AD-A058 830 DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CE--ETC F/G 9/2
THE INTERACTIVE IMAGE DISPLAY SYSTEM.(U)
JUL 78 M HASS, M B MARQUARDT

UNCLASSIFIED DTNSRDC/CMLD-78/07 NL

1 OF 1
AD A058 830

END
DATE FILMED
-11-78
DDC

LEVEL II

12

DAVID W. TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER

Bethesda, Md. 20084

6 THE INTERACTIVE IMAGE DISPLAY SYSTEM,

by

10 Mel Haas and Mary Beth Marquardt

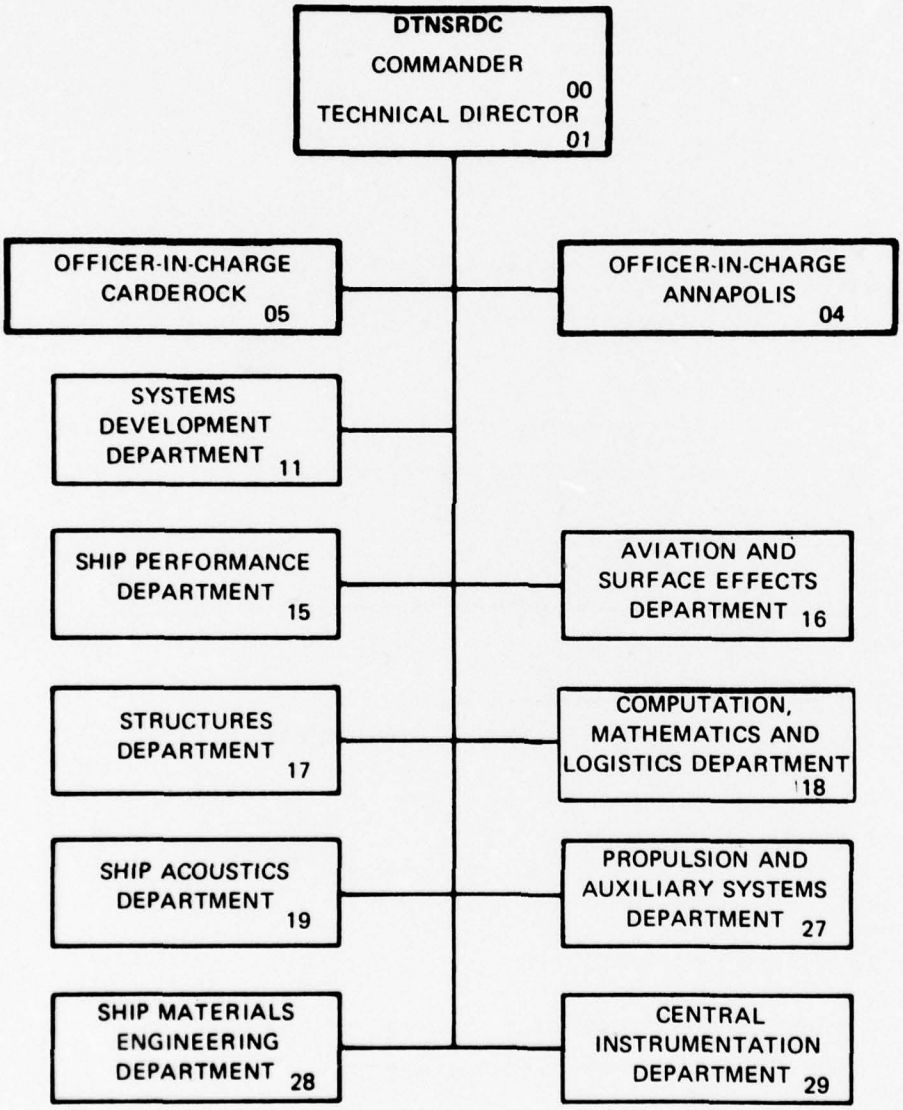
9 Final rept. Sep 74-Jan 76

16 R01403 17 SR01403Q1

COMPUTATION, MATHEMATICS, AND LOGISTICS DEPARTMENT
DEPARTMENTAL REPORT

11 JUL 78 12 40 p. 14 DTNSRDC/CMLD-78/07

MAJOR DTNSRDC ORGANIZATIONAL COMPONENTS



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER DTNSRDC/CMLD-78/07	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) THE INTERACTIVE IMAGE DISPLAY SYSTEM		5. TYPE OF REPORT & PERIOD COVERED Final Sept 1974 - Jan 1976 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Mel Haas and Mary Beth Marquardt		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS David W. Taylor Naval Ship Research and Development Center Bethesda, Maryland 20084		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 61153N, 15321, SR0140301 Work Units: 1-1843-347 1-1808-010
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE JULY 1978 13. NUMBER OF PAGES 41
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Interactive Graphics
Two-Dimensional Cross Plots
Three-Dimensional Perspective Projections

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

IMAGE is a high-performance interactive computer graphics program for displaying 2-D graphs and perspective projections of 3-D data. The development of IMAGE was prompted by the need for a tool for solving numerical fluid dynamics problems, but it was designed to be general enough for use in graphically representing input and output data in any problem area. The IMAGE input data to be examined may be produced by any FORTRAN program with simple WRITE statements. The data can then be displayed, studied, manipulated, and

(Continued on reverse side)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

(Block 20 continued)

modified by the user. Publication quality hardcopy output of selected views is easily and quickly obtained from the CALCOMP or SC 4060 plotters.

IMAGE is designed to run quickly and efficiently. Speed is attained by minimizing the number of requests required to perform a function. Efficiency results from programming each transaction as a separate module, called a task. These tasks are loaded into core memory only as needed. Virtual memory techniques also save core storage.

The program is available for general use on the Tektronix 4014 and the CDC IGS 274 terminals linked to the DTNSRDC CDC 6700 and 6400 computers.

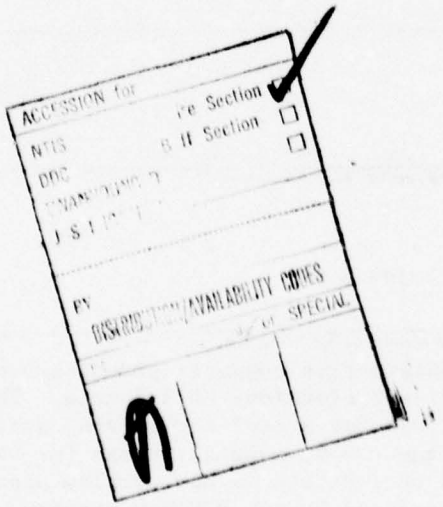


TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

IMAGE is a high-performance interactive computer graphics program for displaying 2-D graphs and perspective projections of 3-D data. The development of IMAGE was prompted by the need for a tool for solving numerical fluid dynamics problems, but it was designed to be general enough for use in graphically representing input and output data in any problem area. The IMAGE input data to be examined may be produced by any FORTRAN program with simple WRITE statements. The data can then be displayed, studied, manipulated, and modified by the user. Publication quality hardcopy output of selected views is easily and quickly obtained from the CALCOMP or SC 4060 plotters.

IMAGE is designed to run quickly and efficiently. Speed is attained by minimizing the number of requests required to perform a function. Efficiency results from programming each transaction as a separate module, called a task. These tasks are loaded into core memory only as needed. Virtual memory techniques also save core storage.

The program is available for general use on the Tektronix 4014 and the CDC IGS 274 terminals linked to the DTNSRDC CDC 6700 and 6400 computers.

INTRODUCTION

IMAGE is an interactive computer graphics program that displays 2-D graphs of user supplied data arrays and projections of user supplied 3-D data. The IMAGE 2-D graphing capability allows a user to very quickly see the relationships among data that can be presented as cross plots of FORTRAN arrays. The user inserts WRITE statements in his program to produce a file for comparison plots, to display from one to nine graphs on a frame at a time, to scale the horizontal and vertical axes, and to plot any array from any file on any graph with marking symbols as desired. (See Figure 1, Typical 2-D Array Cross Plots.)

IMAGE provides perspective projections of 3-D geometric data defined as a collection of points. These points as displayed by IMAGE may be single isolated points in space, points along lines (such as trajectories, streamlines, equi-potential lines, isobars, etc.), and/or points that define a lattice (mesh) of connected line segments to define a surface or structure. User controls are provided to change the viewing parameters

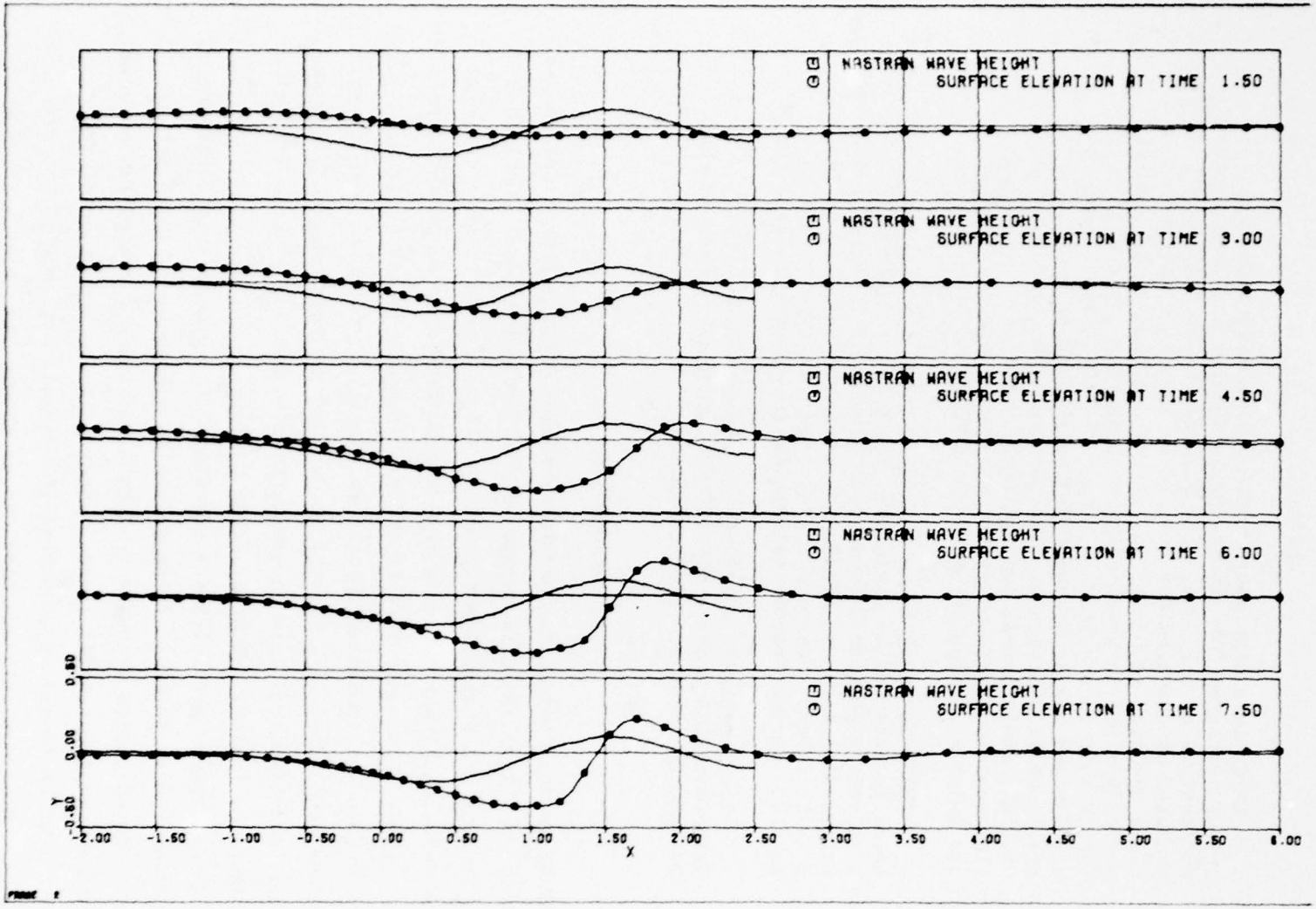


Figure 1 - Typical 2-D Array Cross Plots

and display options as required for detailed examination of the data. (See Figure 2, Typical 3-D Geometry Data Views.)

Publication quality hardcopy of any selected views produced by IMAGE may be easily obtained by postprocessing the IMAGE plotting file on any of the available plotters. Currently at DTNSRDC these plotters include the CALCOMP 936 and 763, the SC 4020, and the SC 4060. Figures 1 and 2 are examples of IMAGE generated hardcopy produced on the CALCOMP 936. This capability is in addition to the Tektronix 4014 terminal instant hardcopy capability.

IMAGE is available on the DTNSRDC CDC 6700 computer system from either the CDC 274 or Tektronix 4014 graphics terminal. After the selected terminal has been accessed, one control card statement begins execution of IMAGE (see Appendix B).

IMAGE is a transaction based, interactive system. The CDC 6700 computer system has a 5- to 15-second basic cycle time for processing each request for service from a terminal, and this cycle time is largely independent of the amount of service requested. This characteristic is exploited in IMAGE by minimizing the number of service requests needed to perform each function. Each transaction in IMAGE involves collecting the parameters of a service request from the user at the terminal and then invoking the CDC 6700 to process the request. This rather simple approach gives IMAGE a 5 to 10 times speed advantage over similar interactive programs on the CDC 6700. For example, a standard graphics program might begin with an initial menu:

READ PLOT DONE .

The user chooses the mode he wishes to enter. If he chooses "READ", the program might respond:

"TYPE IN FILE NAME".

After successfully reading the file, the program might again respond:

"FILE SUCCESSFULLY READ".

This process requires five service requests to perform one function, that of reading a data file.

In IMAGE, the user simply types in:

"RDXYZ, local file name".

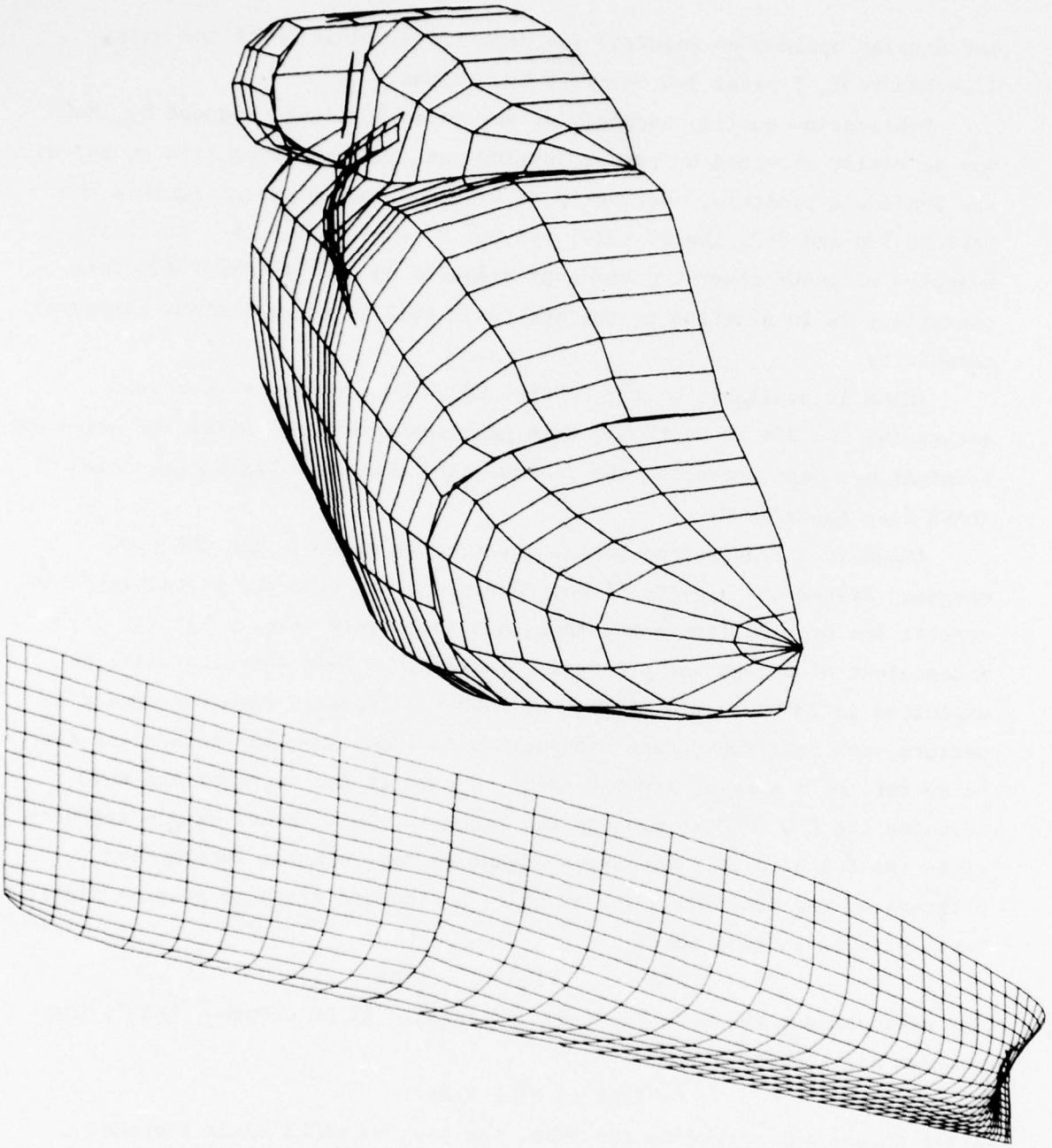


Figure 2 - Typical 3-D Geometry Data Views. Surface panelling of portions of a helicopter fuselage (top) and of a ship hull (bottom).

The program reads the file and responds only if an error occurs. Thus the same READ function is performed with one service request.

Use of a transaction basis for IMAGE minimizes the costs of operation of the program and greatly facilitates future additions and changes in capabilities. The program modules, called tasks, needed to process a transaction are loaded into core memory one-at-a-time as needed. The core memory of the CDC 6700 is not wasted on program code that is not actually being used, and the computer charges to the user reflect this. Each task is designed and coded to process a specified transaction, and is, to the maximum extent possible, independent of the other tasks in the program. It is very easy to define new transactions and to program the required additional tasks. Adding a new feature or capability rarely involves any change to existing code in the IMAGE program.

Virtual memory techniques are used in IMAGE to permit use of very large amounts of data.

PROGRAM ORGANIZATION

The IMAGE graphics system is organized as a collection of program modules (tasks) that interface with a main control program and with data storage areas (see Figure 3).

IMAGE can be divided into four basic groups of tasks: 3-D Geometric Data Points, 2-D Cross Plotting, CALCOMP 3-D, and Utility Subsystems. Each task corresponds to a command word on the menu (see Appendix A; command words are listed alphabetically, not according to sub-system). Each task provides an independent interactive graphics capability and is discussed as an independent entity in this report. Each task or command is available for execution at all times and any logical combination of 2-D, 3-D, or utility operations may be performed.

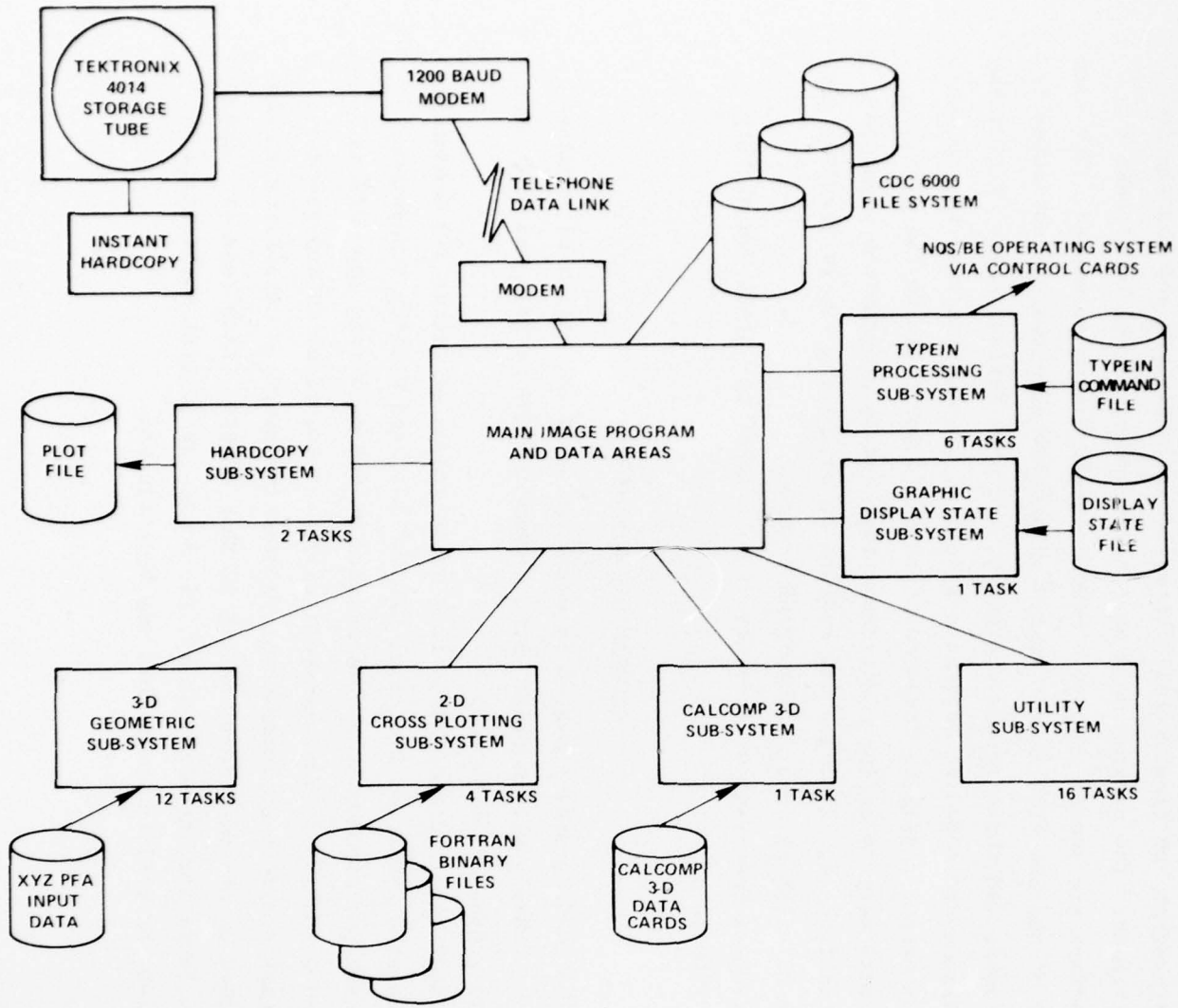


Figure 3 - IMAGE Program Organization Diagram

TYPE-IN PROCESSING

INPUT PROCEDURE

The primary input device for IMAGE is the typewriter keyboard of the Tektronix 4014 or the IGS 274 terminal. The basic technique employed is to type something to have something done. Commands as defined within IMAGE and control cards may be executed by typing in.

On the Tektronix terminal, characters typed on the keyboard appear on the left side of the screen. A cursor (small blinking square) indicates the position of the next symbol. The contents of a type-in line are processed when the RETURN key is pressed. The line may be nullified and restarted by pressing the CTRL and X keys simultaneously, the RETURN, and then retyping the line. An individual character may be replaced by pressing CTRL and H simultaneously, then typing the new character (over the old). Text lines print successively on the screen until the screen is erased in connection with some command (e.g., SETV, DISPLAY, PLOTPD) or by pressing the RESET PAGE key on the terminal.

On the CDC 274 terminal a dashed type-in line appears in the lower left corner of the display screen. A line may be erased and restarted by pressing the RUBOUT key, an individual character by pressing the ← key.

Immediately above the type-in line is a scroll display of previous text lines. New lines enter the bottom of the scroll as old ones are discarded from the top. The number of lines of text which appear is controlled by the LINES command. For example, LINES,10 sets the scroll display to ten lines. LINES or LINES, (with or without comma) sets the scroll display to the default value of five lines.

The LINES command has no effect on the Tektronix terminal text display.

COMMAND PROCESSING

Requests for service from IMAGE are made by typing a command, which is a command word followed by any parameters required or needed for that command. By typing "?" and then pressing the RETURN key, the user obtains a list of the commands in alphabetical order (see Appendix A) on the screen. Typing a particular command word followed by "?" and RETURN will

provide a detailed explanation of the command, any parameters required, and default values.

Any command is available at any time. However the user must recognize which commands apply to the 2-D Array Cross Plotting and which apply to Viewing 3-D Graphic Data Points (see Table 1). For example, the SCALEPD command will not scale 3-D data; the ZOOM command will not enlarge a 2-D Cross Plot.

The parameters for a command may be integer numbers, real numbers, or alphanumeric text as required for the particular command. Individual parameters are separated by one or more blanks or a comma, semi-colon, left parenthesis, or right parenthis. Alphanumeric text parameters may include these delimiters if the parameter is the last one expected. Parameters bracketed by $ symbols are passed to the program exactly as typed.

Integer numbers may be input with a decimal point (i.e., 3.0) as long as the fractional part is zero. Real numbers may be input in any valid format, with or without decimal point, with or without "E" for power of ten exponent.

Every command parameter has a default value, which is the value used by IMAGE if no parameter is typed in. The default value is set at the value normally expected for the parameter or is flagged to indicate to the command processing task that no change from some previously set value is wanted for the parameter. Typing a double comma ",," or truncating the type-in line at the last desired input value will indicate use of a default value.

Command type-in errors return an error message to the screen. Some typical error messages and their causes follow.

"NOT LEGAL COMMAND" The first word of the type-in line is not in the list of IMAGE command words.

"COMMAND NOT UNIQUE" Not enough letters of the command word have been typed to uniquely identify the command wanted. It is not necessary to type the whole command word each time; only enough letters are required to identify the command in the list.

TABLE 1 - LIST OF COMMAND WORDS
ACCORDING TO FUNCTION

Utility Commands
ABORT
CPTIME
CREATEF
DATE
DISPDF
DMDMP
DONE
EXIT
FILES
LINES
PRUDUMP
Q8SUB
RDCDS
RPV
TIME
USEF

2-D Commands	3-D Commands	Calcomp 3-D
GRIDPD	CIDOPT	THRED
PLOTPD	CNGPT	
RDPD	CVG	
SCALEPD	DISPLAY	
	ETOS	
	FIND	
	RDXYZ	
	SAVXYZ	
	SCALE	
	SETV	
	VGDISP	
	ZOOM	

"ILLEGAL CHARACTER IN NUMBER" An integer or real number parameter was mis-typed.

"INTEGER HAS FRACTIONAL PART" Either an integer parameter or a scientific notation exponent has been typed with a decimal point and fraction.

Occasionally a task will request that the user supply a line of alphanumeric text input. The input will not be interpreted as a command when so requested. To execute a command before supplying the requested type-in, a "$" typed before the command word will flag command execution.

The command RDCDS allows the user to drive the IMAGE program from cards. Card images are read from a file and processed as type-in commands. The command is: "RDCDS,lfn." If ,lfn is not typed, the program expects to read from INPUT.

This RDCDS capability could be useful if, for example, a user wanted to obtain hardcopy of some plots produced at the screen in a previous session. All the commands to produce the plots could be written to a file. IMAGE would then generate the plots without further user involvement.

CONTROL CARD EXECUTION

The type-in processor of IMAGE may be used to execute any valid CDC 6700 NOS/BE system control card by typing "#" preceding the control card text and pressing the RETURN key. IMAGE will execute the control card and then display a message on the screen. For example,

```
#ATTACH,FPDUMP,ID=CSYS
```

attaches the system file FPDUMP and produces a message on the screen,

```
#ATTACH,FPDUMP,ID=CSYS
PF CYCLE NO.=001.
```

If execution is unsuccessful, IMAGE produces a message to that effect. Control card errors will be reported in the dayfile by the NOS/BE system. However, IMAGE does not check either the validity of control cards or control card execution errors.

A control card procedure cataloged on a permanent file named PROCFIL in the format of the CDC 6700 NOS/BE Cyber Control Language (CCL) may also be executed in IMAGE. The user types

&, the user ID under which the file is cataloged, the name of the procedure, and the parameters of the procedure, and presses RETURN. For example,

&CAMV,PLT936,CAMV,,,CA0833

executes a procedure to produce a magnetic tape (CA0833) for the CALCOMP 936 plotter. Exactly the same results are obtained by typing

#BEGIN,MYPRO,,CAMV,PLT936,CAMV,,,CA0833.

The messages produced by & procedure execution are reported on the screen in the same way as are those from a single control card execution with #.

UTILITY COMMANDS

Some of the commands enable the user to perform "utility" operations, such as obtaining job status, file manipulation, user controlled job termination, and user requested dumps. Some of the commands were intended for program maintenance (e.g., RPV, Q8SUB, DMDMP, PRUDMP) and should be used only by those who fully understand what these commands accomplish.

JOB STATUS COMMANDS

CPTIME

DATE

DISPDF

TIME

The three values of date, time and central processing time are all displayed by any of the three commands DATE, TIME, or CPTIME depending upon the user's interest. Each of the commands will procude a temporary display of the current date, current time of day and accumulated CP time. The CP time is in accounting seconds and is only approximate.

The command DISPDF, when used on the CDC 274 terminal, will display the dayfile (as many lines as have been allowed by LINES). DISPDF is listed in the command words, but is not operable on the Tektronix.

FILE MANIPULATION COMMANDS

CREATEF

FILES

Q8SUB

USEF

CREATEF

CREATEF catalogs a local file as a permanent file. The parameters are:

a) LFN - local file name to be cataloged

b) PFN - permanent file name which must begin with the four-letter user ID

c) AC - account number (defaults to job account number)

d) NC - number of cycles to retain (default is four cycles)

If there is no previously cataloged file with the permanent file name, a new file is cataloged. Each LFN cataloged under an already established PFN will be cataloged with the same name and passwords, but with the next number in sequence (called the cycle number) to distinguish it from other existing cycles of the same PFN. The ID is the first four letters of the PFN. Lower cycles are purged to maintain NC number of cycles (between 2 and 4) of the PFN. LFN must be a *PF requested file (see USEF).

FILES

FILES will display a list of all files attached to the job. File names marked with * denote permanent files. There are no parameters.

Q8SUB

Q8SUB is a command to substitute a new file name for a FORTRAN program file. The parameters are:

a) FORTRAN program file name

b) new file name

USEF

USEF is a command to create a user specified permanent file. The parameters are:

a) LFN - local file name

b) PFN - permanent file name

c) ID - user ID under which to catalog (may be omitted if the first four characters of PFN are the user ID)

USEF will unload the LFN and then attach the permanent file named by the PFN. If the LFN is preceded by minus (-), the permanent file will be directly attached as the local file name. Otherwise, the permanent file will be copied to a virgin *PF local file. If PFN is 0 (zero), only a virgin *PF local file will be created. For example,

USEF,LFN,0

must be executed before CREATEF can be called to catalog a LFN the first time.

PROGRAM RECOVERY

RPV

RPV sets up the recovery from a program error and allows the user to continue normally.

PROGRAM TERMINATION

ABORT

DONE

EXIT

The user must remember to catalog any files he has created and wishes to save before terminating.

ABORT

ABORT ends the job immediately with no exit processing, such as clearing the display buffer. The user could also abort the job by pressing the % key and A simultaneously. ABORT should be called only when a program error prevents normal termination.

DONE, EXIT

DONE and EXIT are synonyms for normal job termination. Either command may be typed when the user is ready to end a session.

OBTAINING DUMPS

DMDMP

PRUDMP

DMDMP

DMDMP is the command to print octal and display code dump of a data

manager file.

"DMDMP,N" dumps all of data manager logical file N.

PRUDMP

PRUDMP is the command to print octal and display code dump by physical record unit (PRU). There are three possible parameters:

a) LFN (default is TAPE4)

b) starting PRU

c) number of PRU's to dump

"PRUDMP" dumps all of file named TAPE4.

"PRUDMP,LFN" dumps all of file LFN.

"PRUDMP,LFN,N" dumps the content of the file LFN starting at PRU N.

"PRUDMP,LFN,N,M" dumps M PRU's of file LFN starting at PRU N.

2-D ARRAY CROSS PLOTTING

The 2-D Array Cross Plotting capability is provided for convenient, quick examination of data produced by programs written in FORTRAN. In general, the FORTRAN program is run, producing a file of X and Y arrays; the IMAGE program is then used to cross plot selected arrays under control of the user.

IMAGE provides from one to nine individual grids (graphs) in one view, and plots any number of curves of up to 1000 XY points each on any grid or grids, from any files. Automatic scaling is provided to include the range of data on a particular file, and operator control is provided to select portions of the data to be viewed.

DATA FILE INPUT

IMAGE reads a FORTRAN data file in two logical sections.

Section 1. Contains a single X-array and any number of Y-arrays of the same length.

Section 2. Contains any number of X,Y array pairs, where different pairs of arrays may have different lengths.

The format of the file will be described in terms of the FORTRAN statements used to write the file in the user's program. The file must contain the data in exactly the order and formats described, and must not

have any other data written on it.

The file is standard FORTRAN (CDC 6700 NOS/BE FTN Compiler) binary file. It must be assigned by logical unit number in the PROGRAM card of the user's program. In this description, logical unit 10 is used, but any other logical unit would work. For example:

```
PROGRAM ABC(INPUT,OUTPUT,TAPE5=INPUT,TAPE10)
```

assigns logical unit 5 for input data cards and logical unit 10 for writing 2-D arrays for IMAGE.

The arrays are written on the file with FORTRAN unformatted WRITE statements. There is virtually unlimited flexibility in the manner in which the arrays may be dimensioned and indexed in the WRITE statements. For this example the arrays are assumed to be singly dimensioned, separate arrays.

```
DIMENSION X(250) , Y(250),TITLE(4)
```

The X-array contains the point data values to be displayed in the horizontal direction, the Y-array contains the point data values to be displayed in the vertical direction. The TITLE-array contains up to 40 characters of title for each X,Y array pair on Section 2 of the file. IMAGE will handle arrays of up to 1000 point values. Section 1 of the data file is produced by WRITE statements as follows:

```
WRITE(10)NY,NPTS,(X(I),I=1,NPTS)
```

and NY executions of

```
WRITE(10)Z,(Y(I),I=NPTS)
```

NY is an integer variable set to the number of Y-arrays that may be plotted against the initial X-array. NPTS is the length (number of points) of the arrays. Z is a real number that uniquely identifies the particular Y-array. There is no limit to the value of NY. NPTS must be less than 1001.

If no Section 1 data are to be given, the FORTRAN program must still include

```
WRITE(10)NY,NPTS,(X(I),I=1,NPTS)
```

with NY=0 and NPTS=1 or more, and a corresponding number of X values.

Section 2 data are produced by the following statements:

```
WRITE(10)NS
```

and NS executions of

```
WRITE(10)(TITLE(I),I=1,4)
WRITE(10)NPTS,(X(I),Y(I),I=1,NPTS).
```

NS is an integer variable set to the number of X,Y sets that may be plotted. NPTS is the length of the arrays, and may be different for different array sets. TITLE is a 40-character identification of the particular set of X,Y array pairs. The FORTRAN program may include ENCODE statements to set the content of the TITLE array. There is no limit to the value of NS. NPTS must be less than 1001.

If no Section 2 data are to be given, no Section 2 WRITE statements need be executed.

The data file is most easily transferred from the user's FORTRAN program job to the IMAGE graphics terminal session by use of the CDC 6700 NOS/BE permanent file system. In the user's control cards, a "REQUEST, TAPE10,*PF." control card is placed before the FORTRAN program execution card, and a "CATALOG,TAPE10,ID=XXXX." control card placed after it (XXXX denote the user's accounting initials). Then, at the IMAGE graphics terminal the user types:

"ATTACH,TAPE10,ID=XXXX"

to retrieve the data file with local file name (LFN) of "TAPE10".

The 2-D Array Cross Plotting data file is read by IMAGE when the RDPD (read plot data) command is given. The user types:

"RDPD,TAPE10"

to have IMAGE read the data on the local file named "TAPE10". If there are multiple data files, RDPD is used to read each one in turn. The files are given identification numbers in the order in which they are read by RDPD, i.e., the first execution of RDPD gives file 1, the second file 2, etc. There is no limit to the number of files that may be read at one session.

GRID OPTIONS

IMAGE can display from one to nine grids (individual graphs) on the display screen. On the Tektronix each grid appears to be about 10.5" wide with X-axis labels at 5/8" intervals, and height as follows:

Number of Grids	Individual Grid Height (approximate)
1	6 1/4"
2	3 1/8"
3	2"
4	1 3/8"
5	1 3/8"
6	5/8"
7	5/8"
8	5/8"
9	5/8"

Y-axis labels are given at 5/8" intervals on the bottom grid only.

The number of grids and the spacing of the grid lines is specified by command GRIDPD. The default parameters are a single grid with grid lines every 5/8". For example, if the user wants two grids with twice as many grid lines, he types:

"GRIDPD,2,0.5."

Grid lines will be drawn halfway between the default grid lines. If no grid lines are desired, any negative grid spacing may be given. Typing the GRIDPD command erases any previous display.

SCALING

IMAGE sets the range of the X- and Y-axes according to the parameters of the SCALEPD command. The axis values are shown on the bottom grid of the display. Only that portion of each curve that is within the grid is shown.

The parameters of SCALEPD command are:

a) the data value of the leftmost axis marker on the X-axis

b) the data value increment (delta) for each mark on the X-axis

c) the data value of the bottom axis marker on the Y-axis

d) the data value increment (delta) for each mark on the Y-axis

The axis range set by the SCALEPD command applies to all the graphs produced by the succeeding PLOTPD commands. SCALEPD causes any existing display to be erased and a new grid to be displayed.

CROSS PLOT SELECTION

Each individual curve (cross plot of an X,Y array pair) is produced by the execution of a PLOTPD command. The parameters of the PLOTPD command specify:

a) the particular arrays to cross plot
b) the grid on which to plot
c) the section of the file from which to obtain the arrays
d) the file from which to obtain the arrays
e) the plotting symbol spacing
f) the particular plotting symbol to use

Each curve plotted from Section 1 of a data file is identified by a Z-value given with the Y-array to be plotted. The curve is specified in the PLOTPD command by input of the Z-value as the first parameter.

Each curve plotted from Section 2 of a data file is identified by the order in which the array pairs were written on the file. The curve is specified in the PLOTPD command by input of the ordering count (1,2,3,... etc.) as the first parameter.

The grids are numbered 1,2,3,...etc. from the bottom of the screen. The second parameter of the PLOTPD command is the grid number. If this parameter is specified higher than the number of grids displayed, the curve will appear on the highest (top) grid. If this parameter is omitted, the curve will appear on the next highest grid.

The third parameter in the PLOTPD command is the file section (1 or 2). If this parameter is omitted, the section used in the previous PLOTPD command will be used.

The data files are identified by the order in which they are read by execution of the RDPD command. The first file read by RDPD is file 1, the second is file 2, etc. The fourth parameter in the PLOTPD command is the file number (1,2,3,...etc.) of the file from which to obtain the arrays to be plotted. If this parameter is omitted, the same file used in the previous PLOTPD command will be used.

The fifth parameter of the PLOTPD command controls the appearance of the curve displayed. If this parameter is zero (0) or omitted, straight line segments will be drawn between data points and no symbols will be displayed. If the first parameter is a positive number, +n, straight line

segments will be drawn between data points but a symbol will be shown only at every n^{th} point. If the fifth parameter is a negative number, -n, a symbol will be shown only at every n^{th} point and the symbols will not be connected.

The sixth parameter is a code number for the symbol shown on the curve. Table 2 shows the first 13 symbols and their code values. If this parameter is omitted, the next symbol in sequence will be used.

TABLE 2 - PLOTS SYMBOLS AND CODES

Symbol	Code
	0
	1
	2
	3
	4
	5
	6
	7
	8
	9
	10
	11
	12
	13

Therefore, the PLOTPD command will consist of some combination of these parameters:

PLOTPD, Array i.d. no., grid no., file section no., file no., plotting symbol spacing, plotting symbol

Several typical examples of PLOTPD commands and explanations of what will appear on the IMAGE display are given here.

"PLOTPD,1.394.2,1,3"

The Y-array with Z-value of 1.394 will be plotted against the X-array on the 2nd from the bottom grid. The arrays are from Section 1 of the file specified in the 3rd RDPD command given. Straight line segments will connect the data points and no symbols will be shown.

"PLOTPD,1.48,1"

The Y-array with a Z-value of 1.48 will be plotted on the bottom grid. The arrays are from the same section and file as in the previous PLOTPD command. Straight line segments will connect the data points and no symbols will be shown.

"PLOTPD,8,1,2,4,-1,4"

The 8th X,Y array pair will be cross plotted on the bottom grid. The arrays are from Section 2 of the file read by the 4th RDPD command. Every point will be shown with symbol code=4 (see Table 2) with no connecting line segments.

IMAGE places a summary line of the plotted information in the scroll display and on the OUTPUT (Printer) file. A typical summary line follows:

"PLOT CODES=8.0 SECTION 2 FILE TAPE10"

The PLOTPD command may be executed as many times as desired to produce the desired display.

HARDCOPY

To save the picture for postprocessing to a hardcopy recorder, the user types PLOT. The optional parameters are plot number (0 for all plots) and file name (default, PLOTF). The user must remember to catalog the appropriate local file name before leaving IMAGE.

The PLOT command is described more fully under the three-dimensional commands.

Instant hardcopy is available at any time from the Tektronix 4631 Hard Copy Unit. Whatever appears on the screen is copied in a matter of seconds simply by pushing the COPY key at the upper right of the keyboard.

VIEWING 3-D GEOMETRY DATA

DEFINITION OF MESH OF POINTS

IMAGE was designed to accept three-dimensional geometric input compatible with the XYZ Potential Flow Program.[1] The XYZ Potential Flow Program computes irrotational, incompressible potential flow about three-dimensional bodies of arbitrary shape. The three-dimensional bodies are described in terms of ordered points on the model's surface, generating a "mesh of points." The surface of the model is divided into defined sections (see Figure 4). Each section is described independently. This facilitates modifying a model surface by allowing the user to change the description of one section without affecting any other sections. Independent sections also allow for interchanging standard sections from model to model.

Each section is divided into ordered N and M reference lines whose intersections form the corners of the quadrilaterals which approximate the model's surface. The ordered points are the three-dimensional coordinates of these intersections of N and M lines. Geometric input consists of the X-, Y-, and Z-coordinates, the N and M indices, the section identification number, and an indicator for right- or left-handed coordinate system.

DATA FILE INPUT

RDXYZ

Data is accepted in the format compatible with input to the XYZ Potential Flow Program. The format of the data cards or card images on a file will be described in terms of the FORTRAN format statements used to read the data. The data must be exactly in the order and formats to be described:

a) ID Card (8A10), eighty characters of identifying information

b) Parameter Card (8A10), eighty characters of parameter information for the XYZ Potential Flow Program. There is a particular format for the parameters necessary to execute the XYZ Potential Flow

---

[1] Dawson, C.W. and Dean, J.S., "The XYZ Potential Flow Program," NSRDC Report 3892 (June 1972).

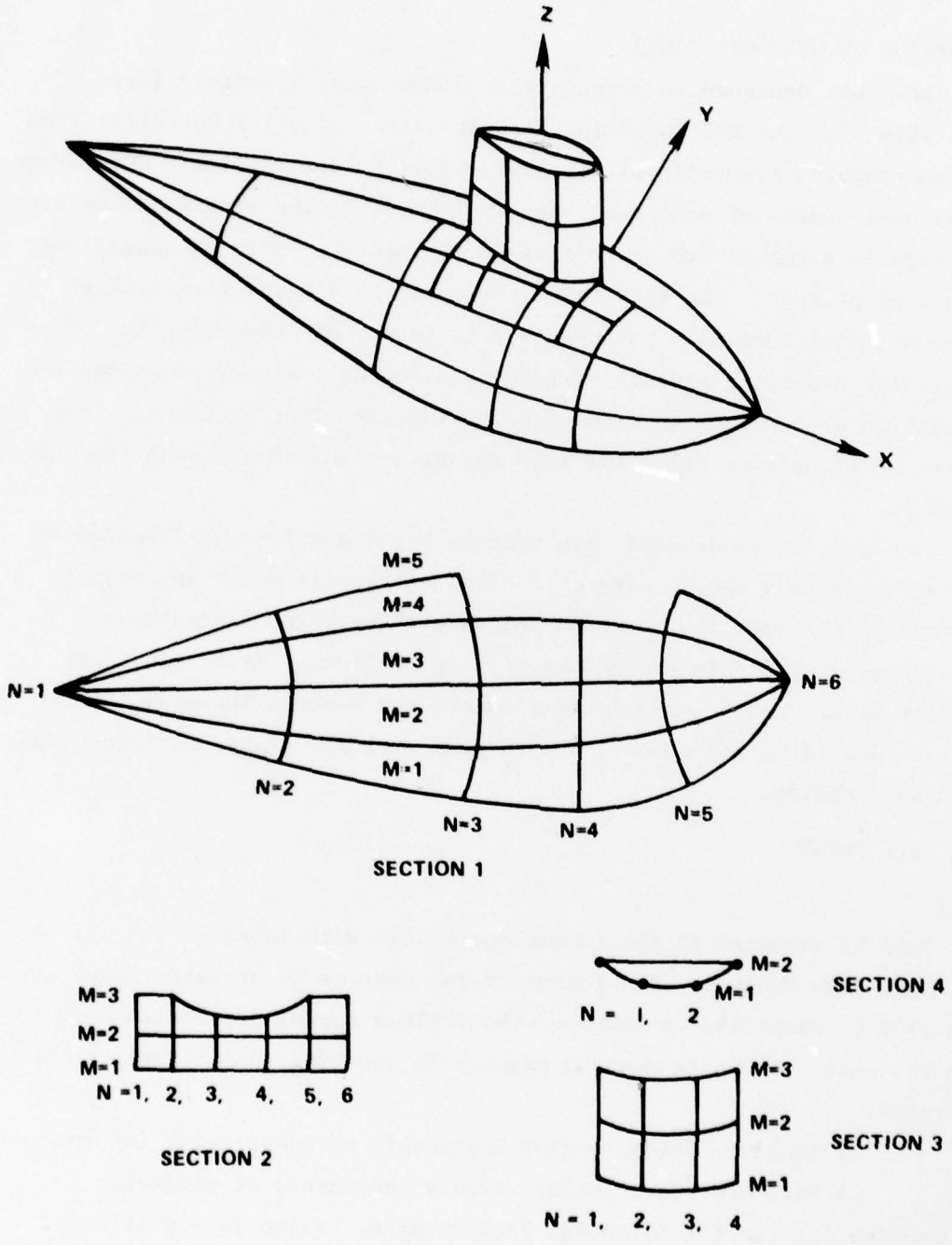


Figure 4 - XYZ Potential Flow Program Geometric Description

Program, but these parameters are not needed in IMAGE. If a user is not planning to use the XYZ Program, a dummy card may be input.

c) XI, YI, ZI, NI, MI, NS, NE (3E12.0, 3I4, I1), X,Y,Z point coordinates, N and M indices, section number, indicator NE = 0 (or blank) for right-handed system, NE ≠ 0 for left-handed.

Point coordinate cards will be read until 1000 points have been input or until an End of File is reached. As many sections as the user needs may be input with up to 1000 points each. Duplicate section numbers can be handled.

The command is

RDXYZ,lfn.

lfn is the local file name of the data file. If lfn is blank, the default file is INPUT.

File information is quickly flashed on the screen in the form of:

MESH "section no." READ "maximum n index" BY "maximun m index".

Other information about the data (possibly indicating an error) might appear, such as:

DUPLICATE MESH NUMBER CHANGED TO "no.",
MESH NUMBER TOO GREAT--IGNORED,
PREMATURE END OF INPUT DATA.

At the end of successful data file input, IMAGE displays

INPUT DATA RECORD PROCESSED.

All sections read are automatically displayed to give the optimum view of the whole object. The line of sight is always directly down one of the coordinate axes. The view geometry parameters used to produce the picture are displayed, giving the user a frame of reference for producing new views.

VIEW GEOMETRY

A general description of the view geometry may aid the user in making changes to the view geometry parameters to obtain the desired views of the model. See Figure 5 as a frame of reference.

Consider the model to be located in open space and the viewer free to move around it. Several parameters determine which view of the body will be displayed: the screen-to-observation-point distance (STOOP), the

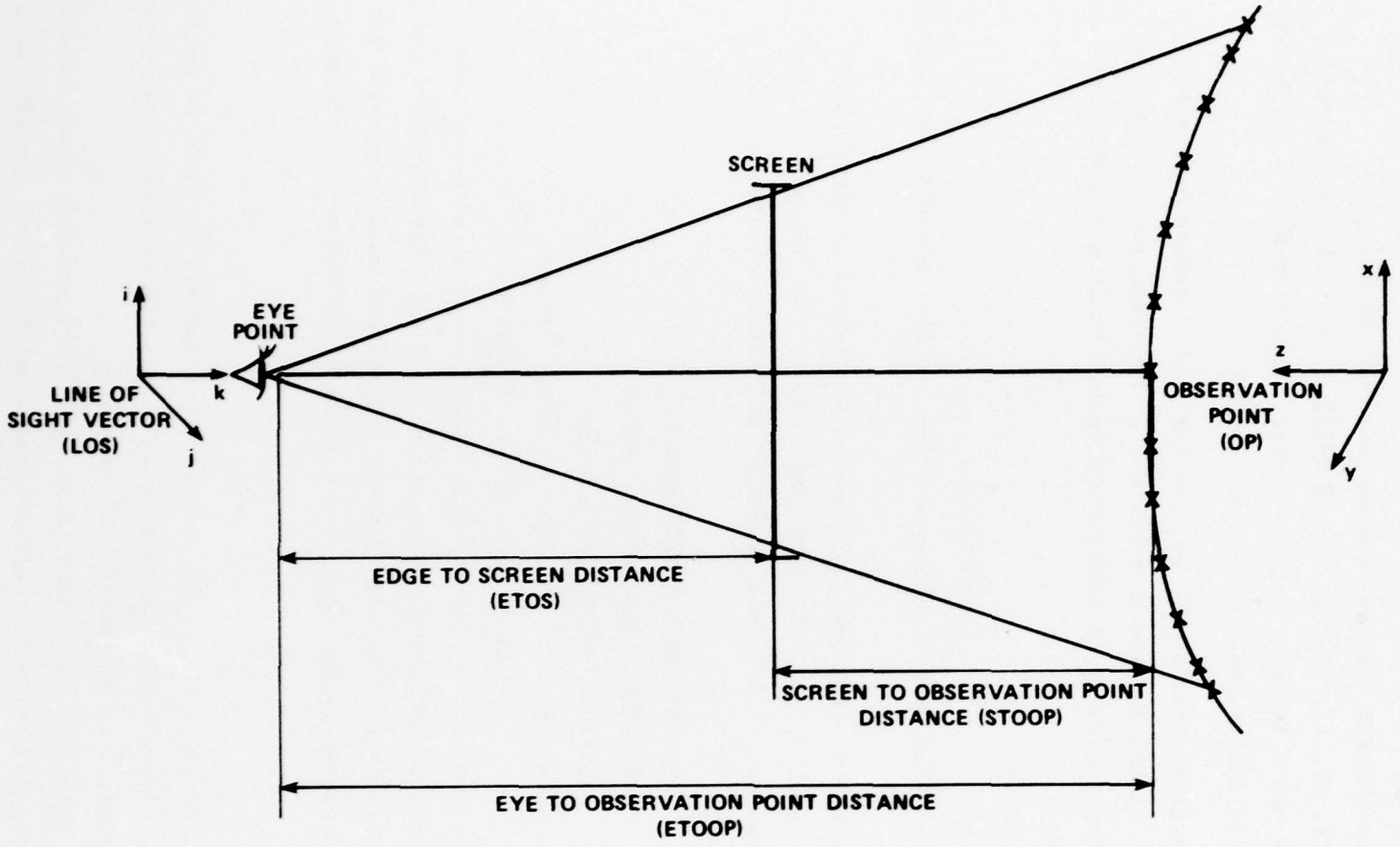


Figure 5 - View Geometry

eye-to-screen distance (ETOS), the line-of-sight vector (X_e,Y_e,Z_e), the observation point (X_0,Y_0,Z_0), and the degrees of rotation about the horizon.

The screen-to-observation-point distance determines how large or small the body will appear. The combination of eye-to-screen distance and screen-to-observation-point distance determines whether the whole body will fit on the screen. The screen-to-observation-point distance must be a positive value, no matter how small, since the eye-to-observation-point distance must be greater than the eye-to-screen distance.

The eye-to-screen distance determines whether the observer sees a perspective view, a wide angle view, or an orthogonal view. A very large eye-to-screen distance will produce an orthogonal view (i.e., no perspective). A very small eye-to-screen distance will produce a wide angle or very distorted view. The default value of 20 inches gives a fairly true perspective view.

The observation coordinates X,Y,Z can be thought of as the point on the body on which the user wishes to focus. In the default view, the observation point is the midpoint of the body.

The line-of-sight vector coordinates X,Y,Z determine the angle at which the body is observed. A value of 1. for one coordinate and 0. for the other two coordinates will produce a view down the respective coordinate axis. Values of 1.,1.,1. will produce a "45° angle" view.

The rotation angle rotates the body the specified number of degrees about the horizon. For example, if a user gives a rotation angle of 90° to a ship which is upright on the screen (or flat in the water), then the ship will be shown on its bow or stern.

CVG
SETV
ETOS
VGDISP

CVG

The command to change view geometry has nine parameters:

CVG, STOOP, L.O.S. DX, DY, DZ, O.P.X, Y, Z, ROT

(a) STOOP - screen-to-observation-point distance. This

parameter is used to find the eye-to-observation-point distance:

ETOOP = STOP + ETOS (see Figure 5)

(b) L.O.S. DX
(c) L.O.S. DY } line-of-sight vector
(d) L.O.S. DZ

The values of the vector components are initially set to give an optimum view, always directly down one coordinate axis (i.e., 1., 0., 0.; 0., 1., 0.; or 0., 0., 1.). The user may change these values to look down another axis or view the object from any angle.

(e) OP X
(f) OP Y } observation point on the model
(g) OP Z

Initially, the observation point is computed to be the center point of the model. The user should remember to keep the coordinate values within the respective ranges of data values. Each line segment is clipped in depth one screen-to-observation-point distance (STOOP) beyond the observation point.

(h) ROT - rotation of the horizon about the line-of-sight in degrees clockwise. The default value is 0.0.

Any number of these parameters may be changed by the user to produce the desired view. A blank indicates no change.

SETV

This command sets up the default view of a three-dimensional object. That view, as described previously, is to produce the largest possible picture of the whole object, always with a line-of-sight directly down one of the coordinate axes.

The observation point is set as the center point of the model. The line-of-sight is down the axis of the shortest side of the model. The length and height of the model are determined for a right-handed system. The maximum between length and height is found to determine a new scale. A new eye-to-screen distance is computed from the new scale. The eye-to-observation-point distance is then determined by a ratio which fits the full model into the frame of the screen.

The user may return to the original view at any time by typing in

SETV. The view geometry parameters will be printed at the top of the screen and the model will be displayed.

ETOS

This command is to change the eye-to-screen distance. The user types in

ETOS,n

where "n" is some number of inches. The default value is 20 inches, which approximates the true eye-to-screen distance and will produce normal perspective. A large number will give a telephoto perspective. A small number results in a wide angle view.

The new eye-to-screen distance is used to compute a new eye-to-observation point distance as a ratio of the old observation point distance to the old eye-to-screen distance. Each line segment is clipped in depth one screen's distance down the Y-axis, so that only a portion of the segment is visible.

VGDISP

This command will display the current view geometry at the top of the screen in the format:

EYE TO SCREEN DISTANCE - (ETOS)
EYE TO OP DISTANCE - (ETOOP=ETOS+STOOP)
SCREEN TO OP DISTANCE - (STOOP)
LINE OF SIGHT - (LOS DX, DY, DZ)
OP COORD - (OP DX, DY, DZ)
HORIZ. ROTATION - (DEG)
EYE POINT COORD - (EP DX, DY, DZ)

When this command is given, several checks for possible errors are made before the above information is displayed. If the line-of-sight vector does not contain a non-zero value,

LINE OF SIGHT NOT UNIQUE

will appear. If the eye-to-screen distance is not positive

EYE TO SCREEN DISTANCE ILLEGAL

will be printed. If the eye-to-observation-point distance is not greater than the eye-to-screen distance

OBSERVATION POINT BEHIND SCREEN

will be the error message.

DISPLAY OPTIONS

DISPLAY

CIDOPT

SCALE

ZOOM

DISPLAY

This command will display all the active mesh sections with the current display options.

The sections are processed, one at a time, looping down the N-direction first, then the M-direction. Each line segment is clipped in depth one screen-to-observation-point distance (STOOP) beyond the observation point. This is considered an invisible "back wall" and only the portion of the segment in front of the wall is visible. The user could actually clip off back sections by picking an appropriate observation point and screen-to-observation-point distance. This technique could be useful since there is no capability to remove hidden lines between sections. That is, if one section appears in back of another in a particular view, no lines will be removed, even with the hidden line capability of CIDOPT.

There are no parameters for the DISPLAY command.

CIDOPT

This command enables the user to change display options, allowing him to display M- and/or N-index lines only, or points only, and/or mesh boundaries only, or to eliminate hidden lines. The type-in command is:

CIDOPT,IDOPT,JDOPT,D.

IDOPT - Option Selection (1 to 4, default 1)

= 1 Draw N- and M-lines

= 2 Draw N-lines only

= 3 Draw M-lines only

= 4 Draw points only

JDOPT - Option Selection (1 to 3, default 1)

= 1 Draw all lines or points

= 2 Draw mesh boundaries only

= 3 Draw "visible" lines or points only

D - Display inhibit (.NOT.D, Do not display)

CIDOPT provides the user with many possible display options. JDOPT=2 will produce an outline of the body and also show the boundary lines beteen sections. JDOPT=3 will eliminate hidden lines within each mesh section. Hidden lines are not removed between sections at this time.

Unless "NOT D", all active mesh sections will be displayed with the selected display options.

SCALE

This command will change the number of inches in one model unit. The type-in command is: SCALE,UNITS.

UNITS - number of inches/model unit.

Default = 1.0 inch

The message is returned on the screen:

SCALE 1 MODEL UNIT = "UNITS" INCHES,

1 INCH = "1.0/UNITS" MODEL UNIT

All distances and point coordinates in the view geometry are in model units.

ZOOM

As the command word indicates, ZOOM allows the user to magnify a particular part of his model. This is done with a tracking cross. The user types in ZOOM, and IMAGE displays the tracking cross on the screen. The user then has three options:

(1) SP CR* without moving the tracking cross- IMAGE moves in halfway to body;

(2) Move the tracking cross to some point on the body SP CR* - IMAGE moves in halfway to a new observation point that is the point on the body closest to the indicated point;

(3) Move the tracking cross to a point defining a corner of an imaginary box around a particular area of the body SP CR*; move the tracking cross to a second point on the diagonal to the first point SP CR* - the corners define nearly or exactly the edges of the new frame and a new

* Space, Return

observation point is computed as the point on the body closest to the midpoint of the two points defining the box.

ZOOM may be used several times in sequence. The user may return to the whole model by calling SETV.

POINT MANIPULATION

FIND

CNGPT

FIND

FIND will find the point on the body closest to the tracking cross and save the mesh section numbers M-index and N-index for future use.

The user types in FIND. The tracking cross appears on the screen, and the user then moves it as close as possible to the desired point on the body. IMAGE returns with the message:

POINT CLOSEST TO CROSS IS (N index,M index) IN SECTION (mesh section no.)

The user may then want to change his point with the next command to be described, CNGPT.

CNGPT

This command will change the point coordinates of the point selected by the tracking cross.

The type-in command is: CNGPT, X,Y,Z.

X - new X-coordinate

Y - new Y-coordinate

Z - new Z-coordinate

The user may change as many or as few of the coordinate values as he wishes. A blank indicates no change. The result will be displayed:

POINT "N index,M index" IN SECTION "mesh section no."

CHANGED TO COORD. "X, Y, Z"

CNGPT will change the last point selected by the tracking cross. If no point has been identified with FIND, an error message will appear: "NO POINT IDENTIFIED TO CHANGE". The body will be redisplayed with the changed point (and all current display options).

DATA OUTPUT

SAVXYZ

PLOT

SAVXYZ

After a user has viewed his model, seen some errors, located the particular points and changed them, he can then save this modified body with SAVXYZ. The type-in command is: SAVXYZ,NEWDF.

This command will produce a new card deck containing three-dimensional data in the format of the XYZ Potential Flow analysis input cards. Card images will be written on the designated file (default, NEWDF). Any changes made during the session will be saved. The message will be displayed:

"n" SECTIONS SAVED ON FILE "NEWDF".

The user must remember to CATALOG this file before leaving IMAGE.

PLOT

This command will produce a plot on a file for postprocessing to a hardcopy device, such as the CALCOMP or SC 4060 plotter. (See Appendix B for CCL procedure to postprocess to the CALCOMP 936). There are several forms of the type-in command:

PLOT - plots current frame on file PLOTF

PLOT,N - plots frame N on file PLOTF

PLOT,0 - plots all frames on file PLOTF

PLOT,,ABC - plots current frame on file ABC

PLOT,N,ABC - plots frame N on file ABC

The default file name is PLOTF. The user must remember to catalog the plot file before leaving IMAGE, in order to be able to postprocess the file for hardcopy.

This command is also available to the two-dimensional cross plotting.

On the Tektronix terminal, instant hardcopy is available at any time by pressing the COPY button on the upper right of the keyboard.

CALCOMP 3-D PACKAGE

Also included in IMAGE is the ability to generate three-dimensional perspective drawings using the CALCOMP 3-D system. With this package, the user is able to obtain a gridded, perspective view of any surface that can be represented as a single-valued function of two variables. Additional features are:

1) the ability to grid ungridded input data
2) the ability to smooth the grid representing a surface
3) the ability to plot and annotate the coordinate axes

There are 16 types of control cards. Varying combinations of these control cards and one or more sets of data must be written to a file as input to THRED. The IMAGE command is:

THRED, INPUT FILE NAME (DEFAULT-INPUT).

THRED cannot be run after RDXYZ has been used.

The user should consult the CALCOMP Manual, "THREE-D, A Perspective Drawing Software System," before attempting to use the THRED command in IMAGE. The manual contains detailed descriptions of the capabilities of the package, input control cards, and output plots.

CONCLUSIONS

IMAGE is a useful and versatile interactive computer graphics tool. However, there are some limitations in data display and some commands may seem cumbersome. Presently, a new Interactive Data Display System is being developed, which will be much more flexible and powerful and which will incorporate and expand the capabilities available in IMAGE.

Until the new system is available, IMAGE will provide reliable and easy display of 2-D data arrays and 3-D geometric data with considerable savings of time and money to the user.

APPENDIX A - LIST OF COMMAND WORDS

```
?
ACTIVE COMMAND WORDS -
ABORT      END JOB WITH NO EXIT PROCESSING
CIDOPT     CHANGE DISPLAY OPTIONS
CNGPT      CHANGE POINT COORDINATES OF POINT SELECTED BY TRACKING CROSS
CPTIME     DATE, TIME, ACCUMULATED CP TIME DISPLAY
CREATEF    CREATE A NEW PERMANENT FILE
CVG        CHANGE VIEW GEOMETRY
DATE       DATE, TIME, ACCUMULATED CP TIME DISPLAY
DISPDF     DISPLAY DAY FILE
DISPLAY    DISPLAY ALL ACTIVE MESH SECTIONS
DMDMP      PRINT OCTAL AND DISPLAY CODE DUMP OF A DATA MANAGER FILE.
DONE       NORMAL END OF JOB
ETOS       SET EYE TO SCREEN DISTANCE
EXIT       NORMAL END OF JOB
FILES      DISPLAY LIST OF FILES FOR THIS JOB
FIND       FIND POINT ON BODY CLOSEST TO TRACKING CROSS
GRIDPD     CREATE NEW GRID/S FOR Y VS. X CROSSPLOTS
LINES      CHANGE NUMBER OF LINES IN TEXT DISPLAY SCROLL
PLOT       PRODUCE PLOT ON FILE FOR POST-PROCESSING TO HARDCOPY RECORDER
PLOTPD     CROSSPLOT Y VS. X
PRUDMP     PRINT OCTAL AND DISPLAY CODE DUMP OF A FILE BY PRU.
Q8SUB      SUBSTITUTE NEW NAME FOR FORTRAN PROGRAM FILE
RDCDS      READ TYPEIN COMMANDS FROM CARDS
RDPD       READ Y VS. X PLOT DATA FROM A FILE
RDXYZ      READ XYZ POTENTIAL FLOW ANALYSIS INPUT GEOMETRY CARDS
RPV        SET REPRIEVE CONDITION CODE
SAVXYZ     SAVE XYZ POTENTIAL FLOW ANALYSIS CARDS OF CURRENT 3-D DATA
SCALE      CHANGE NUMBER OF INCHES IN ONE MODEL UNIT
SCALEPD    SPECIFY RANGE OF Y VS. X CROSSPLOTS
SETV       SET UP THE DEFAULT VIEW OF 3-D OBJECT
THRED      READ CALCOMP 3-D PACKAGE INPUT CARDS AND SHOW PLOT
TIME       DATE, TIME, ACCUMULATED CP TIME DISPLAY
USEF       USE SPECIFIED PERMANENT FILE
VGDISP     DISPLAY VIEW GEOMETRY PARAMETERS
ZOOM       ZOOM IN CLOSER TO BODY AS INDICATED BY TRACKING CROSS
```

APPENDIX B - CYBER CONTROL LANGUAGE (CCL) PROCEDURES

Cyber Control Language (CCL) provides control card manipulation. With one control statement, a set of control statements is initiated. Several CCL procedures from the permanent file PROCFIL, ID=CAMV are useful in using IMAGE.

After LOGIN on the Tektronix, the user types

BEGIN,MYPRO,,CAMV,TIMAGE.

and presses RETURN, to begin execution of the IMAGE program.

A procedure is available to initiate postprocessing of the PLOTF file to the CALCOMP 936 directly from the terminal. The user must have a tape available at the central site. The user types

BEGIN,MYPRO,,CAMV,PLT936,"ID","AC","Code","Bin.","Wid.","Ht".

The parameters are:

ID - four-character user ID. This ID is used to catalog the PLOTF and to identify the job.

AC - ten-character access number for accounting.

Code - User's code.

Bin - visual number identifying the tape. This should be a Diebold Cabinet tape with an assigned number obtained from User Services. This is the tape on which the CALCOMP output is written.

Wid - width of the CALCOMP paper. Default = 100 inches.

Ht - height of the CALCOMP paper. Default = 11 inches.

The ID, AC, Code, and Bin should be input by the individual user. The type-in may be truncated after the Bin number if the user chooses the default paper size. The Plot File (PLOTF) is automatically cataloged. At the end of the job, the lowest cycle PLOTF is purged. The job will be batched to the central site, where the user may pick up his output later.

DISTRIBUTION LIST

Copies:		
2	NAVSEC	
	1	SEC 6136/E.N. Comstock
	1	SEC 6136/B. Keane
12	DDC	

DTNSRDC DIST:

Copies:		
1	1500	W.E. Cummins
1	1552	J.H. McCarthy, Jr.
1	1564	J.P. Feldman
3	1568	S.L. Bales E.W. Foley T.R. Applebee
1	1572	M.D. Ochi
1	1576	W.E. Smith
1	1600	H.R. Chaplin
1	1619	P.S. Montana
1	1700	W.W. Murray
2	1720.3	R.F. Jones P.N. Roth
1	1740.5	S. Zilliacus
1	1800	G.H. Gleissner
1	1802	F.N. Frenkiel
1	1805	E.H. Cuthill
2	1809.3	D. Harris
1	1820	A.W. Camara
1	1840	H.J. Lugt
1	1843	J.W. Schot
100	1843	M.B. Marquardt
1	1844	S.K. Dhir
1	1850	T. Corin
1	1854	C. Johnson
1	1870	G.R. Gray
1	1900	M.M. Sevik
1	522.1	Unclassified Lib (C)
1	522.2	Unclassified Lib (A)

DTNSRDC ISSUES THREE TYPES OF REPORTS

1. DTNSRDC REPORTS, A FORMAL SERIES, CONTAIN INFORMATION OF PERMANENT TECHNICAL VALUE. THEY CARRY A CONSECUTIVE NUMERICAL IDENTIFICATION REGARDLESS OF THEIR CLASSIFICATION OR THE ORIGINATING DEPARTMENT.

2. DEPARTMENTAL REPORTS, A SEMIFORMAL SERIES, CONTAIN INFORMATION OF A PRELIMINARY, TEMPORARY, OR PROPRIETARY NATURE OR OF LIMITED INTEREST OR SIGNIFICANCE. THEY CARRY A DEPARTMENTAL ALPHANUMERICAL IDENTIFICATION.

3. TECHNICAL MEMORANDA, AN INFORMAL SERIES, CONTAIN TECHNICAL DOCUMENTATION OF LIMITED USE AND INTEREST. THEY ARE PRIMARILY WORKING PAPERS INTENDED FOR INTERNAL USE. THEY CARRY AN IDENTIFYING NUMBER WHICH INDICATES THEIR TYPE AND THE NUMERICAL CODE OF THE ORIGINATING DEPARTMENT. ANY DISTRIBUTION OUTSIDE DTNSRDC MUST BE APPROVED BY THE HEAD OF THE ORIGINATING DEPARTMENT ON A CASE-BY-CASE BASIS.