ONE BEFORE RETURNING.

FORTRAN: CALL LPIN (I)

WHERE I IS AN ARRAY DIMENSIONED TO LENGTH 2,
THE ROUTINE WILL RETURN WITH

I(1) = X COORDINATE
I(2) = Y CORRDINATE.

LISP: (LPIN)

THE VALUE RETURNED IS A "CONS" PAIR P, WITH

(CAR P) = X
(CDR P) = Y.

XIII. MISCELLANEOUS PICTURE-HANDLING ROUTINES

XIII.1 BLANK, UNBLANK, AND DESTROY

```
FORTRAN:        CALL BLANK (PIC)
                CALL UNBLANK (PIC)
                CALL DESTROY (PIC)

LISP:           (BLANK (QUOTE PIC))
                (UNBLANK (QUOTE PIC))
                (DESTROY (QUOTE PIC))
```

THESE ROUTINES PERFORM THE FUNCTIONS OF BLANKING, UNBLANKING, AND DESTROYING PICTURES, AS DESCRIBED IN SECTION II. THEY SHOULD NOT BE CALLED DURING THE GENERATION PROCESS FOR A PICTURE.

XIII.2. SUBPICTURE

```
FORTRAN:        CALL SPICT (PICNAME)

LISP:           (SUBPIC (QUOTE PICNAME))
```

WHEN CALLED DURING THE GENERATION PROCESS FOR A PICTURE, THIS ROUTINE MAKES THE PICTURE "PICNAME" A SUBPICTURE OF THE PICTURE BEING GENERATED. THE PICTURE "PICNAME" MUST EXIST AT THE TIME THIS CALL IS MADE.

XIII.3. REPLACE

```
FORTRAN:        CALL REPLACE (PIC1,PIC2)

LISP:           (REPLACE (QUOTE PIC1) (QUOTE PIC2))
```

THIS ROUTINE CAUSES ALL REFERENCES TO "PIC1" TO BE REPLACED BY REFERENCES TO "PIC2." THIS ROUTINE IS USEFUL WHEN IT IS DESIRED TO CHANGE A SUBPICTURE WITHOUT REGENERATING ALL PICTURES OF WHICH IT IS A PART.

XIII.4. COMPIC

```
FORTRAN:        CALL COMPIC (PIC)

LISP:           (COMPIC (QUOTE PIC))
```

THIS CALL MAKES "PIC" A COMPONENT PICTURE.

XIV. MISCELLANEOUS ROUTINES

XIV.1 SYNC AND NOSYNC

NORMALLY A FLAG IS SET AT THE BEGINNING OF THE DISPLAY FILE WHICH CAUSES THE DISPLAY PROCESSOR ALWAYS TO TAKE AN EVEN MULTIPLE OF 1/60 SECOND TO DISPLAY THE DISPLAY FILE. THIS CAUSES UNIFORM BRIGHTNESS TO BE MAINTAINED WHEN SHORT FILES ARE BEING DISPLAYED AND ACTS AS A FAILSAFE AGAINST BURNING A HOLE IN THE PHOSPHOR OF THE CRT SHOULD THE DISPLAY FILE CONSIST ONLY OF A SINGLE INTENSIFIED POINT. IF THE DISPLAY FILE IS LONG, HOWEVER, THIS "SYNCING" MAY RESULT IN FLICKER WHICH WOULD NOT OCCUR OF THE FLAG WERE NOT SET. ROUTINES HAVE BEEN PUT IN THE DISPLAY PACKAGE TO PERMIT USERS TO CONTROL THE STATE OF THIS FLAG. THESE ROUTINES SHOULD BE USED JUDICIOUSLY!

TO SET THE "SYNC" BIT,

FORTRAN: CALL SYNC

LISP: (SYNC),

TO CLEAR THE "SYNC" BIT,

FORTRAN: CALL NOSYNC

LISP: (NOSYNC),

XIV.2 DISEND

FORTRAN: CALL DISEND

LISP: (DISEND)

THIS ROUTINE RELEASES THE DISPLAY AND ALLOWS THE JOB TO BE SWAPPABLE. THE DISPLAY CANNOT BE TURNED ON AGAIN WITHOUT REINITIALIZING THE ENTIRE PACKAGE.

XIV.3 DISTRT

FORTRAN: CALL DISTRT

LISP: (DISTRT)

THIS ROUTINE RESTARTS THE DISPLAY AFTER A CALL TO "DISEND". THE APPEARANCE OF THE DISPLAY AND THE INTERNAL DATA STRUCTURES ARE EXACTLY AS THEY WERE BEFORE "DISEND" WAS CALLED.

XV. SPECIAL FORTRAN ROUTINES

ROUTINES ARE AVAILABLE IN THE FORTRAN VERSION OF THE PACKAGE TO HANDLE TWO SPECIAL DISPLAY APPLICATIONS NOT AVAILABLE TO THE LISP USER.

XV.1 RASTER MODE

A PATCH TO BE DISPLAYED IN RASTER MODE IS SPECIFIED BY GIVING THE PACKED GRAY-LEVEL BYTES IN AN ARRAY. THE GRAY-LEVEL BYTES MAY BE ONE, TWO, OR THREE BITS LONG. THE INTENSITIES CORRESPONDING TO THE SPECIFIED GRAY LEVELS ARE AS FOLLOWS:

	BYTE SIZE		
	1	2	3
GRAY LEVEL			
0	0	1	0
1	7	3	1
2		5	2
3		7	3
4			4
5			5
6			6
7			7

THE RASTER BYTES ARE GIVEN IN THE ORDER OF A LEFT-TO-RIGHT, THEN TOP-TO-BOTTOM SCAN OF THE AREA.

THE RASTER-GENERATING ROUTINE IS INVOKED BY

```
CALL RASTER (ARRAY,X0,Y0,XSIZE,YSIZE,C,I,B),
```

WHERE

ARRAY = NAME OF ARRAY CONTAINING GRAY-LEVEL BYTES
X0,Y0 = SCOPE COORDINATES OF UPPER-LEFT HAND CORNER OF RASTER PATCH
XSIZE,YSIZE = NUMBER OF GRAY-LEVEL BYTES IN THE HORIZONTAL AND VERTICAL DIRECTIONS IN THE PATCH
C = 1 TO DISPLAY INTENSITY X FROM THE TABLE AS INTENSITY 7-X ("COMPLEMENT")
0 TO DISPLAY NORMALLY
I = 1 TO DISPLAY THE PATCH IN TWO INTERLEAVED PASSES THROUGH THE DATA
0 TO DISPLAY NORMALLY
B = BYTE SIZE OF THE BYTES IN "ARRAY."

THE CLIPPING PROCESS DOES NOT APPLY TO RASTER MODE DISPLAY.

XV,2 THREE-DIMENSIONAL DISPLAY ROUTINES

****

ROUTINES ARE PROVIDED FOR PLOTTING CLIPPED, PARALLEL OR POINT PROJECTIONS OF POINTS AND LINES SPECIFIED BY THREE-DIMENSIONAL CARTESIAN COORDINATES. THE THREE-DIMENSIONAL TRANSLATION, ROTATION, AND PERSPECTIVE OF THE DATA IS GOVERNED BY A SINGLE 4X4 HOMOGENEOUS MATRIX WHICH SHOULD BE IN A COMMON AREA "T3" ESTABLISHED IN THE USER'S PROGRAM.

IF NO ROTATION OR TRANSLATION IS IN EFFECT, THE "X" AND "Y" AXES OF THE USER'S THREE-DIMENSIONAL DATA WILL BE THE X AND Y AXES OF THE USER'S DATA PAGE AS ESTABLISHED BY THE "SETWIND" ROUTINE. THE Z AXIS WILL BE PERPENDICULAR TO THE X AND Y AXES AND DIRECTED OUT OF THE SCOPE (RIGHT-HANDED SYSTEM). ROTATIONS ARE SPECIFIED BY PITCH (AROUND THE Z AXIS), ROLL (AROUND THE Y AXIS), AND YAW (AROUND THE X AXIS). THAT IS, TO GET A DIFFERENT VIEW OF HIS DATA THE USER CHANGES NOT THE ACTUAL COORDINATES OF HIS DATA BUT THE VALUE OF THE MATRIX; THEN AT GENERATION TIME A DIFFERENT VIEW WILL BE PRODUCED.

TO SET UP THE MASTER (T3) MATRIX, THE USER SETS UP THE INDIVIDUAL MATRICES DESCRIBING THE TRANSLATION, ROTATIONS, AND PERSPECTIVE OF THE VIEW, THEN MULTIPLIES THEM TOGETHER. EACH OF THESE COMPONENT MATRICIES IS A 4X4 FORTRAN ARRAY. A COMPONENT MATRIX IS THEN SET UP BY CALLING A SUBROUTINE WITH THE NAME OF THE ARRAY.

```
CALL TRANSL (X,Y,Z,NAME)
```

SETS UP A MATRIX TO MOVE THE ORIGIN TO (X,Y,Z).

```
CALL PITCH (X,NAME)
```

SETS UP A MATRIX TO PITCH THE DATA X DEGREES.

```
CALL ROLL (X,NAME)
```

SETS UP A MATRIX TO ROLL THE DATA X DEGREES.

```
CALL YAW (X,NAME)
```

SETS UP A MATRIX TO YAW THE DATA X DEGREES.

```
CALL PERSP (X,NAME)
```

SETS UP A MATRIX TO SHOW THE DATA IN POINT PERSPECTIVE, THE PROJECTION POINT WILL BE X IN FRONT OF THE Z=0 PLANE, AND THE PROJECTION PLANE WILL BE Z=0. IF NO PERSPECTIVE IS SPECIFIED, PARALLEL PROJECTION IS ASSUMED.

THE CALL

```
CALL MULM3 (N,X1,X2,...,XN,X)
```

MULTIPLIES THE N 4X4 MATRICIES X1 ... XN TOGETHER AND STORES THE RESULT IN THE 4X4 MATRIX X. THUS THE ESTABLISH THE MASTER TRANSFORMATION MATRIX, THE USER WOULD MULTIPLY HIS COMPONENT MATRICIES TOGETHER, IN THE DESIRED ORDER, AND STORE THE RESULT IN THE ARRAY IN THE T3 COMMON AREA. NOTE THAT A TRANSLATION FOLLOWING A ROTATION IS DIFFERENT FROM THE SAME TRANSLATION PRECEDING THE ROTATION. THE PERSPECTIVE TRANSFORMATION WILL NORMALLY BE THE LAST TO BE PERFORMED.

THE GENERATING ROUTINES ARE

```
CALL PGEN3 (X,Y,Z,I)

CALL VGEN3 (DELTAX,DELTAY,DELTAZ,I)
```

WHICH ARE STRICTLY ANALOGOUS TO PGEN AND VGEN BUT IMPLICITLY USE THE T3 MATRIX TO FORM THE PROJECTED VIEW.

A USER MAY USE ANY OTHER SCHEME TO SET UP THE MASTER MATRIX, AND MAY THUS PRODUCE OTHER RESULTS (SHEARING, MAGNIFICATION, ETC.).

XV.2 THREE-DIMENSIONAL DISPLAY ROUTINES

****

ROUTINES ARE PROVIDED FOR PLOTTING CLIPPED, PARALLEL OR POINT PROJECTIONS OF POINTS AND LINES SPECIFIED BY THREE-DIMENSIONAL CARTESIAN COORDINATES, THE THREE-DIMENSIONAL TRANSLATION, ROTATION, AND PERSPECTIVE OF THE DATA IS GOVERNED BY A SINGLE 4X4 HOMOGENEOUS MATRIX WHICH SHOULD BE IN A COMMON AREA "T3" ESTABLISHED IN THE USER'S PROGRAM.

IF NO ROTATION OR TRANSLATION IS IN EFFECT, THE "X" AND "Y" AXES OF THE USER'S THREE-DIMENSIONAL DATA WILL BE THE X AND Y AXES OF THE USER'S DATA PAGE AS ESTABLISHED BY THE "SETWIND" ROUTINE. THE Z AXIS WILL BE PERPENDICULAR TO THE X AND Y AXES AND DIRECTED OUT OF THE SCOPE (RIGHT-HANDED SYSTEM). ROTATIONS ARE SPECIFIED BY PITCH (AROUND THE Z AXIS), ROLL (AROUND THE Y AXIS), AND YAW (AROUND THE X AXIS). THAT IS, TO GET A DIFFERENT VIEW OF HIS DATA THE USER CHANGES NOT THE ACTUAL COORDINATES OF HIS DATA BUT THE VALUE OF THE MATRIX; THEN AT GENERATION TIME A DIFFERENT VIEW WILL BE PRODUCED.

TO SET UP THE MASTER (T3) MATRIX, THE USER SETS UP THE INDIVIDUAL MATRICES DESCRIBING THE TRANSLATION, ROTATIONS, AND PERSPECTIVE OF THE VIEW, THEN MULTIPLIES THEM TOGETHER. EACH OF THESE COMPONENT MATRICIES IS A 4X4 FORTRAN ARRAY. A COMPONENT MATRIX IS THEN SET UP BY CALLING A SUBROUTINE WITH THE NAME OF THE ARRAY.

```
CALL TRANSL (X,Y,Z,NAME)
```

SETS UP A MATRIX TO MOVE THE ORIGIN TO (X,Y,Z).

```
CALL PITCH (X,NAME)
```

SETS UP A MATRIX TO PITCH THE DATA X DEGREES.

```
CALL ROLL (X,NAME)
```

SETS UP A MATRIX TO ROLL THE DATA X DEGREES.

```
CALL YAW (X,NAME)
```

SETS UP A MATRIX TO YAW THE DATA X DEGREES.

```
CALL PERSP (X,NAME)
```

SETS UP A MATRIX TO SHOW THE DATA IN POINT PERSPECTIVE, THE PROJECTION POINT WILL BE X IN FRONT OF THE Z=0 PLANE, AND THE PROJECTION PLANE WILL BE Z=0. IF NO PERSPECTIVE IS SPECIFIED, PARALLEL PROJECTION IS ASSUMED.

THE CALL

```
CALL MULM3 (N,X1,X2,...,XN,X)
```

MULTIPLIES THE N 4X4 MATRICIES X1 ,.. XN TOGETHER AND STORES THE RESULT IN THE 4X4 MATRIX X, THUS THE ESTABLISH THE MASTER TRANSFORMATION MATRIX, THE USER WOULD MULTIPLY HIS COMPONENT MATRICIES TOGETHER, IN THE DESIRED ORDER, AND STORE THE RESULT IN THE ARRAY IN THE T3 COMMON AREA, NOTE THAT A TRANSLATION FOLLOWING A ROTATION IS DIFFERENT FROM THE SAME TRANSLATION PRECEDING THE ROTATION, THE PERSPECTIVE TRANSFORMATION WILL NORMALLY BE THE LAST TO BE PERFORMED.

THE GENERATING ROUTINES ARE

```
CALL PGEN3 (X,Y,Z,I)

CALL VGEN3 (DELTAX,DELTAY,DELTAZ,I)
```

WHICH ARE STRICTLY ANALOGOUS TO PGEN AND VGEN BUT IMPLICITLY USE THE T3 MATRIX TO FORM THE PROJECTED VIEW,

A USER MAY USE ANY OTHER SCHEME TO SET UP THE MASTER MATRIX, AND MAY THUS PRODUCE OTHER RESULTS (SHEARING, MAGNIFICATION, ETC.).

XVIII. ASSEMBLY AND LOADING

XVIII.1 FORTRAN

THE FORTRAN VERSIONS OF THE ROUTINES HAVE BEEN MADE INTO
A LIBRARY WHICH IS PART OF THE FORTRAN SUBROUTINE LIBRARY. SINCE
THIS LIBRARY IS NORMALLY SEARCHED EACH TIME A LOAD OCCURS, THE USER
NEED NO LONGER SPECIFY THE DISPLAY PACKAGE AS A MODULE TO BE LOADED
EXPLICITLY; SIMPLY REGARD THE DISPLAY ROUTINES AS FORTRAN LIBRARY
ROUTINES LIKE "SIN" AND "COS". SINCE THE LIBRARY IS LOADED SELECTIVELY, DISPLAY PROGRAMS
WILL BE SHORTER UNDER THE NEW LOADING MECHANISM THAN THE OLDER.

THE STATEMENT "COMMON /T3/ NAME(4,4)" NEED APPEAR IN A USER'S
PROGRAM ONLY IF HE IS USING THE THREE-DIMENSIONAL DISPLAY ROUTINES.

XVIII.2 LISP

TO ASSEMBLE THE DISPLAY PACKAGE FOR USE WITH LISP, INSERT THE
STATEMENT

```
L=0
```

FROM THE TELETYPE BEFORE ASSEMBLING DISCOM.MAC, I.E.:

```
.R MACRO
*DEV:DISLSP.REL=TTY:,DEV:DISCOM.MAC
L=0
↑Z
END OF PASS 1
↑Z
NO ERRORS DETECTED
```

AS THE FIRST STEP IN CREATING A LISP DISPLAY IMAGE,
THE FILE "DISLSP" SHOULD BE INPUT TO LISP:

```
(INC (INPUT DEV: DISLSP))
```

THE REL FILE (DISLSP.REL) MAY NOW BE LOADED, USING THE LISP LOADER,
EITHER INTO BINARY PROGRAM SPACE OR INTO EXPANDED CORE.
THE LISP DISPLAY PACKAGE IS NOW READY TO GO. THE "DISPINIT" ROUTINE
MUST BE CALLED BEFORE ANY OTHER ROUTINE IN THE PACKAGE CAN BE REFERRED TO.

XVIII.3 ASSEMBLY LANGUAGE

THE ROUTINES MAY BE USED IN CONJUNCTION WITH ASSEMBLY CODE BY
USING THE FORTRAN VERSION OF THE PACKAGE (DISCOM.REL) AND IMITATING
THE FORTRAN COMPILER IN THE WAY IT CALLS SUBROUTINES.

XIX. VERSION-DEPENDENT FEATURES

CERTAIN SYMBOLS ARE DECLARED EXTERNAL IN THE SOURCE OF THE DISPLAY PACKAGE. THEY ARE RECORDED HERE AS A MATTER OF RECORD IN CASE FORSC OR LISP CHANGES IN SUCH A WAY THAT LOADING CAUSES THESE SYMBOLS TO BE RESOLVED DIFFERENTLY.

XIX.1 FORTRAN

THE STATEMENTS

```
EXTERN DEPOT.
```

AND

```
TYOLOC: XWD 0,DEPOT,-115
```

APPEAR IN THE SOURCE FILE DISCOM.MAC. TYOLOC SHOULD CONTAIN, AFTER LOADING, THE ADDRESS LABELLED "CO" IN FORSE ("CO" IS NOT INTERNAL IN FORSE), SHOULD A NEW RELEASE OF FORSE BE AQUIRED, THE "FUDGE FACTOR" OF OCTAL -115 MAY HAVE TO BE ADJUSTED.

XIX.2 LISP

THE STATEMENTS

```
EXTERN TYO
```

AND

```
TYOLOC: EXP TYO+50
```

APPEAR IN THE SOURCE FILE DISCOM.MAC. TYOLOC WHOULD CONTAIN, AFTER LOADING, THE ADDRESS LABELLED "TTYO" IN LISP. THE "FUDGE FACTOR" OF OCTAL 50 MAY HAVE TO ADJUSTED IN A NEW RELEASE OF LISP APPEARS.

XX. SAMPLE PROGRAMS

XX.1 FORTRAN EXAMPLES

XX.1.1 A PROGRAM TO SHOW A POINT TRACING THE BOUNDARY OF A CIRCLE.

```
        COMMON /TØ/ DOG
        DIMENSION H(1ØØ),P1(3),P2(3)
100     FORMAT(F)
        ACCEPT 1ØØ,EPSI                          ; GOVERNS THE SPEED
        CALL DSPINI(H,1ØØ)
        CALL DCLPIC(P1,3)
        CALL DCLPIC(P2,3)
        CALL SETWIND(-1,1,-1,1)
        CALL SETPORT(Ø,1Ø23,Ø,1Ø23)
        X=Ø.
        Y=1.
        L=1
        K=.FALSE.
1       GO TO (2,3),L
2       CALL GENINI(P1)
        GO TO 4
3       CALL GENINI(P2)
4       CALL PARMS(1,7)
        CALL PGEN(X,Y,1)
        GO TO (5,6),L
5       CALL GENEND(P1)
        IF (K) CALL DESTROY(P2)
        K=.TRUE.
        GO TO 7
6       CALL GENEND(P2)
        CALL DESTROY(P1)
7       L=3-L
        X=X+EPSI*Y
        Y=Y-EPSI*X
        GO TO 1
        END
```

```
XX.1.2 TO DISPLAY A TUMBLING CUBE WITH CORNERS LABELLED, IN PERSPECTIVE.

       COMMON /T3/ T(4,4)
       DIMENSION H(200),P1(3),P2(3)
       DIMENSION YM(4,4),PRM(4,4),TM(4,4),RM(4,4),PM(4,4)
500    FORMAT('+(',I1,',',I1,',',I1,')')
       CALL DSPINI(H,200)
       CALL DCLPIC(P1,3)
       CALL DCLPIC(P2,3)
       CALL SETWIND(-1,3,-1,3)
       CALL SETPORT(0,1023,0,1023)
       CALL PERSP(1,PRM)
       CALL TRANSL(0,0,1,TM)
       R=0.
       P=0.
       Y=0.
       L=1
       M=.FALSE.
11     CALL ROLL(R,RM)
       CALL YAW(Y,YM)
       CALL PITCH(P,PM)
       CALL MULM3(5,PM,RM,YM,TM,PRM,T)
       GO TO (1,2),L
1      CALL GENINI(P1)
       GO TO 3
2      CALL GENINI(P2)
3      CALL PARMS(1,7)
       DO 4 I=0,1
       DO 4 J=0,1
       DO 4 K=0,1
       CALL PGEN3(I,J,K,0)
       CALL TEXTP
       TYPE 500,I,J,K
       CALL PGEN3(I,J,K,0)
       IF (I .EQ. 1) GO TO 5
       CALL VGEN3(1,0,0,1)
       CALL PGEN3(I,J,K,0)
5      IF (J .EQ. 1) GO TO 6
       CALL VGEN3(0,1,0,1)
       CALL PGEN3(I,J,K,0)
6      IF (K .EQ. 1) GO TO 4
       CALL VGEN3(0,0,1,1)
4      CONTINUE
       GO TO (7,8),L
7      CALL GENEND(P1)
       IF (M) CALL DESTROY(P2)
       M=.TRUE.
       GO TO 9
8      CALL GENEND(P2)
       CALL DESTROY(P1)
9      L=3-L
       R=R+1.
       P=P+2.
       Y=Y+3.
       GO TO 11
       END
```

XX.2 LISP EXAMPLE

****

THIS IS THE DEFINITION OF AN EXPR WHICH PUTS LISP OUTPUT ON THE DISPLAY INSTEAD OF THE TELETYPE.

```
(DEFPROP Q ((LAMBDA NIL (PROG (A QQ QQQ)
        (GENERATE QQ)
A       (SETQ QQQ (EVAL (READ)))
        (DESTROY (QUOTE QQ))
        (GENERATE QQ
                (PARMS 2 7)
                (SETWIN 0 1 0 1)
                (SETPOR 0 1777 0 1777)
                (PGEN 0 .9 0)
                (TEXT QQQ))
        (GO A)))
EXPR)
```

APPENDIX
########

****** CATALOG OF FORTRAN DISPLAY ERRORS ******

WHEN A FATAL ERROR IS ENCOUNTERED BY THE DISPLAY PACKAGE, THE MESSAGE "DISPLAY ERROR NUMBER X" WILL BE TYPED ON THE USER'S TELETYPE AND THE TELETYPE WILL BE RETURNED TO MONITOR LEVEL. BELOW IS A LIST OF THE ERRORS WHICH MAY OCCUR,

ERRORS EXPLAINED AS "TYPE" ARE CAUSED AS FOLLOWS: ARGUMENTS TO THE ROUTINES IN THE DISPLAY PACKAGE ARE EITHER NAMES OF ARRAYS, FLOATING-POINT SINGLE PRECISION NUMBERS, OR FIXED POINT INTEGERS, THE ROUTINES WILL CONVERT THE LAST TWO DATA TYPES TO EACH OTHER WHEN NECESSARY, BUT WILL NOT CONVERT OTHER TYPES (DOUBLE-PRECISION, ETC.). A "TYPE" ERROR IS CAUSED WHEN THE CONVERSION CANNOT BE PERFORMED. ALSO LISTED BELOW FOR "TYPE" ERRORS IS THE NUMBER OF THE ARGUMENT IN THE ARGUMENT LIST, AND THE TYPE TO WHICH THE CONVERSION IS TO BE MADE; THE SPEED WILL BE SOMEWHAT GREATER IF THE ARGUMENT IS GIVEN AS THE CORRECT DATA TYPE EVEN THOUGH THE CONVERSION FROM FLOATING TO FIXED OR VICE VERSA CAN BE PERFORMED.

CERTAIN ROUTINES CAN BE CALLED ONLY DURING GENERATION; OTHERS ONLY OUTSIDE GENERATION. OTHERS (E,G, SETWIND) CAN BE CALLED AT ANY TIME. THE EXPLANATION "OUT OF PLACE" MEANS THAT A ROUTINE HAS BEEN CALLED INCORRECTLY WITH RESPECT TO THE "GENINI" AND "GENEND" ROUTINES.

ERROR NUMBER	ROUTINE	EXPLANATION
0 ## ##	DSPINI	CANNOT INIT DISPLAY, PROBABLY, THE DISPLAY HAS OR YOUR JOB IS TOO LARGE TO FIT IN CORE TOGETHER WITH SOME OTHER JOB ALREADY RUNNING, BEEN ASSIGN'ED OR INIT'ED BY ANOTHER JOB,
1	DSPINI	TYPE 2 FIX
2	GENINI	TRYING TO GENINI AN ALREADY EXISTING FRAME
3	******	NO MORE SPACE FOR DISPLAY FILE, THIS ERROR CAN OCCUR ANYWHERE IN THE GENERATING PROCESS, TO ESTIMATE THE SPACE NEEDED, ALLOW 1 WORD FOR EACH 2 SHORT VECTORS, 1 WORD FOR EACH 6 CHARACTERS, 1 WORD FOR EACH POINT, AND 1 WORD FOR EACH 10 RASTER POINTS; THEN ADD ABOUT 10% FOR OVERHEAD,
4	VGEN	TYPE 1 FLOAT
5	VGEN	TYPE 2 FLOAT
6	VGEN	TYPE 3 FIX
7	PGEN	TYPE 1 FLOAT
8	PGEN	TYPE 2 FLOAT
9	PGEN	TYPE 3 FIX
10	SETWIND	TYPE 1 FLOAT
11	SETWIND	TYPE 2 FLOAT
12	SETWIND	TYPE 3 FLOAT
13	SETWIND	TYPE 4 FLOAT
14	SETPORT	TYPE 1 FIX
15	SETPORT	TYPE 2 FIX
16	SETPORT	TYPE 3 FIX
17	SETPORT	TYPE 4 FIX
18	PARMS	TYPE 1 FIX
19	PARMS	TYPE 2 FIX
20	PARMS	INTENSITY IS NOT BETWEEN 0 AND 7 INCLUSIVE
21	PARMS	SCALE IS NOT 1,2,4, OR 8

```
22      SPICT   NON-EXISTANT FRAME
23      BLANK   NON-EXISTANT FRAME
24      UNBLANK NON-EXISTANT FRAME
25      DESTROY NON-EXISTANT FRAME
26      REPLACE NON-EXISTANT FRAME
27      RASTER  TYPE    2       FIX
28      RASTER  X INITIAL IS OUT OF RANGE
29      RASTER  TYPE    3       FIX
30      RASTER  Y INITIAL IS OUT OF RANGE
31      RASTER  TYPE    4       FIX
32      RASTER  X SIZE < 0
33      RASTER  TYPE    5       FIX
34      RASTER  Y SIZE < 0
35      RASTER  TYPE    6       FIX
36      RASTER  TYPE    7       FIX
37      RASTER  TYPE    8       FIX
38      RASTER  BYTE SIZE IS NOT 1,2, OR 3
39      RASTER  WOULD GO OFF SCREEN HORIZONTALLY
40      RASTER  WOULD GO OFF SCREEN VERTICALLY
41      GEN3    TYPE    1       FLOAT
42      GEN3    TYPE    2       FLOAT
43      GEN3    TYPE    3       FLOAT
44      GEN3    TYPE    4       FIX
45      MULM3   TYPE    1       FIX
46      TRANSL  TYPE    1       FLOAT
47      TRANSL  TYPE    2       FLOAT
48      TRANSL  TYPE    3       FLOAT
49      PITCH   TYPE    1       FLOAT
50      YAW     TYPE    1       FLOAT
```

```
51      ROLL    TYPE    1       FLOAT
52      PERSP   TYPE    1       FLOAT
53      SLEEP   TYPE    1       FIX
54      SETPORT X MIN < Ø OR X MIN > 1Ø23
55      SETPORT X MAX < Ø OR X MAX > 1Ø23
56      SETPORT Y MIN < Ø OR Y MIN > 1Ø23
57      SETPORT Y MAX < Ø OR Y MAX > 1Ø23
58      SETWIND X MIN > OR = X MAX
59      SETWIND Y MIN > OR = Y MAX
6Ø      SETPORT X MIN > OR = X MAX
61      SETPORT Y MIN > OR = Y MAX
62      SPICT   NON-EXISTANT PICTURE
63      ******  TRYING TO CONVERT AN INTEGER > 2↑26 TO
                FLOATING POINT
64      ******  FAILURE DURING SETUWP UUO, SHOULD NOT OCCUR.
65      VGEN    OUT OF PLACE.
66      GENINI  OUT OF PLACE.
67      PARMS   OUT OF PLACE.
68      PGEN    OUT OF PLACE.
7Ø      GENEND  OUT OF PLACE.
71      SPICT   OUT OF PLACE.
74      DESTROY OUT OF PLACE.
75      NOSYNC  OUT OF PLACE.
76      SYNC    OUT OF PLACE.
77      COMPIC  OUT OF PLACE.
78      REPLACE OUT OF PLACE.
79      TEXTP   OUT OF PLACE.
8Ø      LPON    OUT OF PLACE.
81      LPOFF   OUT OF PLACE.
82      R[illegible]ER  OUT OF PLACE.
```

****** CATALOG OF LISP DISPLAY ERRORS ******

WHEN A FATAL ERROR IS ENCOUNTERED BY THE DISPLAY PACKAGE, THE MESSAGE "DISPLAY ERROR NUMBER X" WILL BE TYPED ON THE USER'S TELETYPE AND THE TELETYPE WILL BE RETURNED TO TOP LEVEL. BELOW IS A LIST OF THE ERRORS WHICH MAY OCCUR.

```
83      DISTRT  SAME AS ERROR 0,
****************** E N D ******************
```

ERROR NUMBER	ROUTINE	EXPLANATION
0	DSPINI	CANNOT INIT DISPLAY, PROBABLY, THE DISPLAY HAS BEEN ASSIGN'ED OR INIT'ED BY ANOTHER JOB.
3	******	NO MORE SPACE FOR DISPLAY FILE, THIS ERROR CAN OCCUR ANYWHERE IN THE GENERATING PROCESS, TO ESTIMATE THE SPACE NEEDED, ALLOW 1 WORD FOR EACH 2 SHORT VECTORS, 1 WORD FOR EACH 6 CHARACTERS, 1 WORD FOR EACH POINT, AND 1 WORD FOR EACH 10 RASTER POINTS; THEN ADD ABOUT 10% FOR OVERHEAD.
24	PARMS	INTENSITY IS NOT BETWEEN 0 AND 7 INCLUSIVE
25	PARMS	SCALE IS NOT 1,2,4, OR 8
66	SETPORT	X MIN < 0 OR X MIN > 1023
67	SETPORT	X MAX < 0 OR X MAX > 1023
70	SETPORT	Y MIN < 0 OR Y MIN > 1023
71	SETPORT	Y MAX < 0 OR Y MAX > 1023
72	SETWIND	X MIN > OR = X MAX
73	SETWIND	Y MIN > OR = Y MAX
74	SETPORT	X MIN > OR = X MAX
75	SETPORT	Y MIN > OR = Y MAX
77	******	TRYING TO CONVERT AN INTEGER > 2**26 TO FLOATING POINT

********** E N D **********

BBN GRAFACON: HARDWARE DESCRIPTION

THE GRAFACON OR RAND TABLET IS A GRAPHICAL STYLUS INPUT DEVICE. IT CONSISTS OF A 10" BY 10" GRID OF 1024 BY 1024 WIRES. THERE ARE THEREFORE 2,20 DISTINGUISHABLE POSITIONS ON THE TABLET SURFACE. THESE POSITIONS ARE ASSIGNED COORDINATES IN THE SAME WAY AS COORDINATES ARE ASSIGNED TO THE RASTER POINTS OF THE DEC 340 DISPLAY, WITH (0,0) IN THE LOWER LEFT-HAND CORNER, (0,1777) IN THE UPPER LEFT-HAND CORNER, ETC. SINCE THE PHYSICAL SIZE OF THE TABLET SURFACE IS APPROXIMATELY THE SAME AS THE SIZE OF THE ACTIVE SURFACE OF THE DISPLAY SCREEN, THE GRAFACON IS WELL SUITED FOR USE IN CONJUNCTION WITH THIS DISPLAY.

THE STYLUS SERVES TWO LOGICALLY SEPARATE FUNCTIONS. FIRST, IT ACTS AS AN ANTENNA WHICH SENSES SIGNALS SENT THROUGH THE GRID WIRES AT REGULAR INTERVALS. THESE SIGNALS ARE DIFFERENT AT EVERY POSITION ON THE GRID, SO THAT THE INFORMATION RECEIVED BY THE STYLUS CAN BE CONVERTED BY THE GRAFACON CONTROL INTO THE COORDINATES OF THE STYLUS OVER THE TABLET. SECOND, THE STYLUS HAS A MOVEABLE TIP CONTAINING A MICROSWITCH. THIS SWITCH IS NORMALLY OPEN BUT IS CLOSED WHEN THE STYLUS IS PRESSED DOWN ON A SURFACE. CLOSING THE SWITCH CAUSES A FAINTLY AUDIBLE CLICK.

THE GRAFACON CONTROL SAMPLES COORDINATE DATA AT REGULAR INTERVALS OF APPROXIMATELY 220 MICROSECONDS. THIS PERIOD OF TIME WILL BE REFERRED TO AS ONE GRAFACON TICK. THE DATA MAY BE READ IN FROM THE GRAFACON COORDINATE BUFFER BY THE PDP-10 BY A DATAI 404 INSTRUCTION. FOR A PERIOD OF 40 MICROSECONDS OUT OF THE 220 MICROSECONDS, THE COORDINATE BUFFER IS CHANGING AND THE DATAI WILL PRODUCE INACCURATE DATA. WHEN THE DATA IS READ IN IT IS MARKED WITH A "DATA READY" BIT TO INDICATE WHETHER OR NOT IT WAS READ DURING THE VALID 180 MICROSECONDS OF THE TICK.

IF EITHER THE X OR Y COORDINATE CHANGES BY MORE THAN 1 BETWEEN TWO CONSECUTIVE SAMPLES AND THE PEN SWITCH IS CLOSED, THE ERROR BIT IS SET IN THE INPUT DATA. THIS INDICATES ONE OF THE FOLLOWING:

1) THE PEN SWITCH IS CLOSED, BUT THE STYLUS IS NOT BEING HELD DOWN ON THE TABLET.

2) THE PEN IS BEING MOVED TOO QUICKLY ACROSS THE TABLET. BELIEVE IT OR NOT, IT IS POSSIBLE TO MOVE THE PEN AT A RATE FASTER THAN 45 INCHES PER SECOND, THE RATE AT WHICH MORE THAN ONE WIRE CAN BE CROSSED IN 220 MICROSECONDS.

3) A GRAFACON MALFUNCTION; EITHER THE CONTROL LOGIC IS MALFUNCTIONING, OR, IF THE ERRORS OCCUR EVEN WHEN THE STYLUS IS NOT BEING PRESSED DOWN, THE PEN SWITCH IS BROKEN AND SHORTED CLOSED.

GRAFACON (RAND TABLET) SERVICE ROUTINE

I. INTRODUCTION.

A SERVICE ROUTINE (GFNSER) FOR THE BBN GRAFACON IS NOW PART OF THE MONITOR. THIS ROUTINE PERMITS THE RAND TABLET TO BE USED AS A STANDARD INPUT DEVICE UNDER THE 10/50 TIMESHARING MONITOR, AND ALSO PROVIDES FACILITIES FOR REAL-TIME MODIFICATION OF A DISPLAY FILE FOR THE 340 DISPLAY WHEN THE GRAFACON IS RUNNING. AS A STANDARD INPUT DEVICE, THE GRAFACON IS STARTED BY THE FIRST IN OR INPUT UUO, AND CONTINUES RUNNING UNTIL ALL ITS BUFFERS ARE FULL OR AN END-OF-FILE IS REACHED.

II. SINGLE-STROKE AND MULTIPLE-STROKE DATA COLLECTION.

THE SERVICE ROUTINE PROVIDES FOR TWO TYPES OF DATA COLLECTION, "SINGLE-STROKE" AND "MULTIPLE-STROKE." IN EACH CASE, DATA ARE COLLECTED ONLY WHEN THE PEN IS DOWN, I.E. WHEN THE PEN TIP SWITCH IS CLOSED.

A. SINGLE-STROKE DATA COLLECTION

IN "SINGLE-STROKE" MODE, THE ROUTINE WAITS AFTER THE DEVICE HAS BEEN STARTED UNTIL THE PEN IS PRESSED DOWN, THEN COLLECTS DATA UNTIL THE PEN IS LIFTED UP. THE LIFTING OF THE PEN IS THE END-OF-FILE CONDITION.

B. MULTIPLE-STROKE DATA COLLECTION

IN "MULTIPLE-STROKE" MODE, THE ROUTINE WAITS AFTER THE DEVICE HAS BEEN STARTED UNTIL THE PEN IS DOWN, AND THEN STARTS TO COLLECT DATA. THEREAFTER DATA ARE COLLECTED WHENEVER THE PEN IS DOWN, UNTIL THE PEN HAS BEEN UP (SWITCH OPEN) FOR A SPECIFIED LENGTH OF TIME. THIS "UPTIME COUNT" IS REINITIALIZED EACH TIME THE PEN GOES DOWN. IN THIS MODE, THE EXPIRATION OF THE UPTIME CONSTITUTES THE END-OF-FILE. A WORD OF ALL ONES (=1) IS INSERTED AS A DATA ITEM IN THE INPUT BUFFER EACH TIME THE PEN IS LIFTED AS A SEPARATOR BETWEEN STROKES.

IN EITHER CASE THE DEVICE MAY HAVE TO BE TEMPORARILY SHUT OFF IF ALL THE BUFFERS ARE FILLED.

THE GRAFACON MAY PRODUCE TWO DIFFERENT KINDS OF INTERRUPTS, THEY ARE ON THE SAME INTERRUPT CHANNEL BUT MAY BE SEPARATELY ENABLED.

FIRST, AN INTERRUPT MAY OCCUR WHEN NEW DATA HAS BEEN READ INTO THE GRAFACON COORDINATE BUFFER. SINCE 220 MICROSCECONDS IS MUCH FASTER THAN ONE NORMALLY WANTS TO SAMPLE DATA, THE INTERFACE IS EQUIPPED WITHA A COUNTER SO THAT ONLY EVERY N'TH DATA SAMPLE BY THE GRAFACON CONTROL WILL PRODUCE AN INTERRUPT TO THE PDP-10, HERE N IS AN INTEGER WHICH IS SPECIFIED BY A CONO 404 INSTRUCTION AND MAY BE BETWEEN 1 AND 400 OCTAL INCLUSIVE. THE RESULT IS THAT THE SAMPLING RATE MAY BE VARIED FORM A SAMPLE EVERY 220 MICROSECONDS TO A SAMPLE EVERY 56 MILLISECONDS.

SECOND, AN INTERRUPT MAY OCCUR WHENEVER THE PEN SWITCH CHANGES STATUS (I. E. EITHER OPENS OR CLOSES).

III. DATA MODES.

TWO STANDARD DATA MODES MAY BE USED WITH THE GRAFACON: IMAGE (MODE 10 OCTAL) AND IMAGE BINARY (MODE 13 OCTAL). IN EACH CASE, THE BYTE SIZE IS 36 BITS.

A. IMAGE MODE

IN IMAGE MODE, DATA ARE COLLECTED EXACTLY AS PROVIDED BY THE DEVICE (I.E. BY A DATAI 404).

B. IMAGE BINARY MODE

IN IMAGE BINARY MODE, DATA ITEMS SUPPLIED BY THE GRAFACON IN WHICH THE "DATA READY" BIT (BIT 0) IS OFF OR THE "ERROR" BIT (BIT 1) IS ON ARE IGNORED AND DO NOT APPEAR IN THE INPUT BUFFER. THE "GOOD" DATA WHICH DO APPEAR IN THE INPUT BUFFER ARE STRIPPED TO CONTAIN ONLY COORDINATE INFORMATION: 10 BITS OF Y RIGHT-JUSTIFIED IN THE LEFT HALF OF THE WORD AND 10 BITS OF X RIGHT-JUSTIFIED IN THE RIGHT HALF OF THE WORD. ALL OTHER BITS ARE 0.

IV. SPACING OF DATA.

A. EQUAL-TIME

THE USER MAY SPECIFY THE DATA SAMPLING RATE AS THE CONTENTS OF THE GRAFACON PRESET COUNTER. THAT IS, THE INTEGER N PROVIDED BY THE USER IS 1 LESS THAN THE NUMBER OF GRAFACON TICKS BETWEEN DATA SAMPLES (1 GRAFACON TICK = 220 MICROSECONDS).

B. EQUAL-SPACE

THE USER MAY ALSO SPECIFY THAT THE DATA BE THINNED BY THE SERVICE ROUTINE, I.E. A POINT WILL APPEAR IN THE INPUT BUFFER ONLY IF IT IS AT A DISTANCE OF AT LEAST N IN EITHER X OR Y FROM THE PREVIOUS POINT IN THE BUFFER (EXCEPT FOR THE FIRST POINT OF A STROKE, WHICH ALWAYS WILL APPEAR IN THE BUFFER). IN THIS CASE THE "N" SUPPLIED BY THE USER WILL BE USED BOTH AS THE DATA SAMPLING RATE AND AS THE SPACING PARAMETER.

TO PREVENT UNNECESSARILY FREQUENT INTERRUPTS BY THE GRAFACON, THE SERVICE ROUTINE RESTRICTS N TO BE GREATER THAN OR EQUAL TO 5.

V. TRACKING.

"TRACKING" IS THE FIRST WAY IN WHICH THE GRAFACON SERVICE ROUTINE MAY MODIFY A USER'S DISPLAY FILE WHILE THE GRAFACON IS RUNNING. THE USER MAY SUPPLY AN ADDRESS IN HIS JOB AREA TO BE USED AS THE "TRACKING ADDRESS." AT INTERRUPT LEVEL, THE SERVICE ROUTINE WOULD PLACE THE GRAFACON Y COORDINATE IN BITS 8-17 OF THIS LOCATION, AND THE X COORDINATE IN BITS 26-35. THIS OPERATION WOULD OCCUR WHETHER THE PEN IS UP OR DOWN. TYPICALLY, THIS ADDRESS WOULD CONTAIN A POINT MODE WORD AND WOULD BE PART OF A DISPLAY FILE. THE RESULT IS THE GRAFACON ANALOGUE OF LIGHT-PEN TRACKING.

VI. INKING.

THE SERVICE ROUTINE WILL ALSO PERFORM "INKING" WHILE THE GRAFACON IS RUNNING IF REQUESTED AND A LEGAL INKING BUFFER IS SPECIFIED BY THE USER. "INK" IS A DISPLAY SUBROUTINE WHICH SHOWS THE TRAJECTORY OF THE PEN ON THE TABLET WHEN THE PEN IS DOWN. THE USER SUPPLIED THE BEGINNING AND ENDING ADDRESSES OF THE INK BUFFER; THE SERVICE ROUTINE WILL INITIALIZE, MODIFY, AND CLEAR THE INK BUFFER ON COMMAND. IT IS THE USER'S RESPONSIBILITY TO INSERT A CALL TO THE INK BUFFER IN HIS DISPLAY FILE AS A "DJS INK VTR." THE RETURN WOULD BE IN PARAMETER MODE.

IF THE INK BUFFER IS FILLED, INKING WILL STOP, BUT THE DEVICE WILL NOT BE SHUT OFF AND DATA COLLECTION WILL CONTINUE.

B, USETO

```
        USETO D,E

E:      XWD INKBEG,INKEND
        XWD TRKADR,RATE
        EXP UPTIME
```

THE USETO UUO IS USED TO PASS PARAMETERS TO THE SERVICE ROUTINE. E SHOULD CONTAIN THE ADDRESSES OF THE FIRST AND LAST LOCATIONS OF THE INK BUFFER, IF DESIRED; 0 OTHERWISE. E+1 (LEFT HALF) SHOULD CONTAIN THE TRACKING ADDRESS, IF ANY; 0 OTHERWISE. E+1 (RIGHT HALF) SHOULD CONTAIN THE DATA RATE (OR DATA SPACING) AS AN INTEGER BETWEEN 5 AND 377 OCTAL INCLUSIVE. E+2 SHOULD CONTAIN THE UPTIME PARAMETER IN GRAFACON TICKS. THE MAXIMUM PERMISSABLE UPTIME IS 40000 DECIMAL TICKS (APPROXIMATELY 10 SECONDS).

SPECIAL CASES:

1) IF INKING IS SPECIFIED AND THE CONTENTS OF E IS 0, NO ERROR CONDITION RESULTS BUT INKING WILL NOT OCCUR.

2) IF THE CONTENTS OF E IS NONZERO BUT
   A) INKBEG OR INKEND IS LESS THAN 20 OCTAL OR OUTSIDE THE USER'S JOB AREA
   OR B) INKEND < INKBEG+4
   THEN AN ERROR MESSAGE WILL BE TYPED ON THE USER'S TELETYPE AND HIS JOB WILL BE PUT IN AN ERROR STOP STATE.

3) IF TRACKING IS SPECIFIED AND THE CONTENTS OF E+1 (LEFT HALF) IS 0, NO ERROR RESULTS BUT TRACKING WILL NOT OCCUR.

4) IF THE CONTENTS OF E+1 (LEFT HALF) IS NONZERO BUT IS LESS THAN 20 OCTAL OR OUTSIDE THE USER'S JOB AREA, THEN AN ERROR MESSAGE WILL BE TYPED ON THE USER'S TELETYPE AND HIS JOB PUT IN AN ERROR STOP STATE.

5) IF THE CONTENTS OF E+1 (RIGHT HALF) IS 0, THE MAXIMUM DATA SPACING (377 OCTAL) WILL BE USED.

6) IF THE CONTENTS OF E+1 (RIGHT HALF) IS 1, 2, 3, OR 4, THEN 5 WILL BE USED AS THE DATA SPACING.

7) IF THE CONTENTS OF E+1 (RIGHT HALF) IS GREATER THAN 377 OCTAL, THAN ONLY THE RIGHTMOST 8 BITS WILL BE USED BY THE SERVICE ROUTINE AND COMMENTS 5 AND 6 WILL APPLY.

8) IF THE CONTENTS OF E+2 IS NEGATIVE OR GREATER THAN OR EQUAL TO 40000 DECIMAL, 0 WILL BE USED AS THE UPTIME PARAMETER.

9) IF NO USETO AT ALL IS PERFORMED, THE EFFECT IS THE SAME AS IF THE USETO WERE PERFORMED BUT ALL THREE WORDS E,E+1, AND E+2 WERE 0.

10) IF AN INK BUFFER ALREADY EXISTS WHEN THE USETO IS PERFORMED, IT

C. IN D, AND INPUT D,

THESE UUO'S ARE COMPLETELY STANDARD,

D. CLOSE D,

THE CLOSE UUO IS COMPLETELY STANDARD, IT CLEARS THE END-OF-FILE FLAG SO THAT ANOTHER INPUT CAN BE PERFORMED,

IF THERE IS AN INK BUFFER WHEN THE CLOSE IS PERFORMED, IT IS CLEARED (THE INK IS ERASED),

E. RELEAS D,

THIS UUO IS COMPLETELY STANDARD,

F. EXAMPLE

A "NORMAL" SEQUENCE OF UUO'S FOR THE GRAFACON MIGHT BE AS FOLLOWS: INIT, USETO, INBUF, INPUT, INPUT, ..., INPUT, CLOSE, INPUT, ...

H, R, LEWIS
4/23/69