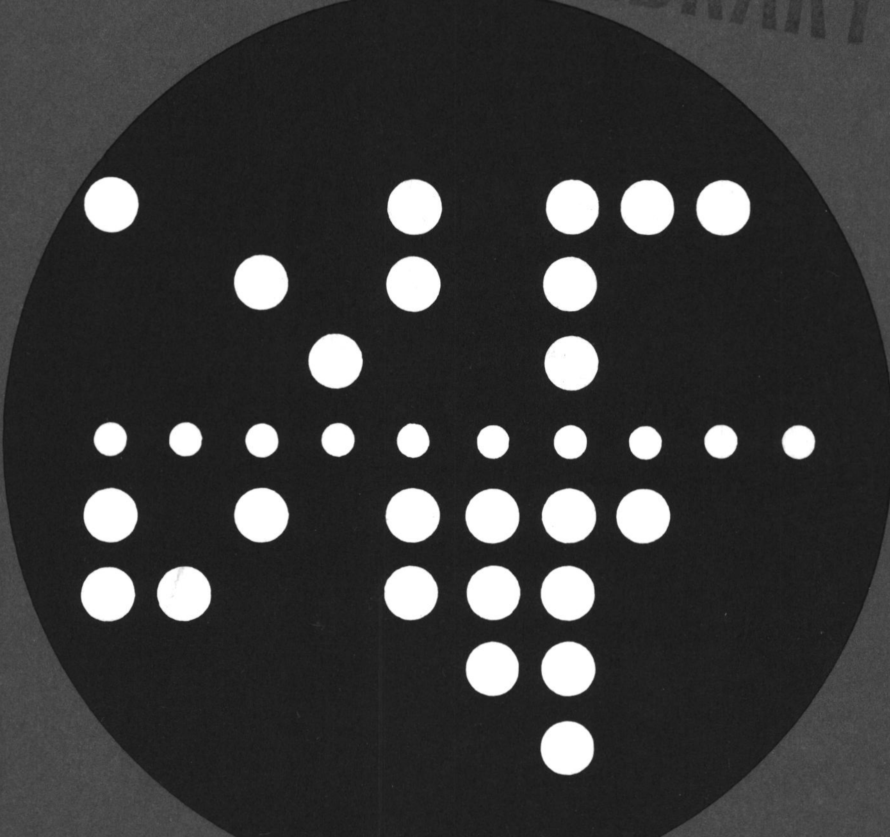
COMPUTING CENTRE NEWSLETTER

JRC COMPUTER GRAPHICS



Commission of the European Communities
JOINT
RESEARCH
CENTRE
Ispra Establishment

SPECIAL ISSUE

JRC Computer Graphics

THE GRAPHIT SYSTEM

Herman I. de Wolde

December 1980

CONTENTS

1. GRAPHIT SYSTEM, INTRODUCTION

Computer graphics is a powerful tool for the representation of complex output. The reason for the late development of the field, in comparison with other aspects of computing, has mainly been due to the high costs of the hardware and the difficulties of designing systems which could serve a large number of essentially different applications.

In recent years the developments have shown increasing momentum; hardware costs have come down rapidly and powerful software packages are now available.

However, a complete system which could serve the entire graphic needs of the Joint Research Centre is not yet marketed. The problem is complicated by the wide range of different activities at the JRC, each requiring a separate approach to Computer Graphics.
To mention just a few of the present activities:

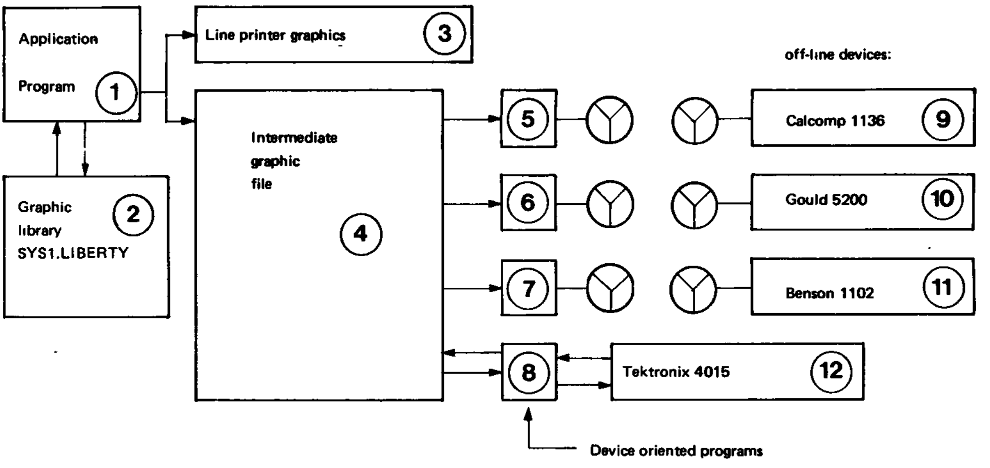
- Stress analyses of structures
- Shock wave propagation in fluids
- Electronic design
- Management information
- Statistical evaluation
- 2D and 3D functions
- Mapping

The Computing Centre has deliberately chosen for a flexible approach allowing for future extensions of the system.
The flexibility is in respect to both the hardware and the software:
- The software system implemented at the JRC-Ispra (known as GRAPHIT) is device independent. Any type of graphic device may be connected in an easy way.
- The software functions may be extended, according to the needs, with materials from various sources.

The concept of the GRAPHIT system is based on a flexible and powerful Fortran library of subroutines and the definition of an Intermediate Graphic File, which is a numerical description of the programmed image.
The elaboration of the Intermediate Graphic File for a particular device is defined as a separate and independent operation.

Diagram 1 gives the currently (Jan. 1981) installed configuration.

This chapter gives a short summary of each of the numbered items. The report continues with a more detailed description of the topics.



1.1 The Application Program

The application program may be any Fortran program. However, to use the GRAPHIT system for graphic output one has to:

- Concatenate the SYS1.LIBERTY library to the other necessary libraries.
- Call the routine GSTART before any graphic output.
- Call the routine GEND to close the Intermediate Graphic File at the end of the job.
- Have available a reserved output file with the specifications: VS, LRECL=800, BLKSIZE=804. The Fortran unit number for this file is 16.

  For the lineprinter graphics it is not necessary to have an Intermediate Graphic File.

1.2 The Graphic Library SYS1.LIBERTY

The Graphic Library SYS1.LIBERTY contains all the necessary Fortran routines for graphic output. The main sub-groups of routines are listed as follows:

- Back-end routines to write to the Intermediate Graphics File the numerical descriptions of the graphic commands.
- The routines for lineprinter graphics.
- The X-collection for easy graphics. Presently these routines draw 2D graphs and histograms in A4 and A5 format. The images are automatically numbered and all origin replacements are supplied by the system.
- GINOGRAF, for 2D plotting of histograms, functions, barcharts, piecharts, etc.
- GINO-F, for 2D and 3D plotting with transformation of structures.
- GINOZONE, for mapping problems.
- A series of high-level routines from various sources for 3D drawings with hidden line removal, grey simulation, etc.

1.3 Lineprinter Graphics

Plotting functions on the lineprinter is, of course, a rather crude way of obtaining graphics but for a first scanning of the results, or as data input check, it may be useful. The routines are embedded in the SYS1.LIBERTY library and do not require an Intermediate Graphic File.

1.4 The Intermediate Graphic File

The Intermediate Graphic File is in fact a numerical description of the desired graphic output. The structure of the file is sequential. The information is in compressed format to save on space requirements. For example, coordinates are mostly specified in just one byte.

The basic element is the command unit of the general form:

N,X,Y,K

in which:
N is the command number
X,Y are coordinates (if relevant)
K is an attribute (if relevant)

For example:

1, 2.0, 5.0, 2 means move the pen to the coordinates (2.0,5.0) with pen down.
6,2 Pen colour red required.

All higher order commands like "draw arc" or alphanumeric characters are broken down into basic plotting commands, which gives the system the required flexibility and easy adapting to additional devices.
The Intermediate Graphic File may be segmented into "pictures", by means of the subroutine PICTNO(K).
A more detailed description of the Intermediate Graphic File is given in appendix A.

1.5)
1.6) Device Oriented Programs
1.7)

The Device Oriented Programs (5,6,7 - fig. 1) prepare the magnetic tapes for the off-line plotters. The three programs have exactly the same structure. The Intermediate Graphic File is read and the individual commands are converted to proper subroutine calls for the chosen device.
Some options are not available on all three plotters. For example, the pen colour is not available on the Gould plotter. A command for another colour is simply ignored in this case.

1.8 Device Oriented Program Tektronix 4015

This program is different from the previous ones as it allows for on-line scanning of the Intermediate Graphic File. A number of options exists for close and detailed inspection of the image:
- Windowing by numerical specification
- Windowing by cursor
- Split-screen representation
- Enlarging
- Editing of drawings

1.9 The Calcomp 1136 penplotter

The Calcomp 1136 penplotter has recently been withdrawn from service by the Computing Centre. However, magnetic tapes for this device may still be prepared as this type of plotter is widely used in other research centers.
The plotter uses roll-paper with about 80 cm of height (y-direction). Normally, three pen colour are available. The device is very useful for large, very accurate drawings.

1.10 The Gould 5200 plotter

The Gould 5200 plotter is an electrostatic drawing device with 200 points per inch, accuracy 0.013 cm. The accuracy is less than for pen plotters but the high speed allows for very large graphic output. The paper speed is 4 cm/second, independent of the drawing intensity. The plotter uses roll-paper with a height of 11 inch (27 cm). Thus, y values may not be larger then this value. Without the use of the subroutine PICTNO, see 5.3, the the plotsize in the X-direction may not exceed 300 cm. However, for very large output, the mentioned subroutine in combination with the GOULDX procedure allows for a production which is limited only by the size of the magnetic tape, see 3.2.
The output quality is very good for reproduction.

1.11 The Benson 1102 plotter

The Benson 1102 plotter is a very accurate three colour penplotter with the following specifications:

- Axial penmovement 5 cm/sec
- Increment size 0.005 cm
- Plotsize in y-direction = paper width 30 cm

The device is suitable for small but highly accurate output.

1.12 Tektronix 4014/4015 terminal

The Tektronix terminals 4014/4015, of which several are installed at the JRC, are particularly apt for graphics output at the test phase of new software products. The graphic results may be scanned immediately after the job is finished. A command language for displaying the drawing partly or completely (including enlargements) has been developed by the Computing Centre.
The 19 inch screen of the Tektronix offers 1024(x) by 780(y) displayable points or 4096(x) by 3120(y) points in the enhanced mode.
A hardcopy may be made, offering a quality which is sufficient for reproduction.

In the following pages all the above mentioned items will be explained more in detail. Descriptions of particular subroutines will be given for the basic routines and in those cases where the software is not from the GINO packages.
All the GINO subroutines are described in the relevant manuals which are available at the Computing Support Library.
The Job Control Language definitions as given here are strictly related to the present mainframe configuration as installed at the JRC, Jan.1981.

It is not necessary for the graphic "pupil" to study all of this document. The information presented is divided into increasing levels of complexity and objects.

This manual contains a description of the present situation. However, as more options become available and the modifications of hardware will certainly require adaption of the system, a method of regular updates has been set up.
Each registered user of the graphic facilities will receive the descriptions of the modifications automatically and other information concerning the graphic applications at the JRC.

Please complete the form at the back of this manual for registration.

2. JCL PROCEDURES

2.1 Reservation of an Intermediate Graphic File

The user has to reserve his own Intermediate Graphic File on an on-line disk volume.
The easiest way to do this, is under TSO. The file will be catalogued automatically:

```
CREARES name USER0X RF1(V) RF2(S) LRECL(800) BLKSIZE(804)
```

in which:
name is the second part of the fully qualified name of the Intermediate Graphic File; TSOxxx.name
TSOxxx is the user identifier
USER0X is the volume name, X may be A, B,C, etc.

A standard quantity of space of 10 blocks with 15 times an increment of 10 blocks is available for the file which is equivalent to 10 tracks. For most of the applications such a space is sufficient. If you have a very large graphic output, the space may be defined larger by the QUANT and INCR options.
To look for the available free spaces on the user disk volumes, one may use the TSO command FREESPA. [See TSO HELP for more information].

2.2 Foreground execution

To execute a computer program with graphic output in foreground, one has to concatenate the SYS1.LIBERTY library with the other requested libraries and to define the Intermediate Graphic File as an output file. Normally, this is accomplished by using the LIB parameter to the LOADGO or LINK TSO command procedure.
The following TSO command procedure, called (for example) run.CLIST. may be used for the execution:

```
PROC 0
FREEALL
ALLOC DA('TSOxxx.name') FILE(FT16F001)
ALLOC DA(*) FILE(SYSPRINT)
LOADGO prog.OBJ FORTLIB LIB('SYS1.LIBERTY')
FREEALL
END
```

in which:
run.CLIST is the name of the procedure
TSOxxx.name is the fully qualified name of the Intermediate Graphic File
prog.OBJ is the object deck of the program
The terms written in capitals are obligatory.

Once this procedure has been prepared, you may execute the program by the command:

```
EXEC run.CLIST
```

In this example it is assumed that the input data comes from the terminal and the output is written to the terminal.

2.3 Batch execution

For the execution of an application program in batch, the deck composition is as follows:

```
//      JOB (your JOB card)
//      EXEC FTG1CLG,PRN=ERTY,VLB=COPICB,ULB=DISK
//CMP.SYSIN DD *
     fortran deck
/*
//GO.FT16F001 DD DSN=TSOxxx.name,UNIT=DISK,DISP=(OLD,KEEP)
//GO.SYSIN    DD *
     input data
/*
```

in which:

TSOxxx.name is the fully qualified name of the Intermediate Graphic File

It is not necessary to specify the volume if the file has been reserved under TSO (using CREARES) because it is then catalogued. However, if the Intermediate Graphic File has been reserved in another way without being catalogued, the parameter "VOL=SER=:..." must be added.

3. ELABORATION OF THE INTERMEDIATE GRAPHIC FILE

Once an Intermediate Graphic File has been composed, there are several ways available to visualize the results. The quickest way is by means of a Tektronix terminal under TSO. In general this facility is used during the test phase. If the graphic output is according to the requirements, the same file may be elaborated for the Gould or the Benson plotter to obtain the definitive results.

3.1 Graphit - Tektronix

The Graphit-Tektronix system uses a set of commands to display the entire Intermediate Graphic File or part of it. The screen is divided into a scratchpad, on which the literal communications between the user and the system are displayed and a square projection surface for the actual plotting.
The graphic part of the screen may be divided in up to 4 sections each of which may be adressed separately. A windowstack may be used to memorize up to 26 window definitions.
A time oriented Interrupt system may be invoked for early stops during the display of complicated drawings.
The cursor may retrieve coördinates from any of the screenparts for additional window definitions.
An editing option is available to prepare a report version of the image by means of the hardcopy unit; alphanumeric text may be added to the drawing.
The system runs under TSO. After LOGON with the procedure FG1LOG, one has to type:
GRAPHIT, for the allocation of the necessary files.
After the READY message type:

PLOTE filename volume

in which:
filename is the fully qualified name of the Intermediate Graphic File
volume is the disk volume.

3.1.1 Terminology

The following terminology is used:

Window - A rectangular part of the complete image as represented numerically by the Intermediate File. It is specified by coordinates in centimetres: XMIN, XMAX, YMIN, YMAX.

Windowname - One alphabetical character which names a window.

Windowstack - A series of specified windows; coordinates in centimetres with their single character windowname.

Displaystack - A matrix specifying numerically the displayed windows by respectively, screentype, screenpart, coordinates of the windows, coordinates of the viewports.

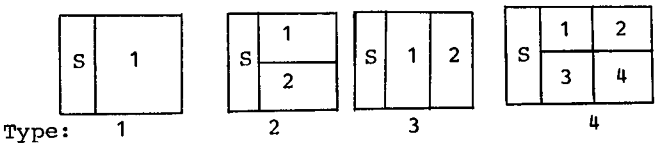
Viewport - A rectangular part of the screen where a picture, or a window in a picture, may be displayed. Coordinates are given in screen points.

Scratchpad - Part of the screen where literal communications between the user and the system are displayed.

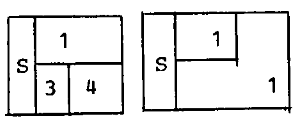
Screentype - A subdivision of the screen. At the default option, screentype 1, the screen is divided in a scratchpad, left part, and a square image part.
The image part may in turn be divided in up to four parts:
Screentype 2: Division in two subscreens, horizontally
Screentype 3: Division in two subscreens, vertically
Screentype 4: Division in four square parts

Screenpart

Numerical identification of a part of the screen:



in which:
S = Scratchpad
The screen may belong to different screentypes at the same time i.e. combinations such as the following are allowed:



.... etc

Plotting mode

After defining the viewport, by screentype and screenpart, and the window, the user has then to specify a plotting mode:

n - natural, the X,Y ratios remain unchanged, the image is a reduction or enlarging of the original while maintainingthe proportions.
o - optimal, the viewport is used to its maximal extend, the X,Y proportions may be changed.
r - real, starting from the left under point of the window, all distances are maintained. The image on the screen is an exact projection of the original. Excessive parts of the window are cut off.
c - copy, same as "real" for the hardcopy output; the image on the hardcopy is an exact projection of the original; all distances are maintained from the bottom left corner of the window. Excessive parts of the window are clipped.

3.1.2 Graphit Commands

Conventions:

- only the first four characters of each command are obligatory
- commands and arguments are separated by at least one blank
- a complete instruction may not exceed 28 characters, including blanks
- requested integers must be written as integers
- real constants may be written as real or as integers

Commands presently available are:

SUMMARY
The system writes a table with the specifications of the Intermediate File:
Total number of plot operations.
Extreme values of the entire file, picture numbers and picture extremes, if relevant.

SCREEN m
m is screentype, default m=1 (see table terminology).

BELL xx
xx = ON (default)
xx = OFF
To set the acoustic signal for attention.

INTERRUPT i
During the display of an image the processing stops any i seconds and waits for a second level command.
The second level commands are:

INTERRUPT j	= change i to j
STOP	return to main program
(void)	continue

default value i=0 bypass, interrupt system.

WINDOW a xmin xmax ymin ymax
Introduces a new window in the windowstack with an identifying name a, one alphabetical character. If a previously defined window has the same name, it will be overwritten, otherwise the new definition will be added to the stack.

STACK
The system writes a table with the window definitions.

DISPLAY k a x
The system displays window a on screenpart k, according to plotting mode x in which x may be n(natural), o(optimal), r(real or c(copy) (see plotting mode definition).
The allowed values for k depend on the current screentype definition.

DISPLAY k p x
The system displays picture p on screenpart k.
The picture numbering is based on the use of the Subroutine PICTNO(K) in the application program.

CURSOR a m k
A cursor defined window is added to the stack, the windowname is a, one alphabetical character, and the coordinates are estimated according to the window as is displayed on screenpart k for screentype m.
Second level commands:
X - (CR) for bottom - left corner followed by:
X - (CR) for top - right corner.
The cursor may also be set outside the specified window.

CLEAR
Erases screen, cancels display stack and sets the line counter of the scratchpad to zero.

EDIT
Erases screen and repeats the last display command for each screenpart. No messages are written, to allow for the editing and subsequent production of a report version on the harcopy unit.
At the end of the displaying, the system stops and the user may introduce comments on the screen by means of the keyboard. The alpha-cursor may be placed in any position by the keys:

Tab	Move right
Backspace	Move left
LF	Move down
CTRL-K	Move up

Available character sizes for the Tektronix 4015 terminal are:

KEYS	TOTAL LINES	NUMBER OF CHARACTERS PER LINE	ON SCRATCHPAD
ESC 8	35	74	16
ESC 9	38	81	18
ESC :	58	121	27
ESC ;	64	132	30

During the execution of the EDIT Command the interrupt, if any, is bypassed.
Both ASCII and APL characters may be used.
However, character deletion is not possible.

ENHANCED
The system uses 4096 x 3120 screenpoints in place of the standard 1024 x 780 points to produce more accurate drawings. This command is only available for terminals with the enhanced graphics option.

NORMAL
The default value NORMAL resets the 1024 x 780 screenpoints.

FRAME
The default option FRAME draws the rectangular viewport around the drawing.

NOFRAME
Omits the drawing of the viewport.

END
Program exit.

3.1.3 Error messages

The system contains a series of tests on the validity of the commands.
Continuation of the job is normally possible by specifying the correct instruction. The following list specifies the numbered messages.

SYNTAX ERROR N

N<28 The N-th character in the command expression is erroneous.
N=28 Command expression is too long; only a maximum of 27 characters and blanks are allowed.
N=35 First 4 characters are not recognized as a command keyword.
N=50 The required screenpart is not consistent with the screentype. For example screentype 1 does not allow for screenpart 2.
N=60 The requested picturenumber or window name does not exist.
N=81 Command error at interrupt level; at this moment only the following commands are legitimate:
INTERRUPT K (K=integer)
STOP
.... (VOID)

ERROR IN DISPLA N

N=46 Display stack error
N=83 Requested window not found in window stack
N=84 Requested screenpart not available
N=85 Erroneous screentype
N=86 Erroneous plotting mode
N=87 Error at reading the Intermediate Graphic File

CURSOR ERROR N

N=53 Requested screenpart does not exist
N=54 Requested screenpart is empty
N=55 Negative surface for requested window

ERROR IN DISPLAY FILE

The application program does not contain a call to GEND to close the Intermediate Graphic File or the program was halted by abend. Sometimes part of the file might be displayed by repeating the display instruction.

3.1.4 Example of GRAPHIT-Tektronix

The following example of the GRAPHIT-Tektronix elaboration of an existing Intermediate Graphic File uses a part of the available commands.
The instructions of the operator are given in lower case characters and the system responses are written in upper case. The carriage return/ENTER character is marked by a (CR).
The demonstration file, SESPIR on volume COPICB, as used in this example, will remain available to the users for some time to get acquainted with the system.

The sequence of entering the system is as follows:

```
Logon TSOname/password proc(fg1log) (CR)
.....
READY
graphit (CR)
READY
plote sespir copicb (CR)
```

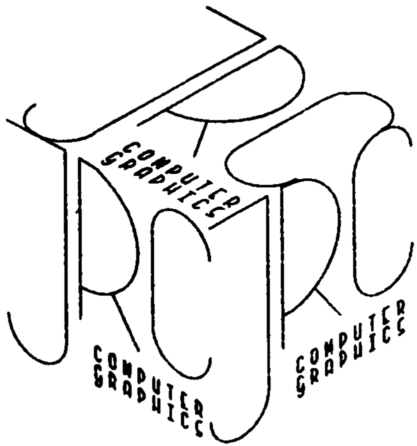
The screen will be cleared and the following message will be displayed:

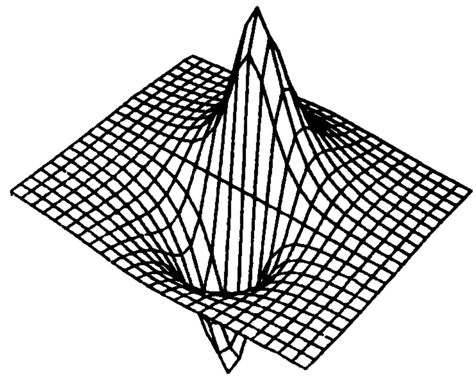
GRAPHIT VERSION NOVEMBER 1979

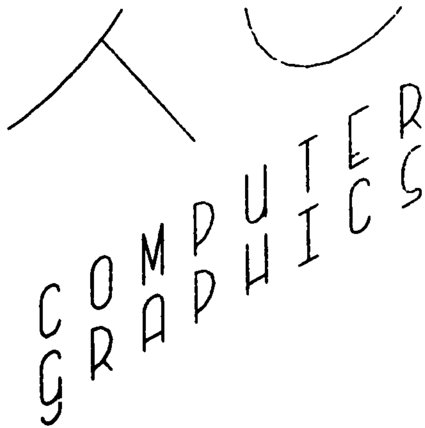
followed by an acoustic signal.
Any instruction to the system is terminated by the bell signal to indicate the availability for following commands.

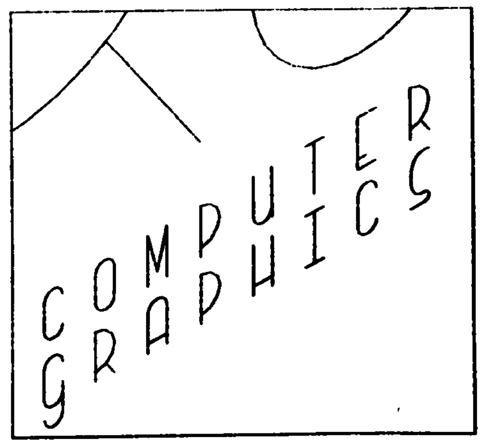
The example shows the system response to the 'summary' command.
The total number of the plotting commands for the entire file is displayed together with the boundary values of the X and Y coordinates. The following picture numbers are defined by the use of the subroutine PICTNO in the application program.
For each of the segments the boundary values are written.
The screen is divided in four subscreens and the segments 8 and 7 are displayed. Then the 'cursor' command is given to define a window on the first drawing. The cursor appears twice for the bottom-left point and for the top-right corner of the requested detail. The 'stack' command displays the noted coordinates and the related name of this cursor defined window.
The window is twice shown, once without the viewport and the second time after the command 'frame'.
The session may be closed by the 'end' instruction.
Any sequence for display requests is allowed.

```
GRAPHIT VERSION NOVEMBER 1979
summary
NUMBER OF COMMANDS           8772
           XMIN     XMAX    YMIN   YMAX
FILE       1.99   176.96    1.99  19.85
     1     2.19    22.01    1.99  19.45
     2    22.19    42.27    1.99  18.73
     3    43.07    62.27    1.99  18.29
     4    62.19    82.27    1.99  19.85
     5    82.19   102.01    1.99  19.19
     6   102.19   122.27    1.99  15.76
     7   122.19   142.27    1.99  19.69
     8   147.04   156.96    6.59  17.73
     9   167.04   176.96    1.99   7.72
noframe
enhanced
screen 4
display 1 8 n
display 2 7 o
cursor a 4 1
stack
WINDOW   XMIN     XMAX    YMIN   YMAX
A       153.91   157.02   7.65  10.70
display 3 a n
frame
display 4 a n
```









3.2 GRAPHIT - Gould 5200 plotter

For the preparation of a tape for the Gould 5200 plotter from the data as supplied by an Intermediate Graphic File, there exist two procedures named GOULD and GOULDX.
The procedure GOULD is for graphic output of less than 3 meters of length.
(Note. Larger production requires the GOULDX procedure together with calls to the subroutine PICTNO in the application program.)

This distinction must be made because of the particular properties of the Gould plotting system. A normal penplotter draws the lines according to the specified coordinates. The Gould plotter elaborates the drawing by consecutive vertical lines in drawing segments of up to 120 inches. This means that continuous drawings of more than 120 inches in the X direction are not possible. However, if the PICTNO definitions of picture segments are used, the GOULDX procedure distinguishes these segments as independent drawings and processes them consecutively.

The procedure is as follows:

```
//      JOB.....
$OC TP9=GRxxxx,SL,Y
$OC M=GOULD T=GRxxxx ABEND=NO F=1
//      EXEC GOULD,TAPE='GRxxxx'
//GLD.FT15F001 DD DSN=TSOxxxx.name,DISP=(OLD,KEEP)
//GLD.SYSIN    DD *
factor, iform, ncopy, int    (format F6.2,3i6)
inp(i)                       (format 12i6)
/*
```

in which:

$OC cards are HASP control cards

factor is the scale factor. If not specified the program takes a default value of 0.32, being the ratio between the paper heigth of the Gould plotter and the Calcomp plotter

iform is the background pattern number
- = 0 white background (default)
- = 1 squared background
- =-1 black background with erased lines

int is the line intensity:
- =-2 triple intensity
- =-1 double intensity
- = 0 normal (default)
- = 1 half intensity
- = 2 one-third intensity

inp(i) are the picture number specifications.
default: All pictures will be drawn.

GRxxxx is the number of the labelled tape (Note - tape must be 9-track, 1600bpi)

TSOxxxx.name is the fully qualified name of the Intermediate
Graphic File. If this file is not catalogued, one must also specify the volume name.

3.3 Graphit - Benson 1102 plotter

The procedure for the processing of the Intermediate Graphic File to a Benson tape is as follows:

```
//    JOB.....
$OC TP9=GRxxxx,SL,Y
$OC M=BENSON T=GRxxxx ABEND=NO F=1
//    EXEC BENSON,TAPE='GRxxxx'
//BNS.FT15f001 DD DSN=TSOxxxx.name,DISP=(OLD,KEEP)
//BNS.SYSIN DD *
factor   (format F6.2)
inp(i)   (format 12i6)
/*
```

in which:

factor is the scaling factor. If omitted the program multiplies all coordinate values by 0.37. This is the ratio between the Benson plotter y-size and the Calcomp plotter y-size

inp(i) are the desired picture numbers as supplied in the application program by the calls to PICTNO
Default: All pictures will be processed.

GRxxxx is the number of the labelled tape with density 1600 BPI.

TSOxxxx.name is the fully qualified name of the Intermediate Graphic File.

4. LINEPRINTER GRAPHICS

Plotting functions on a lineprinter is, of course, a rather crude way of obtaining graphics, but for a first scanning of the results and for input verification it may be useful.
Presently this chapter contains only one routine with a very simple calling sequence for easy use.

4.1 XPRPLT

The calling sequence is as follows:

```
CALL XPRPLT (X,Y,N,M,NY)
```

in which:

X is a vector containing N abscis values
Y is a matrix with dimensions (NY,MA). Each column may contain N ordinate values
N is the number of coordinate pairs with $N \leq NY$
M is the number of curves to be plotted with the limitations $M \leq 10$ and $M \leq MA$
NY is the dimensional column length of matrix Y

The results are printed over 80 columns and may consequently also be used for a printer plot on the 2741 terminal.
The print symbols for the different curves are 1,2,.. etc.
For the crosspoints of the curves the symbol 'M' is printed.
Text information may be added beneath the diagram by a simple WRITE statement.

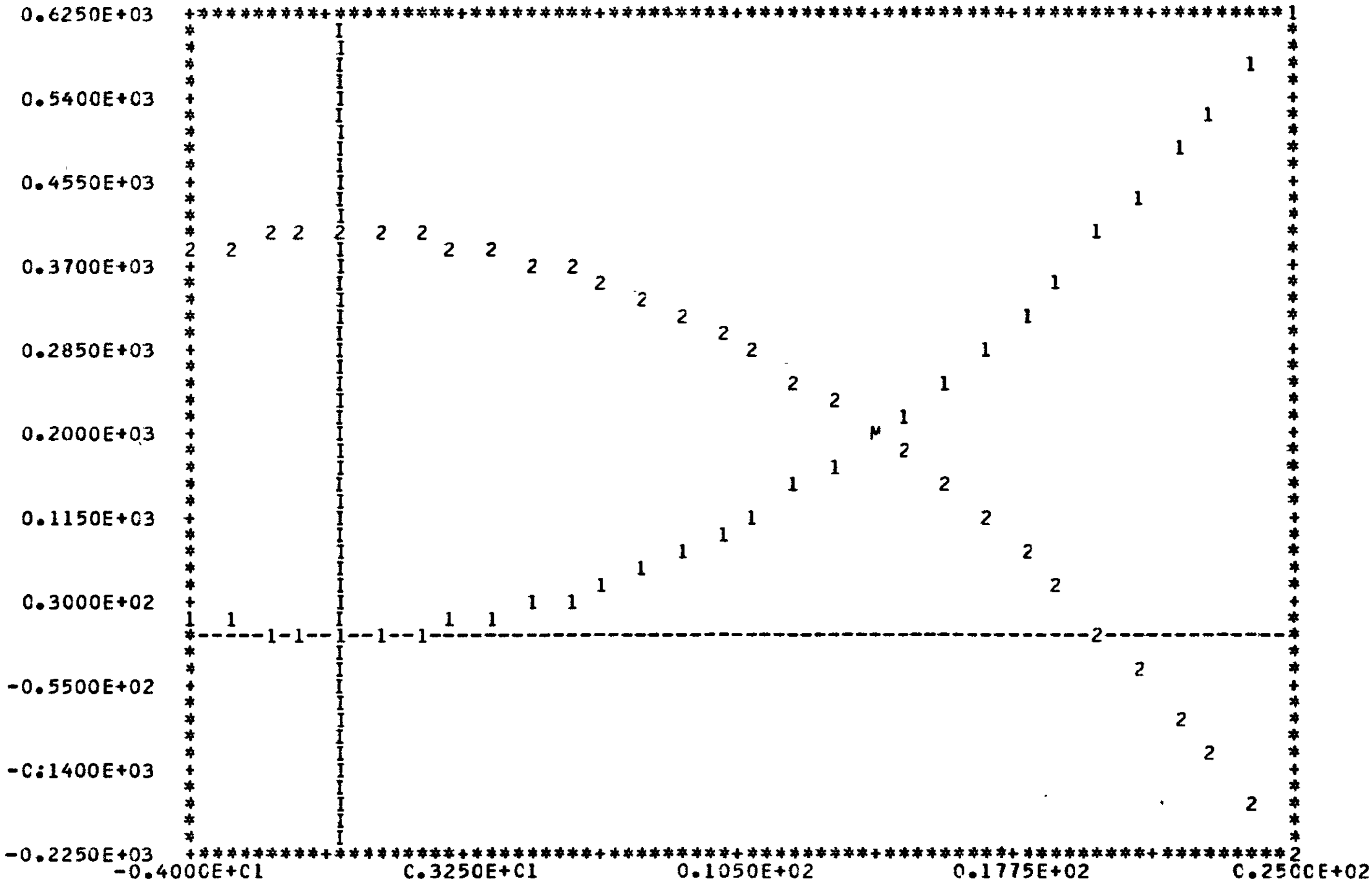
Example of Use in batch

```
//          JOB(YOUR JOB CARD)
$      CLASS 2
//     EXEC  FTG1CG,PRN=ERTY,ULB=DISK,ULB=COPICB
//CMP.SYSIN  DD  *
C      MAIN PROGRAM TO DEMONSTRATE XPRPLT SUBROUTINE
       DIMENSION X(1000),Y(500,20)
       DO 100 I=1,30
       X(I)=FLOAT(I)-5.0
       Y(I,1)=X(I)**2
       Y(I,2)=400.0-Y(I,1)
   100 CONTINUE
       CALL XPRPLT(X,Y,30,2,500)
       WRITE(6,110)
   110 FORMAT(' EXAMPLE OF OUTPUT BY XPRPLT ROUTINE')
       STOP
       END
/*
```

Example of TSO Usage(lines typed by user are shown in lower case)

```
list lpgraph.fort
 LPGRAPH.FORT
00050 C      MAIN PROGRAM TO DEMONSTRATE XPRPLT SUBROUTINE
00060        DIMENSION X(1000),Y(500,20)
00070        DO 100 I=1,30
00080        X(I)=FLOAT(I)-5.0
00090        Y(I,1)=X(I)**2
00100        Y(I,2)=400.0-Y(I,1)
00110    100 CONTINUE
00120        CALL XPRPLT(X,Y,30,2,500)
00130        WRITE(6,110)
00140    110 FORMAT(' EXAMPLE OF OUTPUT BY XPRPLT ROUTINE')
00150        STOP
00160        END
READY
fort lpgraph
G1 COMPILER ENTERED
SOURCE ANALYZED
PROGRAM NAME = MAIN
*  NO DIAGNOSTICS GENERATED
READY
loadgo lpgraph fortlib lib('sys1.liberty')
```

Output (see example on next page) will be produced on terminal.



EXAMPLE OF OUTPUT BY XPRPLT ROUTINE

5. BASIC ROUTINES

This chapter describes some basic routines which are frequently used. Some of these subroutines are from the GINO packages. Others are developed at the JRC to facilitate the graphic tasks and satisfy the requirements of the GRAPHIT system.

5.1 GSTART

The subroutine GSTART must be called before any graphic operation:

CALL GSTART

5.2 GEND

The subroutine GEND must be the last graphic call to close the Intermediate Graphic File properly:

CALL GEND

5.3 PICTNO

The subroutine PICTNO allows the user to segment the Intermediate Graphic File:

CALL PICTNO(K)

in which K are consecutive integer numbers, up to 255.
PICTNO is called before the related segment. The routine stores the X value of the current pen position. At the elaboration of the Intermediate Graphic File, either by means of the Tektronix or at the preparation of a tape for one of the plotter, the user may address a segment by its number. All the pen movements between the coordinates X(K) and X(K+1), will then be plotted.

5.4 SHIFT2

The routine to move the zeropoint of the axis system is as follows:

CALL SHIFT2(DX,DY)

The original zeropoint of the axis system is the left under point of the drawing surface.
DX and DY are the displacements in centimetres (real arguments).
An equivalent shift routine for 3D space will be discussed in the chapter on transformations.

Note. A shift transformation as caused by this routine does not apply to characters unless CHASWI is called, see 5.6.3.1

5.5 Pen movements

Although neither the Tektronix 4015 nor the Gould plotter have actually a "pen", the concepts of preparing an image with some writing instrument, an electronbeam or a series of activated points on the electrostatic writing head, is called a pen movement.

The SYS1.LIBERTY library distinguishes:

- absolute movements
- relative movements

5.5.1 MOVTO2

The calling sequence is:

```
CALL MOVTO2(X,Y)
```

By this command the pen is moved to the coordinates X,Y in centimetres, in relation to the current axis system, which is the initial axis system, (zero point left under of drawing space), modified by calls to SHIFT2 (absolute movement).
During this operation the pen does not write.

5.5.2 MOVBY2

The calling sequence is:

```
CALL MOVBY2(DX,DY)
```

The pen is moved over the distance (DX,DY) without writing (relative movement).

5.5.3 LINTO2

The calling sequence is:

```
CALL LINTO2(X,Y)
```

The pen writes a straight line from the present pen position to the coordinates X,Y in centimetres, in relation to the current axis system (absolute movement).

5.5.4 LINBY2

The calling sequence is:

```
CALL LINBY2(DX,DY)
```

The pen writes a straight line from the present pen position over a distance of DX,DY centimetres (relative movement).

5.6 Characters

Many routines are available for the drawing of character strings. However, for simple applications only a few will be mentioned here. Extensive information is supplied by the GINO-F manual.

5.6.1 CHAHOL

The calling sequence is:

```
CALL CHAHOL(17HWRITE CHARACTER*.)
```

in which:
17 is the number of characters including *., but excluding the character H which stays for "Hollerith".

The routine writes the character string starting from the present pen position. The string should not exceed 140 characters and must be closed by '*.'. After the writing the pen remains at the bottom right-hand corner of the last character.
Additionally one may define upper case and lower case characters:

*L means shift to lower case
*U means shift to upper case

For example:

```
CALL CHAHOL (23HW*LRITE *UC*LHARACTER*.)
```

is displayed as Write Character

5.6.2 CHASIZ

The routine CHASIZ defines the size of the single characters. If this routine is not called, an easy readable size is chosen as the default option.
The calling sequence is:

```
CALL CHASIZ(WIDTH,HEIGHT)
```

in which:
WIDTH and HEIGHT are the standard dimension in centimetres.

5.6.3 Inclination of character strings

The following routines belong in fact to the chapter on transformations but as the inclination of character strings is also frequently required for simple applications, a short description is given here.
Any transformation routine, like SHIFT2, ROTAT2, etc. transforms the axis system. However, character drawing is excluded from these conversions unless explicitely stated.

5.6.3.1 CHASWI

The calling sequence is:

CALL CHASWI(K)

in which:

K=0 default, axis transformations do not apply to character strings

K=1 switches character transformation on.

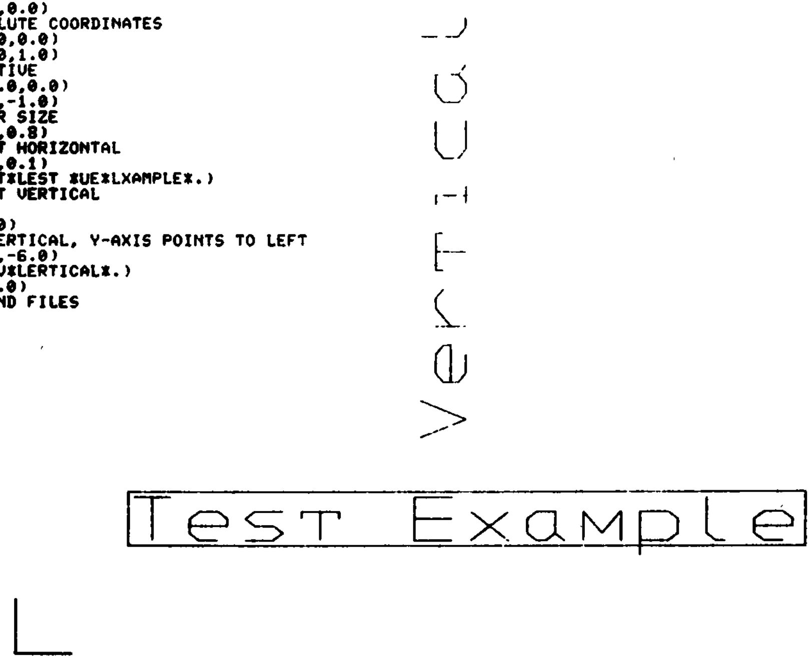
5.6.3.2 ROTAT2

The calling sequence is:

CALL ROTAT2(A)

The axis system is rotated counterclockwise over an angle of A degrees.
All following drawing commands are in relations to the new axis system.
If this routine is only used to write a character string with a certain inclination, the programmer has to reposition the coordinate system afterwards by an opposite rotation.
Note in the following example (5.7) that after the first rotation the X-axis is vertical. This determines the arguments in the subsequent application of the MOVT02 routine.

```
C      PROGRAM TO DEMONSTRATE BASIC GRAPHIC ROUTINES
C      INITIALIZE GRAPHIT
       CALL GSTART
       CALL PICTNO(1)
C      MARK COORDINATES ZERO POINT
       CALL MOVTO2(1.0,0.0)
       CALL LINTO2(0.0,0.0)
       CALL LINTO2(0.0,1.0)
C      MOVE ORIGIN
       CALL SHIFT2(2.0,2.0)
C      DRAW BOX
       CALL MOVTO2(0.0,0.0)
C      DRAW LINES ABSOLUTE COORDINATES
       CALL LINTO2(12.0,0.0)
       CALL LINTO2(12.0,1.0)
C      DRAW LINES RELATIVE
       CALL LINBY2(-12.0,0.0)
       CALL LINBY2(0.0,-1.0)
C      DEFINE CHARACTER SIZE
       CALL CHASIZ(1.0,0.8)
C      CHARACTER OUTPUT HORIZONTAL
       CALL MOVTO2(0.1,0.1)
       CALL CHAHOL(20HT*LEST *UE*LXAMPLE*.)
C      CHARACTER OUTPUT VERTICAL
       CALL CHASWI(1)
       CALL ROTAT2(90.0)
C      X-AXIS IS NOW VERTICAL, Y-AXIS POINTS TO LEFT
       CALL MOVTO2(2.0,-6.0)
       CALL CHAHOL(12HV*LERTICAL*.)
       CALL ROTAT2(-90.0)
C      CLOSE GRAPHIT AND FILES
       CALL GEND
       STOP
       END
```



6. EASY GRAPHICS

A basic problem of computer graphics, next to the rather complex nature of the discipline, is that the average user seems to avoid a profound knowledge of the field, as he considers these facilities as secondary to his main objectives.
To assist such a "casual" user of the graphic facilities, a series of simplified routines have been developed and installed, called the X-collection.

The general characteristics of these routines are:

- The name starts with X
- All displacements of origins for a new image are automatically handled. The user is not obliged to think about the actual placement of a picture on the drawing surface
- The pictures are automatically numbered for easy displaying. Graphs start with picture number 100 and histograms start with picture number 150
- The calling sequences are kept as simple as possible
- Application of additional routines for more sophisticated graphic output remains possible

Presently only 2D graphs and histograms in A4 and A5 format, with logarithmic and linear scales, are possible. In the future the collection will be extended according to user requirements.

6.1 XGRAF4

The calling sequence to produce an A4-format graph is:

```
CALL XGRAF4(X,Y,N,K,L)
```

in which:

X	is a vector with the abscis values
Y	is a vector with the ordinate values
N	is the number of coordinate pairs
K=-1	x-axis logarithmic, y-axis logarithmic
K=0	x-axis linear, y-axis linear
K=1	x-axis logarithmic, y-axis linear
K=2	x-axis linear, y-axis logarithmic
L=0	horizontal A4 format
L=1	vertical A4 format

6.2 XGRAF5

The calling sequence is:

```
CALL XGRAF5(X,Y,N,K,L)
```

The definition of the arguments is the same as for XGRAF4.
The routine produces an A5 format graph.

6.3 XGRAF

The routine XGRAF serves to add an additional curve to the latest graph as set up by either XGRAF4 or XGRAF5.

The calling sequence is:

CALL XGRAF(X,Y,N)

in which:
X and Y are the two vectors containing the N coordinate pairs.
If the values of X and Y exceed the present axis system, the excessive values are omitted.

6.4 XHIST4

The calling sequence is:

CALL XHIST4(Y,N,K,F,XL,XH,L)

This routine produces a histogram in A4 format.

Y is a vector containing the column heights
N is the number of columns
K=0 the y-axis is linear
K=1 the y-axis is logarithmic
F is the width of the column on the interval expressed as fraction: $0<F\leqslant 1.0$; F=1.0 gives a histogram without vertical lines
XL is the X-value low, minimum X value; this value is the centre of the first interval on the X-axis
XH is the X-value high, maximum X value; this value is the centre-value of the last interval of the histogram
L=0 horizontal A4-format
L=1 vertical A4-format

6.5 XHIST5

The calling sequence is:

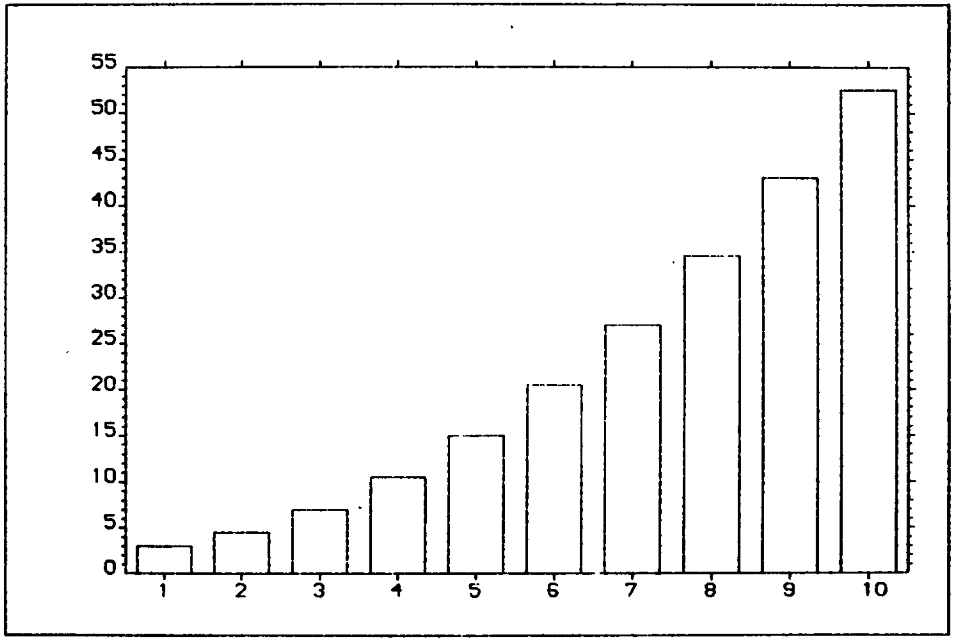
CALL XHIST5 (Y,N,K,F,XL,XH,L)

The definition of the arguments is the same as for XHIST4.
The routine produces an A5 format histogram.

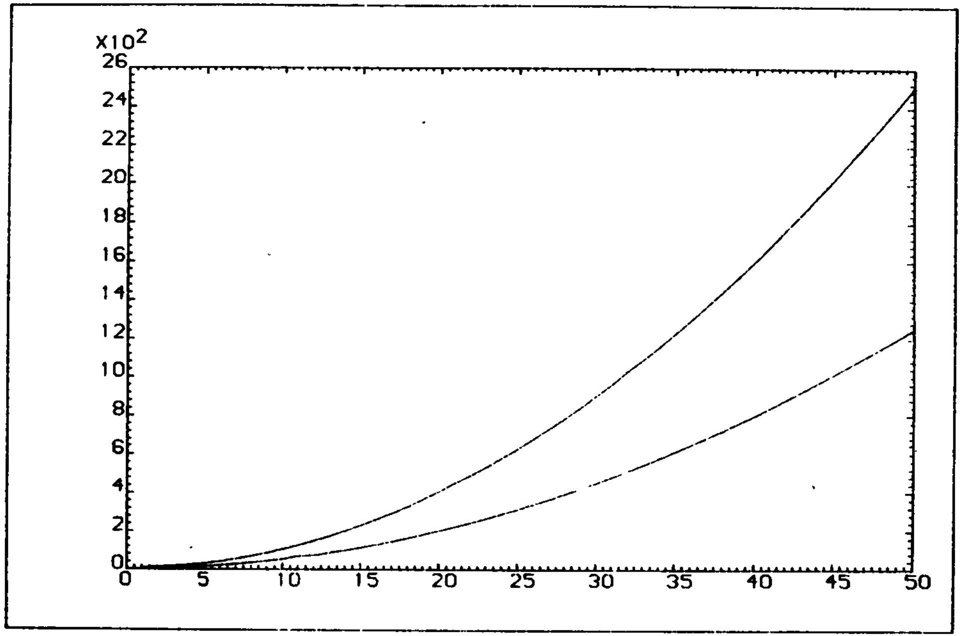
6.6 Example of easy graphics routines

```
C DEMONSTRATION PROGRAM EASY GRAPHICS
      DIMENSION X(100),Y(100)
C     THE CALL TO GSTART IS MANDATORY FOR GRAPHICS APPLICATIONS
      CALL GSTART
C     CONSTRUCT A FUNCTION
      NPOINT=50
      DO 100 I=1,NPOINT
      X(I)=FLOAT(I)
  100 Y(I)=X(I)**2+5.0
C     DRAW A GRAPH AS FORMAT HORIZONTAL, LINEAR X, LINEAR Y
      CALL XGRAF5(X,Y,NPOINT,0,0)
C     CONSTRUCT A SECOND FUNCTION
      DO 120 I=1,NPOINT
  120 Y(I)=0.5*Y(I)
C     ADD THE SECOND FUNCTION TO THE GRAPH AS HORIZONTAL
      CALL XGRAF(X,Y,NPOINT)
C     DRAW A HISTOGRAM AS FORMAT HORIZONTAL, LINEAR Y
      CALL XHIST5(Y,10,0,0.7,1.0,10.0,0)
C     THE CALL TO GEND IS MANDATORY TO END GRAPHICS APPLICATION
      CALL GEND
      STOP
      END
```

```
GRAPHIT VERSION NOVEMBER 1979
enhanced
noframe
summary
NUMBER OF COMMANDS       1904
        XMIN    XMAX   YMIN   YMAX
FILE   -0.01   43.01  -0.01  14.96
 100   -0.01   21.51  -0.01  14.96
 150   21.59   43.01   0.09  14.96
display 1 150 n
```



display 1 400 n



7. EASY GRAPHICS, MORE SOPHISTICATED

The routines of the previous chapter are designed for a quick and easy checking of data. For example, large quantities of input data may be verified by some simple graphs.
To maintain the simplicity, no options are available in the X-collection for the addition of titles or other comments on the graphic output.
If the programmer wishes to produce more sophisticated graphs and histograms, he will have to study the GINOGRAF manual, where all facilities for the production of graphs, histograms, bar charts, etc. are specified.
However, as all the options of the entire SYS1.LIBERTY library are integrated, one may combine other subroutines with those of the X-collection to obtain a more complete output.

In this chapter an example is given in which alphanumeric information is added to an image as produced by the XGRAF5 routine. Equivalent procedures may be applied for the drawings by the XHIST routines.

7.1 GRASPA

The routine GRASPA serves to find the coordinates in centimetres of a point in the previously constructed axis system.

The calling sequence is:

```
CALL GRASPA(FX,FY,X,Y)
```

in which

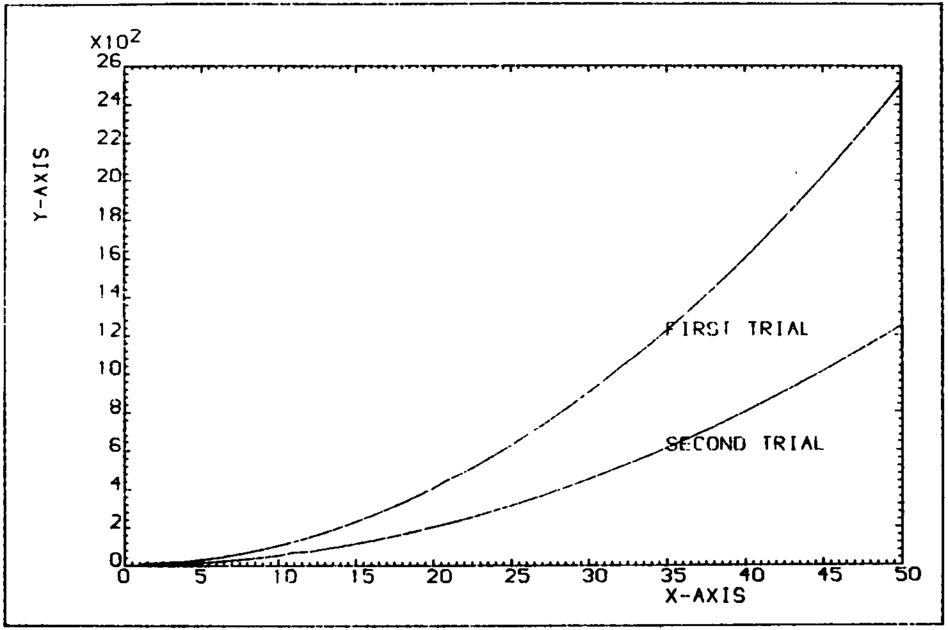
FX and FY are a pair of function value coordinates of the most recent set-up graph

X and Y are the resulting coordinates in centimetres

Once GRASPA has been called, the user may move the pen to the calculated coordinates X,Y and consequently write some alphanumeric information by the routine CHAHOL.
The necessary routines are described in the chapter on basic routines. The next paragraph supplies an example.
Please study carefully the exigency of the arguments in the statement CALL MOVTO2(YR,-XR).
The same routine GRASPA has to be used to memorize the pen position before title writing. Once the titles have been written, the pen must be moved to the stored coordinates, for the following graphs.

```
C DEMONSTRATION OF EASY GRAPHICS, MORE DIFFICULT
      DIMENSION X(100),Y(100)
C     THE CALL TO GSTART IS MANDATORY FOR GRAPHICS APPLICATIONS
      CALL GSTART
C     CONSTRUCT A FUNCTION
      NPOINT=50
      DO 100 I=1,NPOINT
      X(I)=FLOAT(I)
  100 Y(I)=X(I)**2+5.0
C     DRAW A GRAPH AS FORMAT HORIZONTAL, LINEAR X, LINEAR Y
      CALL XGRAF5(X,Y,NPOINT,0,0)
C     CONSTRUCT A SECOND FUNCTION
      DO 120 I=1,NPOINT
  120 Y(I)=0.5*Y(I)
C     ADD THE SECOND FUNCTION TO THE GRAPH AS HORIZONTAL
      CALL XGRAF(X,Y,NPOINT)
C     EASY GRAPHICS, MORE DIFFICULT, ADD TITLES
C     THE PEN POSITION HAS TO BE STORED BECAUSE TITLES
C     ADDING DISTURBS THE SEQUENCE FOR FOLLOWING DRAWINGS
      CALL POSSPA(XST,YST,ZST)
      CALL GRASPA(35.0,1200.0,XQ,YQ)
      CALL GRASPA(35.0,600.0,XR,YR)
      CALL MOVTO2(XQ,YQ)
      CALL CHAHOL(13HFIRST TRIAL*.)
      CALL MOVTO2(XR,YR)
      CALL CHAHOL(14HSECOND TRIAL*.)
      CALL GRASPA(35.0,-200.0,XQ,YQ)
      CALL GRASPA(-5.0,1800.0,XR,YR)
      CALL MOVTO2(XQ,YQ)
      CALL CHAHOL(8HX-AXIS*.)
      CALL CHASWI(1)
      CALL ROTAT2(90.0)
      CALL MOVTO2(YR,-XR)
      CALL CHAHOL(8HY-AXIS*.)
      CALL ROTAT2(-90.0)
C     RESET PEN ON POSITION BEFORE TITLE DRAWING
      CALL MOVTO2(XST,YST)
      CALL GEND
      STOP
      END
```

```
GRAPHIT VERSION NOVEMBER 1979
summary
NUMBER OF COMMANDS        1477
        XMIN    XMAX   YMIN   YMAX
FILE   -0.01   21.51  -0.01  14.96
 100   -0.01   21.51  -0.01  14.96
noframe
enhanced
display 1 100 n
```



8. 2D FUNCTION GRAPHS, GINOGRAF

GINOGRAF offers facilities for producing graphs, histograms, bar charts and pie charts using one of two different methods. With the first method, the graph is produced by a simple single call routine which perfoms automatically the scaling and annotation. It may be called "Easy GINOGRAF".
However, compared to the facilities of the previously described Easy Graphics, the user has to take care of size definitions, replacements in the field of drawing and numbering of the individual segments.

The second method, called "Sophisticated GINOGRAF", offers an extensive number of options to produce a sophisticated graphic output.
As all the available routines are described in the GINOGRAF manual, this chapter will limit the descriptions to some examples and an additional feature which is possible through the in-house developed GREYSI routine.

The examples in the GINO-F manual should be used with care as there are some errors in them.

8.1 Easy GINOGRAF

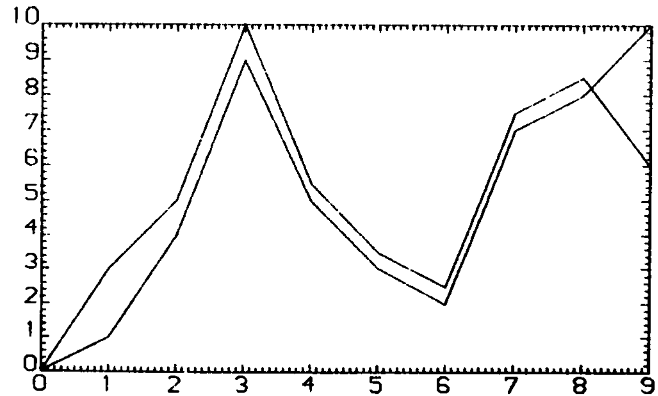
The following routines are available to produce complete graphic pictures:

- GRAF - complete X,Y graphs
- GRAPOL - add a curve to last X,Y graph
- HISCHA - complete histogram drawing
- GRAHIS - add a histogram to last complete histogram drawing
- BARCHA - complete barchart drawing
- GRABAR - add second bar representation to last complete barchart
- PIECHA - complete piechart

The descriptions of these routines are found in the GINOGRAF manual. The actual place in the drawing area is specified by the routine WINDO2. After the application one has to call the routine WINDOW(0) to switch windowing off. Because all displaying outside a specified window is nullified.

Caution: At present there is a restriction to the following use of the routines specified. The above mentioned routines cannot be used together with SHIFT2 calls in the application program.

```
C DEMONSTRATION PROGRAM EASY GINOGRAF
C
      DIMENSION X(20),Y(20),Z(20)
      DATA X/0.0,1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0/
      DATA Y/0.0,1.0,4.0,9.0,5.0,3.0,2.0,7.0,8.0,10.0/
      DATA Z/0.0,3.0,5.0,10.0,5.5,3.5,2.5,7.5,8.5,6.0/
      CALL GSTART
      CALL PICTNO(1)
C     SET UP SPACE FOR THE GRAPH
      CALL WINDO2(0.0,15.0,0.0,10.0)
C     SET UP AXIS SYSTEM AND DRAW FIRST GRAPH
      CALL GRAF(X,Y,10,0)
C     ADD SECOND CURVE
      CALL GRAPOL(X,Z,10)
      CALL GEND
      STOP
      END
```



8.2 Sophisticated GINOGRAF

GINOGRAF offers an extensive set of subroutines to compose sophisticated 2D representation of graphs, histograms and barcharts.
Each of the following actions requires a separate subroutine call:

- Define the place of the axis on the drawing surface
- Define the scaling of the axis
- Define the grid and draw the axis
- Draw the curve
- Add titles

The examples as given in the next paragraph, show the flexibility of the system. Again, for the full description of the subroutines we refer to the GINOGRAF manual.
The first example shows a graph with linear scales and two curves; one obtained by straight lines between the reference points, the other by smoothing of the curve by cubic spline. The size of the characters may be changed by the routine CHASIZ.
The routine GRASYM puts the symbols on the reference points: A set of 8 different centralized symbols is available. However if this set is too small for certain applications, you may retrieve the coordinates of the reference points by the routine GRASPA(FX,FY,X,Y) in which FX,FY are function coordinates values and X and Y are resulting coordinates in centimetres of the reference point. A move to (X,Y) and subsequent writing of any appropriate character gives equivalent results.
The same technique is used to add labels to the drawing. This procedure has been explained in section 7.

The routine GRID takes care of the drawing of the axis system and background pattern, if wanted. However, two calls to AXIDRA, respectively for the X-axis and Y-axis may produce a graph without the frame: Only the axis are drawn in this case.
Example 3 shows the dangers of smoothing; when the variations in Y-values are large, the cubic spline smoothing is absolutely unrealistic. Better results may be obtained by the introduction of additional points by linear interpolation.
Example 4 shows a more complicated histogram; two sets of column heights are represented in one drawing. The model is given here as an illustration of the flexibility of the system. The function values of the bars are converted into centimetres on the drawing surface by the routine GRASPA. The shading of the columns is done by the routine GREYSI, which does not belong to GINOGRAF set.

8.2.1 GREYSI

The GREYSI routine shades polygons of any type, convex, concave or extra concave, with parallel lines.

The calling sequence is:

```
CALL GREYSI(X,Y,N,NL,ALFA,K)
```

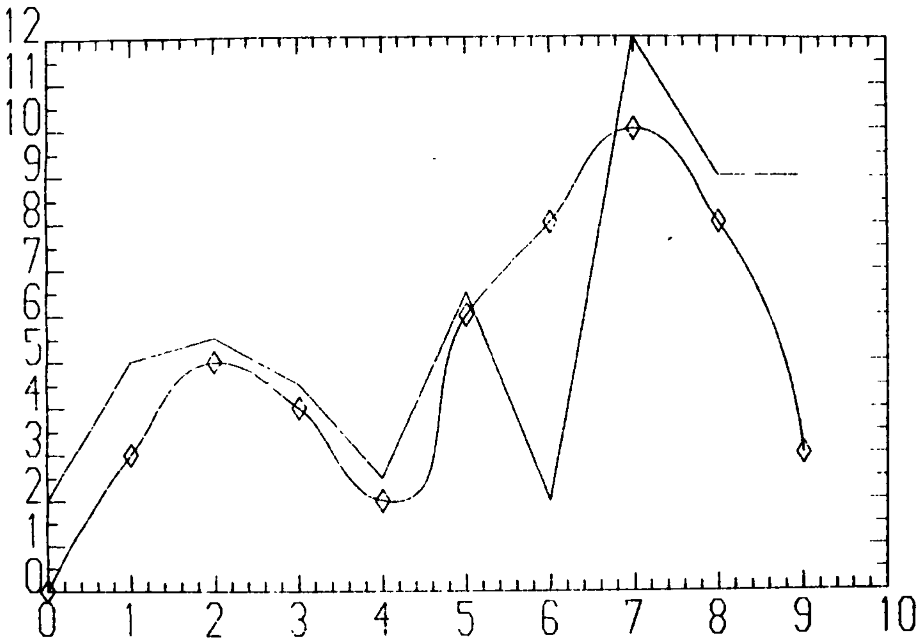
in which:

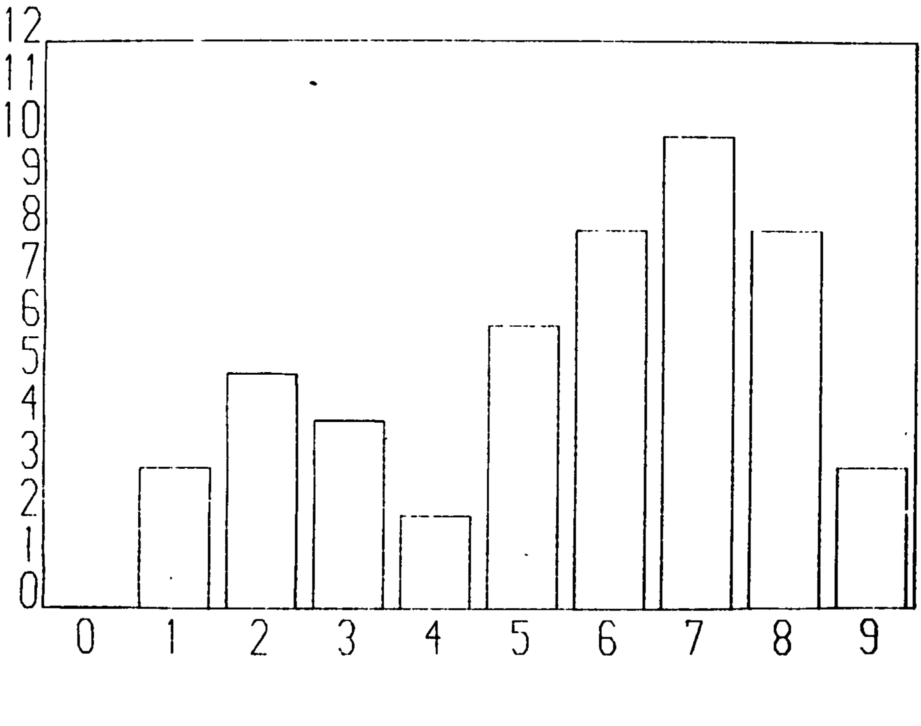
X and Y	are two vectors containing the coordinates in centimetres of the vertices
N	is the number of vertices
NL	is the number of parallel lines per centimetre
ALFA	is the angle in degrees with positive X-axis of the parallel shading lines
K=0	no polygon contour is drawn
K=1	the polygon contour is also displayed

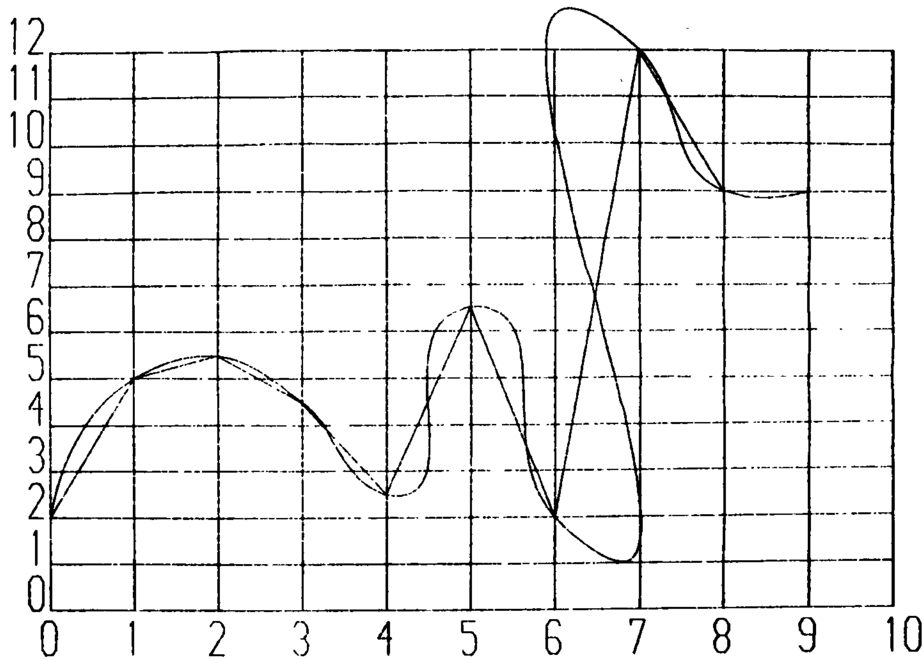
```
C PROGRAM TO DEMONSTRATE SOPHISTICATED GINOGRAF
C
      DIMENSION X(10),Y(10),Z(10)
      DATA X 0.0,1.0,2.0,3.0,4.0,5.0,6.0,7.0,8.0,9.0/
      DATA Y 0.0,3.0,5.0,4.0,2.0,6.0,8.0,10.0,8.0,3.0/
      DATA Z 2.0,5.0,5.5,4.5,2.5,6.5,2.0,12.0,9.0,9.0/
      CALL GSTART
C     DEFINE CHARACTER SIZE
      CALL CHASIZ(0.15,0.25)
C
C** EXAMPLE 1 **********
C
C     DEFINE PLACE OF AXIS IN FIELD OF DRAWING
      CALL PICTNO(1)
      CALL AXIPOS(0,2.0,2.0,6.0,1)
      CALL AXIPOS(0,2.0,2.0,4.0,2)
C     DEFINE SCALING
      CALL AXISCA(1,9,0.0,10.0,1)
      CALL AXISCA(1,12,0.0,12.0,2)
C     DEFINE GRID AND DRAW AXIS
      CALL GRID(-2,1,1)
C     DRAW THE CURVES WITH SMOOTHING AND WITH STRAIGHT LINES
      CALL GRACUR(X,Y,10)
      CALL GRAPOL(X,Z,10)
C     PUT SYMBOLS ON Y-POINTS
      CALL GRASYM(X,Y,10,6,0)
C
C** EXAMPLE 2 **********
C
C     DEFINE HISTOGRAM AXIS IN FIELD OF DRAWING
      CALL SHIFT2(12.0,0.0)
      CALL PICTNO(2)
      CALL AXIPOS(1,0.0,2.0,6.0,1)
      CALL AXIPOS(1,0.0,2.0,4.0,2)
C     DEFINE SCALING
      CALL AXISCA(5,10,0.0,9.0,1)
      CALL AXISCA(1,12,0.0,12.0,2)
C     DRAW A HISTOGRAM
      CALL GRAHIS(Y,10,0.8)
      CALL GRID(0,1,1)
```

```
C
C** EXAMPLE 3 **********
C
C     DEFINE AXIS IN FIELD OF DRAWING
      CALL SHIFT2(12.0,0.0)
      CALL PICTNO(3)
      CALL AXIPOS(0,2.0,2.0,6.0,1)
      CALL AXIPOS(0,2.0,2.0,4.0,2)
C     DEFINE SCALING
      CALL AXISCA(1,9,0.0,10.0,1)
      CALL AXISCA(1,12,0.0,12.0,2)
C     DEFINE GRID AND DRAW AXIS
      CALL GRID(3,1,1)
C     DRAW THE CURVE WITH SMOOTHING
      CALL GRACUR(X,Z,10)
      CALL GRAPOL(X,Z,10)
C
C** EXAMPLE 4 **********
C
C     DEFINE AXIS IN FIELD OF DRAWING
      CALL SHIFT2(12.0,0.0)
      CALL PICTNO(4)
      CALL AXIPOS(1,0.0,2.0,6.0,1)
      CALL AXIPOS(1,0.0,2.0,4.0,2)
C     DEFINE SCALING
      CALL AXISCA(5,10.0.0,9.0,1)
      CALL AXISCA(1,12,0.0,12.0,2)
C     DRAW TWO HISTOGRAMS
      CALL GPID(2,1,1)
      CALL TWOHIS(Y,Z,10,0.0,9.0,0.9,0.3,10,20,45.0,135.0)
C     CLOSE FILES
      CALL GEND
      STOP
      END
```

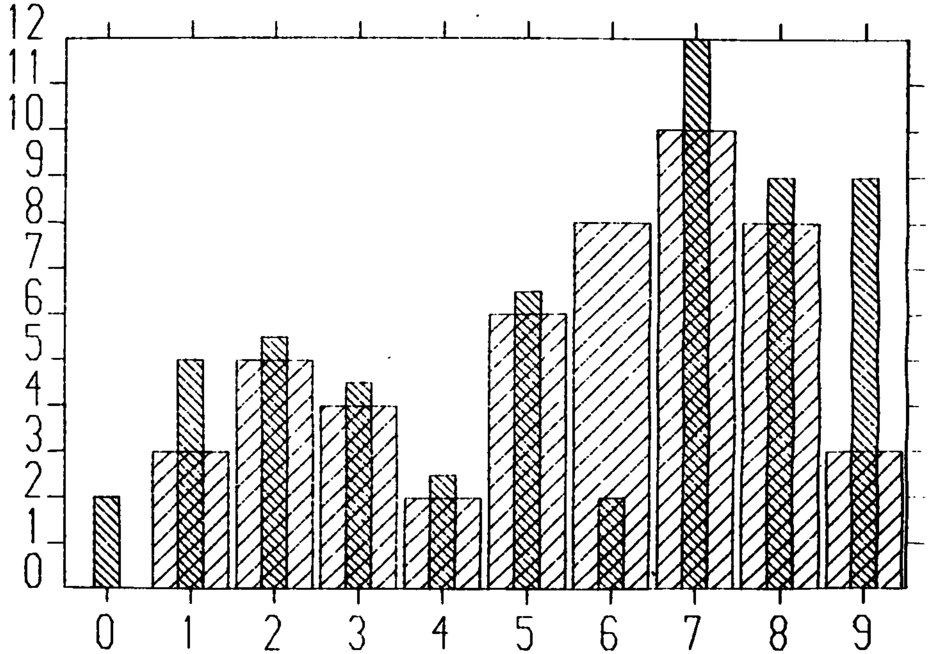
```
      SUBROUTINE TWOHIS(YA,ZA,NINTS,VBEG,VEND,
     1F1,F2,N1,N2,ALFA1,ALFA2)
C     TWOHIS DRAWS TWO SETS OF HISTOGRAMS
C     IN A PRESET AXIS SYSTEM
C     YA AND ZA ARE THE COLUMN HEIGHT VECTORS
C     NINTS IS NUMBER OF INTERVALS
C     VBEG AND VEND ARE EXTREMES OF X-AXIS
C     F1 AND F2 ARE COLUMN WIDTHS AS FRACTION ON INTERVAL
C     N1 AND N2 ARE PARALLEL SHADING LINES PER CENTIMETER
C     ALFA1 AND ALFA2 ARE INCLINATION OF SHADING LINES
      DIMENSION YA(1),ZA(1),X(5),Y(5),XX(5),YY(5),NN(2),F(2),ALFA(2)
      F(1)=F1
      F(2)=F2
      NN(1)=N1
      NN(2)=N2
      ALFA(1)=ALFA1
      ALFA(2)=ALFA2
      DELTA=(VEND-VBEG)/FLOAT(NINTS-1)
      DO 150 K=1,2
      DO 140 I=1,NINTS
      X(1)=FLOAT(I-1)*DELTA+VBEG-0.5*DELTA+(1.0-F(K))*0.5
      X(2)=FLOAT(I)*DELTA+VBEG-0.5*DELTA-(1.0-F(K))*0.5
      X(3)=X(2)
      X(4)=X(1)
      X(5)=X(1)
      Y(1)=0.0
      Y(2)=0.0
      Y(3)=YA(I)
      IF(K.EQ.2) Y(3)=ZA(I)
      Y(4)=Y(3)
      Y(5)=Y(1)
      DO 100 J=1,5
  100 CALL GRASPA(X(J),Y(J),XX(J),YY(J))
      NP=5
      CALL GREYSI(XX,YY,NP,NN(K),ALFA(K),1)
  140 CONTINUE
  150 CONTINUE
      RETURN
      END
```


12
11
10
9
8
7
6
5
4
3
2
1
0
0
1
2
3
4
5
6
7
8
9
10


12
11
10
9
8
7
6
5
4
3
2
1
0
0
1
2
3
4
5
6
7
8
9



12
11
10
9
8
7
6
5
4
3
2
1
0
0
1
2
3
4
5
6
7
8
9
10



12
11
10
9
8
7
6
5
4
3
2
1
0
0
1
2
3
4
5
6
7
8
9

9. GINOZONE MAPPING

The package GINOZONE is an integral part of the SYS1.LIBERTY library.
It has been designed for mapping problems including geographical statistical data representation.
As of yet no tests have yet been performed at the Computing Centre.
However, interested parties may consult the GINOZONE manual and apply the package.
In a future version of this document information will be included about the use of GINOZONE.

10. 3D COMPUTER GRAPHICS

The development of graphic representation in three dimensions is a rather complicated task. Especially as the consequences of the transformations in 3D space are frequently difficult to handle. The programmer is strongly advised:

- to keep separated the actual application program and the adjoined graphical part
- to develop the graphics part step by step and to check the results as frequently as possible
- to keep the subroutine calls for transformations as elementary as possible

The development of 3D graphic results requires the availability of a graphic terminal; the turn-around time for plotter output is too long for testing. As an example, the illustrative programs in the next paragraphs required about 40 times a compilation and execution to obtain the graphic results.

10.1 GINO-F

The GINO-F packages contain the basic routines for 2D plotting and all the higher level routines for transformations in space. It is advisable for the first trials in 3D not to use all the sophisticated options such as switching transformations on and off.

Two basically different methods are available for 3D plotting:

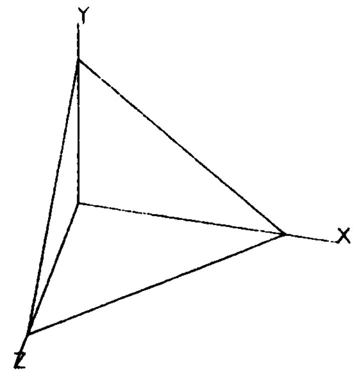
- Working on a fixed coordinate system and commanding the pen movements in 3D space as is done in the first example. However, for complicated drawings such procedure is not advisable because of the unnecessary computing of all the coordinates.
- Designing in 2D those parts of the projected object which occur in planes, followed by assembling of the 2D elements by the available transformations. The manipulations with the JRC computer graphics emblem is an example of this method.

Both methods may of course be mixed. Furthermore an assemblage of 2D elements into a 3D object, may in turn be used as an entity. The whole object may be shifted, rotated, sheared and scaled without any modification of the actual subroutine.
A detailed discussion on all aspects of GINO-F in the context of this document would be superfluous as the description is already given in the GINO-F manual. However, we strongly advise any programmer who wishes to use the package, to start with the given examples and to experiment by adding additional features.

10.1.1 Example of GINO-F application

The given example is a very simple illustration of direct 3D plotting. All commands for the pen movements in 3D space may be used.
Only one transformation is necessary, (CALL SHIFT3), to position the origin more towards the centre of the drawing surface as the default axis-system has the origin at the left under point of the drawing surface with the z-axis perpendicular to this plane.
The routine AXON3 defines the view point.

```
C      PROGRAM TO DEMONSTRATE 3D COMPUTER GRAPHICS
       CALL GSTART
       CALL SHIFT3(2.0,5.0,0.0)
       CALL MOVTO3(0.0,0.0,0.0)
       CALL AXON3(5.0,20.0,20.0)
       CALL CARTES(5.0)
       CALL MOVTO3(4.0,0.0,0.0)
       CALL LINTO3(0.0,4.0,0.0)
       CALL LINTO3(0.0,0.0,4.0)
       CALL LINTO3(4.0,0.0,0.0)
       CALL GEND
       STOP
       END
       SUBROUTINE CARTES(SIZ)
C      CARTES DRAWS THE PRESENT AXIS SYSTEM
       CALL MOVTO3(0.0,0.0,0.0)
       CALL LINBY3(SIZ,0.0,0.0)
       CALL CHAHOL(3HX*.)
       CALL MOVTO3(0.0,0.0,0.0)
       CALL LINBY3(0.0,SIZ,0.0)
       CALL CHAHOL(3HY*.)
       CALL MOVTO3(0.0,0.0,0.0)
       CALL LINBY3(0.0,0.0,SIZ)
       CALL CHAHOL(3HZ*.)
       CALL MOVTO3(0.0,0.0,0.0)
       RETURN
       END
```

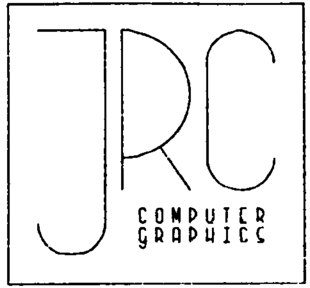


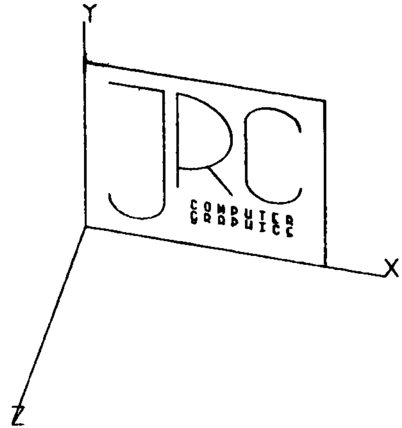
10.1.2 Example 2 of GINO-F application

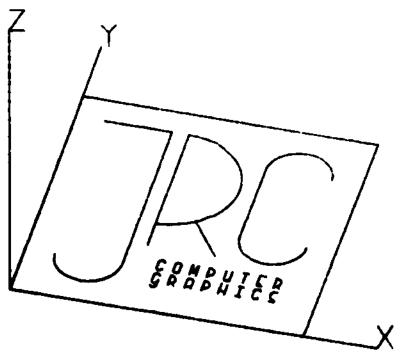
The centre routine JRC is a subroutine programmed in a 2D space, which resides in the SYS1.LIBERTY library. After chosing a particular place for the axis system, performed by a call to SHIFT3, and the definition of the viewpoint by AXON3, the emblem is shown in 3D space. Variations in the arguments of AXON3 will produce different pictures.
The 2D picture remains fixed to its place in the X-Y plane. A rotation by ROTAT3 on the X-axis over -90 degrees shows that the whole axis system is rotated; the z-axis points now to the top of the drawing surface and the y-axis points backwards.
The routine JRCBOX is an entity; it may be called at any moment, after fixing the place of the axis system and the viewpoint. The consecutive movements of the complete axis system including the actual emblem, may be followed by the sequence of calls to the subroutines.

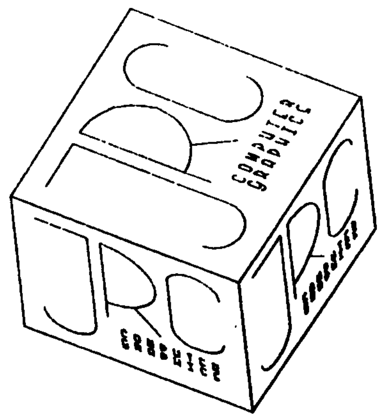
```
C  PROGRAM TO DEMONSTRATE 3D COMPUTER GRAPHICS
      CALL GSTART
      CALL SHIFT3(2.0,5.0,0.0)
      CALL MOVTO3(0.0,0.0,0.0)
C     SHOW THE EMBLEM IN 2D
      CALL PICTNO(1)
      CALL JRC(4.0)
      CALL SHIFT3(5.0,0.0,0.0)
C     SHOW THE EMBLEM IN 3D
      CALL PICTNO(2)
      CALL SHIFT3(1.0,1.0,0.0)
      CALL WINDOW(3)
      CALL MOVTO3(0.0,0.0,0.0)
      CALL AXON3(5.0,20.0,20.0)
      CALL JRC(4.0)
      CALL CARTES(5.0)
      CALL SHIFT3(5.0,0.0,0.0)
C     ROTATE EMBLEM ON X-AXIS
      CALL PICTNO(3)
      CALL SHIFT3(1.0,1.0,0.0)
      CALL MOVTO3(0.0,0.0,0.0)
      CALL ROTAT3(1,-90.0)
      CALL JRC(4.0)
      CALL CARTES(5.0)
C     ROTATE TO INITIAL POSITION
      CALL ROTAT3(1,90.0)
C     DRAW EMBLEM BOX
      CALL SHIFT3(5.0,6.0,0.0)
      CALL PICTNO(4)
      CALL SHIFT3(4.0,0.0,0.0)
C     PARALLEL PROJECTION
      CALL AXON3( 5.0,5.0,20.0)
      CALL JCRBOX(4.0)
      CALL GEND
      STOP
      END
```

```
      SUBROUTINE CARTES(SIZ)
C     CARTES DRAWS THE PRESENT AXIS SYSTEM
      CALL MOVTO3(0.0,0.0,0.0)
      CALL LINBY3(SIZ,0.0,0.0)
      CALL CHAHOL(3HX*.)
      CALL MOVTO3(0.0,0.0,0.0)
      CALL LINBY3(0.0,SIZ,0.0)
      CALL CHAHOL(3HY*.)
      CALL MOVTO3(0.0,0.0,0.0)
      CALL LINBY3(0.0,0.0,SIZ)
      CALL CHAHOL(3HZ*.)
      CALL MOVTO3(0.0,0.0,0.0)
      RETURN
      END
      SUBROUTINE JCRBOX(SIZ)
C     JCRBOX PRODUCES A 3D EMBLEM BASED ON 2D DRAWING
      CALL MOVTO3(0.0,0.0,0.0)
C     SHIFT 2D EMBLEM PARALLEL TO X-Y PLANE
      CALL SHIFT3(0.0,0.0,SIZ)
C     DRAW 2D EMBLEM
      CALL MOVTO3(0.0,0.0,0.0)
      CALL JRC(SIZ)
C     ROTATE ON Y-AXIS
      CALL ROTAT3(2,90.0)
C     SHIFT  PARALLEL TO X-Y PLANE
      CALL SHIFT3(0.0,0.0,SIZ)
C     DRAW 2D EMBLEM
      CALL MOVTO3(0.0,0.0,0.0)
      CALL JRC(SIZ)
C     NOW TOP OF BOX
C     ROTATE ON X-AXIS
      CALL ROTAT3(1,-90.0)
C     MOVE TO TOP OF BOX
      CALL SHIFT3(0.0,0.0,SIZ)
C     DRAW 2D EMBLEM
      CALL MOVTO3(0.0,0.0,0.0)
      CALL JRC(SIZ)
C     RETURN AXIS SYSTEM TO SITUATION ON ENTERING SUBROUTINE
      CALL SHIFT3(0.0,0.0,-SIZ)
      CALL ROTAT3(1,90.0)
      CALL SHIFT3(0.0,0.0,-SIZ)
      CALL ROTAT3(2,-90.0)
      CALL SHIFT3(0.0,0.0,-SIZ)
      RETURN
      END
```


JRC
COMPUTER
GRAPHICS


Y
X
Z
JRC
COMPUTER
GRAPHICS


Z
Y
X
JRC
COMPUTER
GRAPHICS


JRC
COMPUTER
GRAPHICS

10.2 3D functions with hidden line removal

The SYS1.LIBERTY library contains two subroutines for the representation of three dimensional functions with hidden line removal.
The first one, named SRFACE, has been obtained from the National Centre for Atmospheric Research, Boulden, Colorado, USA.
It is a very sophisticated routine with many options. However as the preparation of the input data is rather complicated and error prone, it will not be described here in detail. Programmers who are looking for more options as are given by the more simple routine CARPET, may contact the Support Group for the description of the SRFACE routine.

10.2.1 CARPET

CARPET is a simplified version of SRFACE, with the GINO-F definition of the axis-system. The drawing surface is the X-Y plane and the z-axis is perpendicular to the drawing surface in the direction of the observer.

The calling sequence is:

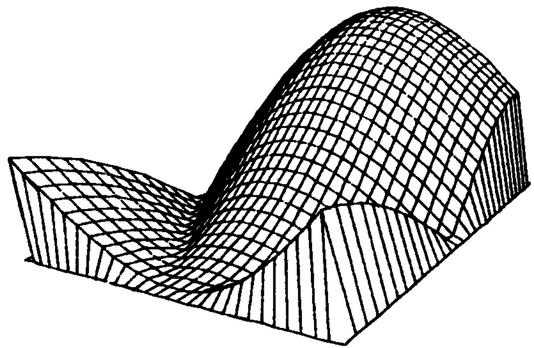
```
CALL CARPET(X,Z,Y,NX,NZ,S,SIZE)
```

in which:

- X — is a linear array with dimension NX, containing strictly by monotonic increasing X values. Equal spacing is not required.
- Z — is a linear array with dimension NZ containing the values of the second independent variable.
- Y — is a two-dimensional matrix with dimensions NX and NZ, containing the function values. The Y values must be of the same magnitude as the X and Z values.
- S — is a linear array, dimension 3, containing the coordinates of the viewpoint. The line of sight is S -(0,0,0). The best pictures are obtained if the distance is 5 to 10 times the size of the projected body.
- SIZE — the image will be drawn approximately in a square with sides of SIZE centimetres. The position of the pen before drawing, will be the left-under corner of the square.
- NX,NZ — are the dimensions of the arrays X,Z and the matrix Y as previously specified. The arrays and the matrix should be dimensioned with exactly these sizes in the calling sequence. It is not possible to plot only a part of a larger Y-matrix.
  Limitations:
  $NX \leq 50$
  $NX*NZ < 1000$

The subroutine CARPET plots the 3D function with the hidden lines removed. If the programmer wants to draw a "skirt", he may add zero values around the Y matrix as is done in the example.

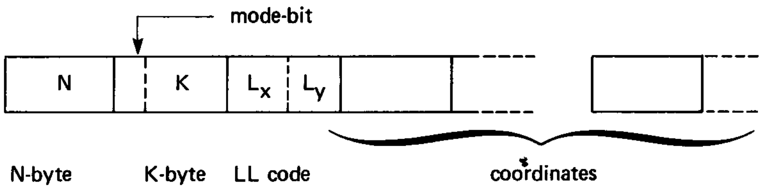
```
C     3D FUNCTIONS WITH HIDDEN LINE REMOVAL
      DIMENSION X(30),Z(30),Y(30,30),S(3)
      DATA S/5.0,2.0,5.0/
      CALL GSTART
      N=30
      EN=FLOAT(N-1)
      DEL=3.14159/EN
      DO 100 I=1,N
      X(I)=FLOAT(I-1)*DEL
  100 Z(I)=FLOAT(I-1)*DEL
      DO 110 I=1,N
      DO 110 J=1,N
  110 Y(I,J)=ABS(SIN(X(I))+COS(Z(J)))
      DO 105 I=1,N
      Y(1,I)=0.0
      Y(I,1)=0.0
      Y(N,I)=0.0
  105 Y(I,N)=0.0
      CALL CARPET(X,Z,Y,N,N,S,20.0)
      CALL GEND
      STOP
      END
```



Appendix: A

STRUCTURE OF THE INTERMEDIATE FILE

The basic elements of the Intermediate File are expressed in bytes. The shortest element is one byte. Other functions need up to nine bytes. The general configuration is as follows:



The N-byte contains a fixed number which indicates the particular meaning of the function and defines as such also the number of associated bytes. There are thus 255 different functions possible of which, for the here presented configuration, only 5 are used.

* N=1 The function defines a penmovement from the present position to a position as specified by the x,y coordinates.
The first bit of the K-byte is called the "mode-bit".
If the mode bit is zero, the coordinates are specified incremental.
If the mode bit is one, the coordinates are given as absolute values in relation to the lower-left angle of the drawing surface.
The value of K, without the mode-bit, defines the pen mode:

K=2 pen down (writing)
K=3 pen up (no writing)
K=0 no modification in pen mode

The LL code in the third byte defines the number of bytes for respectively the x coordinates and y coordinates. If the coordinates are defined as incremental (mode bit in K byte is zero), the LL code contains also the sign according to the following table

S_y	L_y		S_x	L_x					
					1	1	1 byte	x	pos
				1		2	2 bytes	x	pos
				1	1	3	3 bytes	x	pos
			1		1	5	1 byte	x	neg
			1	1		6	2 bytes	x	neg
			1	1	1	7	3 bytes	x	neg
		1				16	1 byte	y	pos
	1					32	2 bytes	y	pos
	1	1				48	3 bytes	y	pos
1		1				80	1 byte	y	neg
1	1					96	2 bytes	y	neg
1	1	1				112	3 bytes	y	neg

The following bytes contains either the absolute x and y coordinates or the incremental distances.

The coordinates are specified as integer numbers with an accuracy of 0.001 cm. To save space and maintain the accuracy, one in ten values is given as absolute value and the following nine are specified incremental. However this ratio may be changed easily; see the description of the routine BUFFER.

* N=3 Defines the end of the Intermediate File. This function does not contain other arguments.

* N=4 Closes one buffer record. This function does not contain other arguments.

* N=6 Other pen colour is wanted. The K byte contains the number of the newpen.

* N=7 Initiates a new picture segment. The following K byte contains the number of the segment, $K < 256$. This feature allows for partial graphic output. One may instruct the device oriented programs to draw only specific pictures from the Intermediate File.

Any other functions can be added with other positive values of N, with the limitation that a complete set of values is of the format:

N,X,Y,K

in which

N and K are of integer*4 type and
X and Y are of real*4 type

Depending on the significance of the N value, some of the other arguments may be omitted.

Additional options, specified by N, need modifications of the routines BUFFER and READB.

The compressing of the data as explained before, is performed by the routine BUFFER.
Presently the Intermediate File is composed of records of 800 bytes. Each record is ended with one byte of value 4. The Intermediate File recognizes only one coordinate system; the (0,0) point is always the lower-left point of the drawing surface and cannot be replaced.
The units of lenghts are centimeters.
The maximum field for a single distance is 3 bytes, which corresponds to a maximum value of 16777 cm, that limits the size of the drawings in both directions.

Appendix B:

List of subroutines

The following list of subroutines is not exhaustive; the GINO manuals contain many other functions. However, the here mentioned subroutines are largely sufficient for the common use of the graphic facilities and it is not advisable to apply other routines than these because of probable errors.
The list specifies the manual for consulting:

GF - Manual GINO-F
GG - Manual GINOGRAF
GZ - Manual GINOZONE
GB - Green Book on Graphics, this volume

The last column specifies the source of the routines. Some routines are modified by the Computing Cenre, to allow for necessary standardizations or simplification of use.

CAD - Computer Aided Design, Cambridge
JRC - Joint Research Centre
NCAR - National Centre for Atmospheric Research, Colorado, U.S.A.
IMSL - International Mathematical and Statistical Library

ARCBY2	RELATIVE DRAW 2D ARC	GF	CAD
ARCBY3	RELATIVE DRAW 3D ARC	GF	CAD
ARCINC	ACCURACY OF ARCS	GF	CAD
ARCTO2	ABSOLUTE 2D ARCS	GF	CAD
ARCTO3	ABSOLUTE 3D ARCS	GF	CAD
ARCTOL	TOLERANCE OF ARCS	GF	CAD
AXIDRA	DRAW AN AXIS	GG	CAD
AXILAB	DRAW LABELS FOR AXIS	GG	CAD
AXIPOS	DEFINE POSITION OF AXIS	GG	CAD
AXISCA	DEFINE TYPE OF SCALING FOR AXIS	GG	CAD
AXON3	AXONOMETRIC VIEUWPOINT	GF	CAD
BARCHA	COMPLETE BARCHART	GG	CAD
BROKEN	SET LINE MODE	GF	CAD
BUFFER	OUTPUT TO INTERMEDIATE FILE	GB	JRC
CARPET	3D FUNCTIONS WITH HIDDEN LINE REMOVAL	GB	JRC+NCA
CHAA1	CHARATER OUTPUT FROM A1 ARRAY	GF	CAD
CHAANG	ORIENTATION OF STRING	GF	CAD
CHAARR	CHARACTER OUTPUT FROM ARRAY	GF	CAD
CHAASC	CHARACTER OUTPUT ASCII NUMBER	GF	CAD
CHACEN	ONE CHARACTER CENTERED OUTPUT	GF	CAD
CHAENQ	CHARACTER SPECIFICATION ENQUIRY	GF	CAD
CHAESC	ESCAPE CHARACTER DEFINITION	GF	CAD
CHAFIX	INTEGER NUMBERS	GF	CAD
CHAFLO	REAL NUMBERS	GF	CAD
CHAHOL	CHARACTER OUTPUT	GF,GB	CAD
CHAINT	OUTPUTS INTEGER AS DECIMAL	GF	CAD
CHASWI	CHARACTER TRANSFORMATION SWITCH	GF,GB	CAD
CHAZIS	DEFINE CHARACTER SIZE	GF,GB	CAD
CURBY2	SMOOTH CURVE 2D INCREMENTAL	GF	CAD
CURTO2	SMOOTH CURVE 2D ABSOLUTE	GF	CAD
DASENQ	LINE MODE ENQUIRY	GF	CAD
DASHED	SET LINE MODE	GF	CAD
DRAW2	DRAW MULTI POINT LINE 2D INCREMENTAL	GF	CAD
DRAW3	DRAW MULTI POINT LINE 3D INCREMENTAL	GF	CAD
ERRMAX	SET MAXIMUM NUMBER OF ERRORS	GF	CAD
FROM3	PERSPECTIVE PROJECTION VIEUWPOINT	GF	CAD
GEND	LAST GRAPHIC CALL OBLIGATORY	GB	JRC
GRABAR	DRAW BARS IN PRESET AXIS SYSTEM	GG	CAD
GRACUR	DRAW SMOOTH CURVE IN PRESET AXIS SYSTEM	GG	CAD
GRAF	DRAW COMPLETE GRAPH	GG	CAD

GRAHIS	DRAW HISTOGRAM IN PRESET AXIS SYSTEM	GG	CAD
GRALIN	STRAIGHT LINE ACCORDING TO GRAPHICAL AXIS	GG	CAD
GRAMOV	MOVE PEN ACCORDING TO GRAPHICAL AXIS	GG	CAD
GRAPOL	PLOT CURVE IN PRESET AXIS SYSTEM	GG	CAD
GRASPA	SPECIFY SPACE COORDINATES OF FUNCTION POINT	GG,GB	CAD
GRASYM	SYMBOLS ON CERTAIN POINTS OF FUNCTION	GG	CAD
GREYSI	GREY LEVEL SIMULATION FOR POLYGONS	GB	JRC
GRID	TYPE OF BACKGROUND FOR GRAPHS	GG	CAD
GSTART	FIRST GRAPHIC CALL OBLIGATORY	GB	JRC
HISCHA	COMPLETE HISTOGRAM	GG	CAD
ITALIC	DRAW FOLLOWING CHARATERS ITALIC	GF	CAD
LINBY2	STRAIGHT LINE DRAWING INCREMENTAL	GF,GB	CAD
LINBY3	STRAIGHT LINE 3D INCREMENTAL	GF	CAD
LINTO2	STRAIGHT LINE 2D ABSOLUTE	GF,GB	CAD
LINTO3	STRAIGHT LINE 3D ABSOLUTE	GF	CAD
MOVBY2	NON-DRAWING MOVE 2D INCREMENTAL	GF	CAD
MOVBY3	NON-DRAWING MOVE 3D INCREMENTAL	GF	CAD
MOVTO2	NON-DRAWING MOVE 2D ABSOLUTE	GF,GB	CAD
MOVTO3	NON-DRAWING MOVE 3D ABSOLUTE	GF	CAD
PENENQ	PEN COLOUR ENQUIRY	GF	CAD
PENSEL	PEN COLOUR DEFINITION	GF	CAD
PICTNO	SEGMENTING DRAWINGS IN SEQUENTIAL ORDER	GB	JRC
PIECHA	COMPLETE PIECHART	GG	CAD
PIEPAP	PIECHART FRAME	GG	CAD
PIESEG	INSERTS SEGMENT IN CURRENT PIECHART FRAME	GG	CAD
PIESET	RESTORES DEFAULT PIECHART FRAME	GG	CAD
POLBY2	MULTI POINT LINE 2D INCREMENTAL	GF	CAD
POLBY3	MULTI POINT LINE 3D INCREMENTAL	GF	CAD
POLTO2	MULTI POINT LINE 2D ABSOLUTE	GF	CAD
POLTO3	MULTI POINT LINE 3D ABSOLUTE	GF	CAD
POSPIC	PEN POSITION ENQUIRY PICTURE COORDINATES	GF	CAD
POSSPA	PEN POSITION ENQUIRY SPACE COORDINATES	GF	CAD
PROJ3	PERSPECTIVE VIEUW VIEUWPOINT	GF	CAD
READB	READ FROM INTERMEDIATE FILE	GB	JRC
ROTAT2	ROTATION 2D	GF,GB	CAD
ROTAT3	ROTATION 3D	GF	CAD
SCALE2	SCALE FACTORS FOR X AND Y	GF	CAD
SCALE3	SCALE FACTORS FOR X, Y AND Z	GF	CAD
SCALE	SCALE FACTOR FOR ALL DIMENSIONS	GF	CAD
SHEAR2	SHEARING FOR EITHER X OR Y	GF	CAD

SHEAR3	SHEARING IN 3D	GF	CAD
SHIFT2	MOVE SPACE AXIS 2D INCREMENTAL	GF,GB	CAD
SHIFT3	MOVE SPACE AXIS 3D INCPEMENTAL	GF	CAD
SPAGRA	FUNCTION VALUE OF A POINT TO CURRENT AXIS SYSTEM	GG	CAD
SYMBOL	DRAW ONE CENTERED SYMBOL	GF	CAD
SYMBY2	DRAW A SERIES OF CENTERED SYMBOLS 2D INCPEMENTAL	GF	CAD
SYMBY3	DRAW A SERIES OF CENTERED SYMBOLS 3D INCREMENTAL	GF	CAD
SYMTO2	DRAW A SERIES OF CENTERED SYMBOLS 2D ABSOLUTE	GF	CAD
SYMTO3	DRAW A SERIES OF CENTERED SYMBOLS 3D ABSOLUTE	GF	CAD
TRABEG	STORES COPY OF CURRENT TRANSFORMATIONS	GF	CAD
TRAEND	RESTORES LAST TRANSFORMATIONS	GF	CAD
TRANSF	TRANSFORMATION SWITCH	GF	CAD
WINDO2	SET 2D WINDOW	GF	CAD
WINDO3	SET 3D WINDOW	GF	CAD
WINDOW	WINDOW SWITCH	GF	CAD
XGRAF4	COMPLETE GRAPHS A4 FORMAT	GB	JRC
XGRAF5	COMPLETE GRAPHS A5 FORMAT	GB	JRC
XGRAF	ADD CURVE TO LAST AXIS SYSTEM	GB	JRC
XHIST4	COMPLETE HISTOGRAM A4 FORMAT	GB	JRC
XHIST5	COMPLETE HISTOGRAM A5 FORMAT	GB	JRC
XPRPLT	GRAPHS BY LINEPRINTER	GB	JRC+IMS

APPENDIX C
****

LIST OF GINO ERRORS
****

1 CALL TO GINO BEFORE DEVICE NOMINATED
2 PICBEG BEFORE DEVICE NOMINATION
3 PICEND OUTSIDE PICTURE SEGMENT
4 TOO MANY ERRORS,PROGRAM HALTED
5
6 NO DEVICE NOMINATED,PROGRAM HALTED
7 NO CALL TO DEVEND IN PROGRAM
8 DEVICE NOT AVAILABLE
9 QUALIFYING ROUTINE CALLED AFTER DEVICE INITIALIZATION
10 CURSOR NOT AVAILABLE
11 CURSOR MISREAD
12 CHARACTER STRING ILLEGALLY TERMINATED IN CALL TO CURDEF
13 ILLEGAL CHARACTER AFTER ESCAPE CHARACTER IN CURDEF
14 UNDEFINED CURSOR FUNCTION
15 CALL TO CURSOR BEFORE CALL TO CURDEF
16 CROSS NOT AVAILABLE
17 INPUT RECORD OF INCORRECT LENGTH ENCOUNTERED IN BACK-END
18
19
20 PAPER REQUESTED IS TOO LARGE
21 ATTEMPT TO DRAW OVER X NEGATIVE MARGIN
22 ATTEMPT TO DRAW OVER X POSITIVE MARGIN
23 ATTEMPT TO DRAW OVER Y NEGATIVE MARGIN
24 ATTEMPT TO DRAW OVER Y POSITIVE MARGIN
25 ATTEMPT TO DRAW OVER Z NEGATIVE MARGIN
26 ATTEMPT TO DRAW OVER Z POSITIVE MARGIN
27 ATTEMPT TO POSITION ANCHOR OR COPY SEGMENT OUTSIDE DEVICE LIMITS
28 CALL TO PICBY WITH SEGMENT WITH UNDEFINED ANCHOR
29
30 FIELD WIDTH GREATER THAN 32 IN CHAINT, CHAFLO OR CHAFIX
31 NUMBER TOO LARGE FOR FIELD IN CHAINT
32 CHARACTER STRING INCORRECTLY TERMINATED
33 ILLEGAL CHARACTER AFTER ESCAPE CHARACTER
34 MORE THAN 6 DECIMAL PLACES IN CHAFIX
35 CHARACTER NOT AVAILABLE ON OUTPUT DEVICE
36 ATTEMPT TO POSITION OUTSIDE DEVICE LIMITS USING CHAPOS
37 ATTEMPT TO DEFINE ILLEGAL ESCAPE CHARACTER
38
39
40 CIRCLE NOT DEFINED IN ARC ROUTINES
41
42 SPACEPOS OUT OF DATE IN ARC OR IRC ROUTINES
43 NO NON-TRIVIAL INCREMENTS GIVEN IN CURVE ROUTINES

44 ONE NON-TRIVIAL INCREMENT GIVEN WITH INCONSISTENT END
CONDITION IN CURVE ROUTINES
45
46
47
48
49
50 DASH LENGTH PLUS DOT LENGTH GREATER THAN REPEAT LENGTH IN DASHED
51 ILLEGAL LINE TYPE IN CALL TO DASHED
52
53 TOO MANY PENS REQUESTED
54 ILLEGAL OPERATION ON PEN
55 UNAVAILABLE PEN REQUESTED
56
57
58
59
60 CALL TO TRAEND BEFORE CORRISPONDING CALL TO TRABEG
61 TRABEG'S NESTED MORE THAN TEN DEEP
62 AXIS OUT OF RANGE IN ROTAT3
63 SPACEPOS OUT OF DATE IN POSSPA
64
65 CURRENT TRANSFORMATION CONTAIN PERSPECTIVE BUT PROGRAM
DOES NOT CALL PERSPECTIVE ROUTINE
66 ILLEGAL ARGUMENTS TO VIEWSE
67
68
69
70
70 ILLEGAL CHARACTER FOLLOWS ESCAPE CHARACTER IN KEYDEF OR KEYARR
71 END OF PROCESSING FOLLOWS ',' IN SUB-STRING OF KEY DEFINITIONS
72 NUMERIC OR PRINT CHARACTER FOUND WHERE ',' EXPECTED IN KEY LIST
73 NUMERIC OR PRINT CHARACTER FOUND WHERE ',' EXPECTED IN
IMPLEMENT LIST
74 INVALID IMPLEMENT SPECIFIED IN KEY DEFINITIONS
75 '.' FOUND IN IMPLEMENT LIST
76 ',' FOUND WHERE NUMERIC EXPECTED IN IMPLEMENT LIST
77 PRINT CHARACTER FOUND WHERE NUMERIC EXPECTED IN IMPLEMENT LIST
78 '/' FOLLOWS ',' IN IMPLEMENT LIST
79
80 SEGMENT NUMBER OUT OF RANGE
81 CALL TO MODEND BEFORE CORRISPONDING CALL TO MODBEG
82 DATA TYPE OUT OF RANGE IN EVESET OR EVEDEL
83
84
85
86 KEY DEFINITION STRING INCORRECTLY TERMINATED
87 MAXIMUM ALLOWED NUMBER OF KEY DEFINITIONS EXCEEDED
88 INVALID CHARACTER FOUND IN KEY DEFINITION STRING - IGNORE
89 ILLEGAL UNPACKING FORMAT REQUESTED IN KEYARR

90 OUTPUT LIMIT EXCEEDED
91 END OF FILE OR TAPE REACHED
92
93
94
95
96
97
98
99
100 SYSTEM ERROR - PLEASE CONSULT PROGRAMMING ADVISOR
101 SYSTEM ERROR - PLEASE CONSULT PROGRAMMING ADVISOR
102
103
104
105
106
107
108
109
110 FILE IN WRONG CODE FOR PICTUR
111 PICTURE NOT FOUND IN PICTUR
112 ILLEGAL CODE IN PICTUR
113 CALL TO DASHED OR BROKEN REQUIRED IN PROGRAM
114 CALL TO ARC ROUTINE REQUIRED IN PROGRAM
115 CALL TO SYMBOL REQUIRED IN PROGRAM
116 CALL TO DOT REQUIRED IN PROGRAM
117
118 CALL TO DASHED REQUIRED IN PROGRAM
119
120 ILLEGAL ARGUMENT TO WINDOW
121
122
122
123
124
125

LIST OF GINOGRAF ERRORS
****

```
1 NUMBER OF COLUMNS NEGATIVE OR ZERO
2 VALUE FOR LOG AXIS NEGATIVE OR ZERO
3 HISTOGRAM TYPE AXIS NOT SPECIFIED - DEFAULT ASSUMED
4 ERROR IN LOG DATA
5
6
7
8
9
10
11
11
12
13 X-VALUE NEGATIVE OR ZERO FOR LOG AXIS
14 Y-VALUE NEGATIVE OR ZERO FOR LOG AXIS
15 NUMBER OF POINTS TO BE JOINED NEGATIVE OR ZERO
16 NUMBER OOF SYMBO;LS TO BE PLOTTED NEGATIVE OR ZERO
17 AXIS LENGTH NEGATIVE OR ZERO
18 AXIS SCALLINGING TYPE OUT OF RANGE - DEFAUTLT ASSUMED
19 NUMBER OF INTERVALS NEGATIVE OR ZERO
20 RANGE OF VALUES ZERO
```

To: Computing Support Library

Ms. A. Cambon
Building 36
Div. 1 Dep. A
JRC Euratom
21020 Ispra
Italy

I want to be registered as a user of the graphics facilities of the JRC, Ispra Establishment, and will consequently receive in the future the updates to this manual.

Name:.................................

Address:.............................

....

....

Signature:..........................