NASA CONTRACTOR REPORT

NASA CR-2430

NASA CR-2430

IMAGE: A COMPUTER CODE FOR GENERATING PICTURE-LIKE IMAGES OF AEROSPACE VEHICLES

by C. R. Glatt

Prepared by
AEROPHYSICS RESEARCH CORPORATION
Hampton, Va. 23666
for Langley Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • SEPTEMBER 1974

1. Report No. NASA CR-2430	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle IMAGE: A Computer Code for Generating Picture-Like Images of Aerospace Vehicles.		5. Report Date September 1974
		6. Performing Organization Code
7. Author(s) C. R. Glatt		8. Performing Organization Report No. LTN-11
		10. Work Unit No.
9. Performing Organization Name and Address AEROPHYSICS RESEARCH CORPORATION P. O. Box 7007 Hampton, Virginia 23666		11. Contract or Grant No. NAS 1-12008
		13. Type of Report and Period Covered Contractor Report
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington D. C. 20546		14. Sponsoring Agency Code

15. Supplementary Notes

Topical Report

16. Abstract

An independent geometry display program called IMAGE is presented. The use of the program permits the editing of geometric configurations and the monitoring of geometric perturbations.

17. Key Words (Suggested by Author(s)) Design, synthesis, ODIN, geometry, display	18. Distribution Statement Unclassified, Unlimited STAR Category 31

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	68	$3.75

*For sale by the National Technical Information Service, Springfield, Virginia 22151

Page Intentionally Left Blank

PREFACE

This report was prepared for Contract NAS 1-12008, Expansion and Extension of the ODIN/RLV Computer Program - Task 2, Evaluate and Improve the Existing ODIN Program Library. The contract was funded by the National Aeronautics and Space Administration, Langley Research Center, Space Systems Division.

The ODIN procedure is a design analysis technique which allows the use of existing computer codes as part of a larger simulation. Communication of information among computer codes is accomplished by means of a data base repository accessible and managed by the ODIN executive computer code, DIALOG.

The objective of the contract was the development of an independent geometry display program. The result was a computer code for editing geometry and monitoring geometric perturbations.

Page intentionally left blank

CONTENTS

LIST OF FIGURES

IMAGE: A COMPUTER CODE FOR GENERATING PICTURE-LIKE IMAGES FOR AEROSPACE CONFIGURATIONS

BY C. R. Glatt

Aerophysics Research Corporation

SUMMARY

The IMAGE program uses a surface definition based upon quadrilateral elements to describe a picture-like drawing of an arbitrary vehicle. The computer program is an important component of the Optimal Design Integration (ODIN) System. The program is used to visually check geometric data input and monitor geometric perturbations. In addition, the picture presentation can be annotated with textual information.

Usually, the user is interested in three-view drawings, however, arbitrary viewing angles may be specified as part of the normal input. In this manner, pseudo perspective drawings of the geometry may be obtained. The drawings are not true perspective drawings as the vanishing point is at infinity. This report describes the use of IMAGE, both as an independent program and as an ODIN element.

INTRODUCTION

The generation of geometry for aerospace configurations in digital format is one of the most tedious tasks in the design analysis process and is also difficult to check. The program described in this report is designed specifically to provide a pictorial representation of the digital geometric input. The program capability provides a visual check for errors in geometry and a picture presentation which can be annotated with textual information.

The IMAGE program is an extraction from the program of reference 1, but has undergone extensive modification and simplification in an effort to provide a more useful addition to the ODIN (Optimal Design Integration) program library (see reference 2). Although developed primarily for use in the ODIN system, the IMAGE program is equally useful as an independent program. The program was written for the CDC 6600 computer. Some coding peculiar to the CDC machine must be changed for use on other machines but this coding is a small portion of the total program.

This report was prepared for Contract NAS 1-12008, A Study Effort to Extend and Improve the ODIN Procedure - Task 2, Evaluate and Improve the Existing ODIN Program Library. The objective of this task was the improvement and redocumentation of the IMAGE program as a member of the ODIN library. The contract was funded by the National Aeronautics and Space Administration, Langley Research Center, Space Systems Division.

SURFACE MODEL

The surface shape of the configuration is described to the IMAGE program by ordered sets of points in three dimensional space. The geometry is specified in 80 column (BCD) card format. It may be input directly or generated by another program and passed to the IMAGE program as a binary or BCD file. The program rotates the geometric data to prespecified viewing angles then transforms it onto a plane coincident with the plane of the paper. The geometry is reordered into quadralateral elements for plotting.

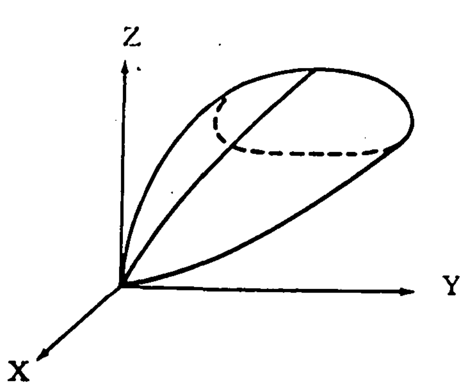
A grouping of four surface points is used to describe a quadralateral surface element. An organization of a large number of related elements forms a component. A number of components may be used to give a complete description of the configuration. Each component is an independent unit of geometry which may be drawn separately or collectively with other components. In general, the equations for the geometry rotations described here apply to any orientation angle. The equations required to produce the perspective drawings are derived in the following paragraphs. Figure 1 is a collection of component geometrics representing a shuttle orbiter configuration arranged in a three view drawing by the program.

Coordinate System

Each point on the surface is described by its coordinates in the body reference coordinate system.

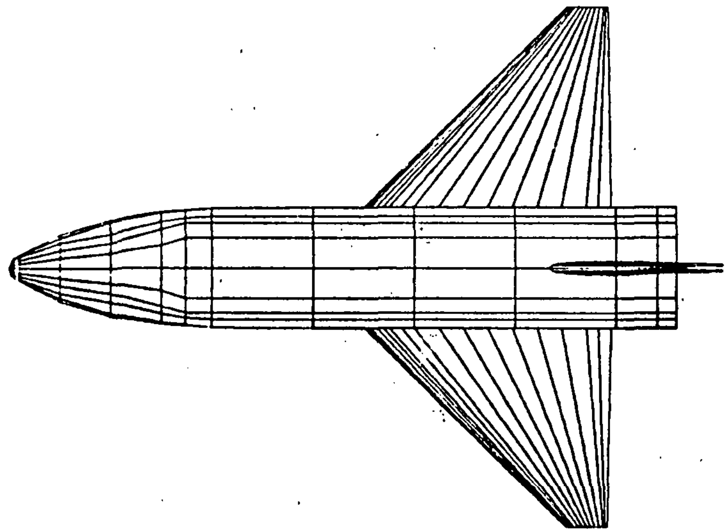
$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

The body reference coordinate system is assumed to be a conventional right-handed Cartesian system as illustrated below:



O D I N / I M A G E PICTURE DRAWING PROGRAM

ORBITER DESIGN 25K P/L 1 INCH= 132.000 INCHES



VEHICLE CHARACTERISTICS

1. MASS PROPERTIES

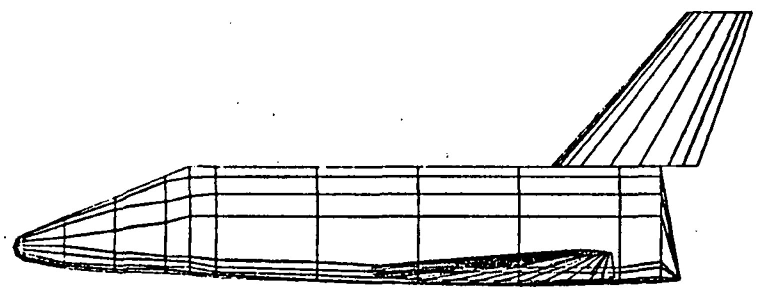
LANDED WEIGHT	196994.5	LB
ENTRY WEIGHT	186710.7	LB
DESIGN CG. SUBSONIC. 25K P/L	69.85972	FT
DESIGN CG HYPERSONIC. 13K P/L	69.49021	FT

2. GEOMETRY

BODY LENGTH	110.0000	FT
TOTAL WING AREA	2500.000	SQFT
CHORD. THEO ROOT	50.89356	FT
CHORD. TIP	6.851471	FT
ASPECT RATIO	3.000000	
LEADING EDGE SWEEP ANGLE	45.00000	DEG
TRAILING EDGE SWEEP ANGLE	0.000000	DEG
ELEVON AREA. TOTAL	375.0000	SQFT
EXPOSED WING LOCATION	58.49000	FT

3. AERODYNAMIC CHARACTERISTICS

DESIGN TRIM LIFT COEFFICIENT	.6257300	
DESIGN MINIMUM LANDING SPEED	187.8537	KNOTS
MAX TRIM ALPHA HYPERSONIC DESIGN COND	88.63326	DEG



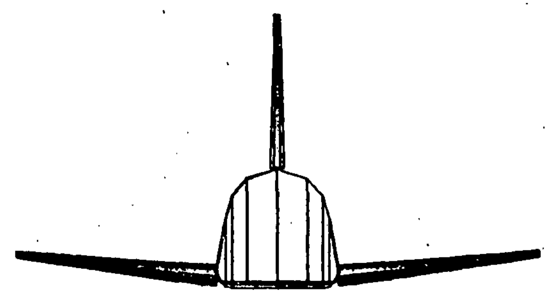
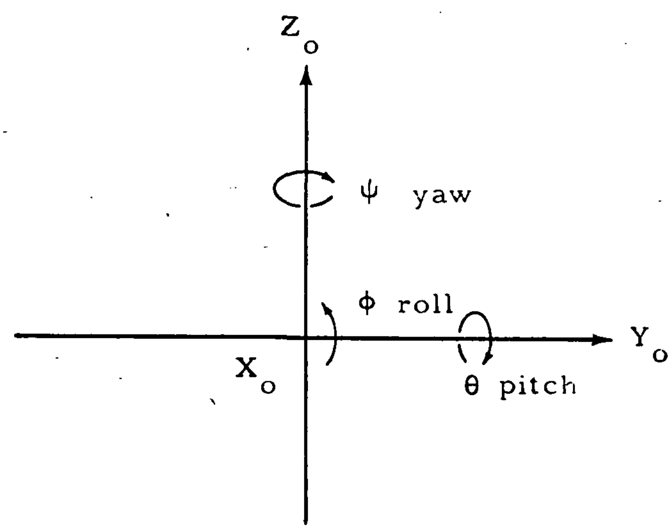


FIGURE 1 ILLUSTRATION OF PICTURE AND TEXT OPTIONS IN IMAGE.

Coordinate Transformations

To create the perspective drawings illustrated in this report each surface point on the body must be rotated to the desired viewing angle and then transformed into a coordinate system in the plane of the paper. With zero rotation angles the body coordinate system is coincident with the fixed system in the plane of the paper.



The rotations of the body and its coordinate system to give a desired viewing angle are specified by a yaw-pitch-roll sequence (Ψ,θ,ϕ). The rotation is given by the following relationship:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix}\phi\end{bmatrix}\begin{bmatrix}\theta\end{bmatrix}\begin{bmatrix}\psi\end{bmatrix} \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix}$$

This sequence is important to remember when describing the desired viewing angles to the IMAGE program.

The rotation matrices ψ, θ and ϕ are given by:

$$\left[\psi\right] = \begin{bmatrix} \cos\psi & \sin\psi & o \\ -\sin\psi & \cos\psi & o \\ o & o & 1 \end{bmatrix}$$

$$\left[\theta\right] = \begin{bmatrix} \cos\theta & o & -\sin\theta \\ o & 1 & o \\ \sin\theta & o & \cos\theta \end{bmatrix}$$

$$\left[\phi\right] = \begin{bmatrix} 1 & o & o \\ o & \cos\phi & \sin\phi \\ o & -\sin\phi & \cos\phi \end{bmatrix}$$

or

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \left[E\right] \begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix}$$

where

$$\left[E\right] = \left[\phi\right]\left[\theta\right]\left[\psi\right]$$

Since each point on the surface is given by its coordinates in the X, Y, Z system, its position in the fixed coordinate system (X_o, Y_o, Z_o) may be found by the inverse of the above process:

$$\begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix} = \left[E\right]^{-1} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

If this operation is carried out, the resulting relationship is obtained.

$$\begin{bmatrix} X_o \\ Y_o \\ Z_o \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi & -\sin\psi\cos\phi+\sin\theta\cos\psi\sin\phi & \sin\psi\sin\phi+\sin\theta\cos\psi\cos\phi \\ \cos\theta\sin\psi & \cos\psi\cos\phi+\sin\theta\sin\psi\sin\phi & -\cos\psi\sin\phi+\sin\theta\sin\psi\cos\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$X_o = X(\cos\theta\cos\psi) + Y(-\sin\psi\cos\phi+\sin\theta\cos\psi\sin\phi) + Z(\sin\psi\sin\phi+\sin\theta\cos\psi\cos\phi)$$

$$Y_o = X(\cos\theta\sin\psi) + Y(\cos\psi\cos\phi+\sin\theta\sin\psi\sin\phi) + Z(-\cos\psi\sin\phi+\sin\theta\sin\psi\cos\phi)$$

$$Z_o = X(-\sin\theta) + Y(\cos\theta\sin\phi) + Z(\cos\theta\cos\phi)$$

We may now use these last two equations to transform a given point on the body (X, Y, Z) with a specified set of rotation angles (ψ,θ,ϕ) into the plane of the paper (the Y, Z system). With the CALCOMP library subroutines it is a simple matter to plot these data and to connect the related points with straight lines.

The above relationships completely describe the transformation required to rotate every point on the vehicle to the desired angle, then into the plane of the paper. However, the resulting drawing is difficult to interpret because all hidden lines are drawn. Further, since only one half, or one quarter (for symmetrical vehicles) of the coordinate points are usually input, some additional calculations are desirable.

The surface points are grouped into quadrilateral surface elements for analysis by the program. Each quadrilateral is drawn as an individual 'curve.' The collection of all 'curves' represents the complete configuration. The technique also provides a convenient means of limiting the drawn lines to those normally seen by the viewer at the defined viewing angles.

Each input element is replaced by a plane quadrilateral surface made up of the four lines connecting the points. The quadrilateral characteristics are used to determine the visibility of the four lines. The quadrilateral characteristics include the area, centroid and the direction cosines of the surface unit normal. The surface unit normals may be transformed through the required rotation angles just as was done for the individual points. The resulting value of the component of the unit normal in the X_o direction (out of the plane of the paper) may be found from the following equation:

$$n_{x_o} = n_x(\cos\theta\cos\psi)+n_y(-\sin\psi\cos\phi+\sin\theta\cos\psi\sin\phi)+n_z(\sin\psi\sin\phi+\sin\theta\cos\psi\cos\phi)$$

where n_x, n_y, n_z are the components of the surface unit normal in the vehicle reference system.

If n_{x_o} is positive, then the surface element is facing the viewer. If n_{x_o} is negative, the element faces away from the plane of the paper. This result is used in the program to provide the option of deleting most of those elements on a vehicle that normally could not be seen by a viewer. The picture is thus made more realistic and easier to interpret. Confusing elements which are on the back side of a component do not appear. No criterion is provided, however, for the deletion of those elements that face the viewer but are blocked by other body components. If desired, more realistic drawings may be obtained by selective deletion of sections or half sections and by a proper selection of viewing angle.

PROGRAM USAGE

The computer program usage requirements described in this section are generally oriented toward the CDC 6600 computer version and specifically toward the Langley Research Center (LRC) installation. The actual program input requirements described are largely applicable wherever the program is installed but the control cards for the retrieval and execution of the program will differ from computer to computer and from installation to installation.

The use of the program requires three types of input; the control cards, the actual program input and the post processing instructions for generating CALCOMP plots. Figure 2 illustrates the deck setup for executing the program and indicates the separate submittal of a plot request card. One input case is illustrated. Multiple cases may be computed by repeating the case data illustrated.

Control Cards

The program input data is preceded by the operating system control cards required to retrieve from permanent storage and execute the program. Figure 3 illustrates the control cards for three different ways of using IMAGE at LRC. Figure 3A shows the method of execution from the stored machine language program. Figure 3B illustrates a compile, load and execute sequence from stored source code, assuming program modifications are desired. Figure 3C illustrates the use of IMAGE within the ODIN (Optimal Design Integration) system. See reference 2 for ODIN system usage.

The dashes on the JOB, USER and REQUEST cards indicate missing information which the user must supply in accordance with LRC computer complex accounting procedures. Further, the wedge number (program storage location in data cell) is subject to change. The latest wedge number is available from LRC.

The user must supply the estimated run time in CPU seconds, the octal field length for the job and the number of operating systems (O/S) calls. These are given in the above order on the JOB card. Typical values for a single case are tabulated below:

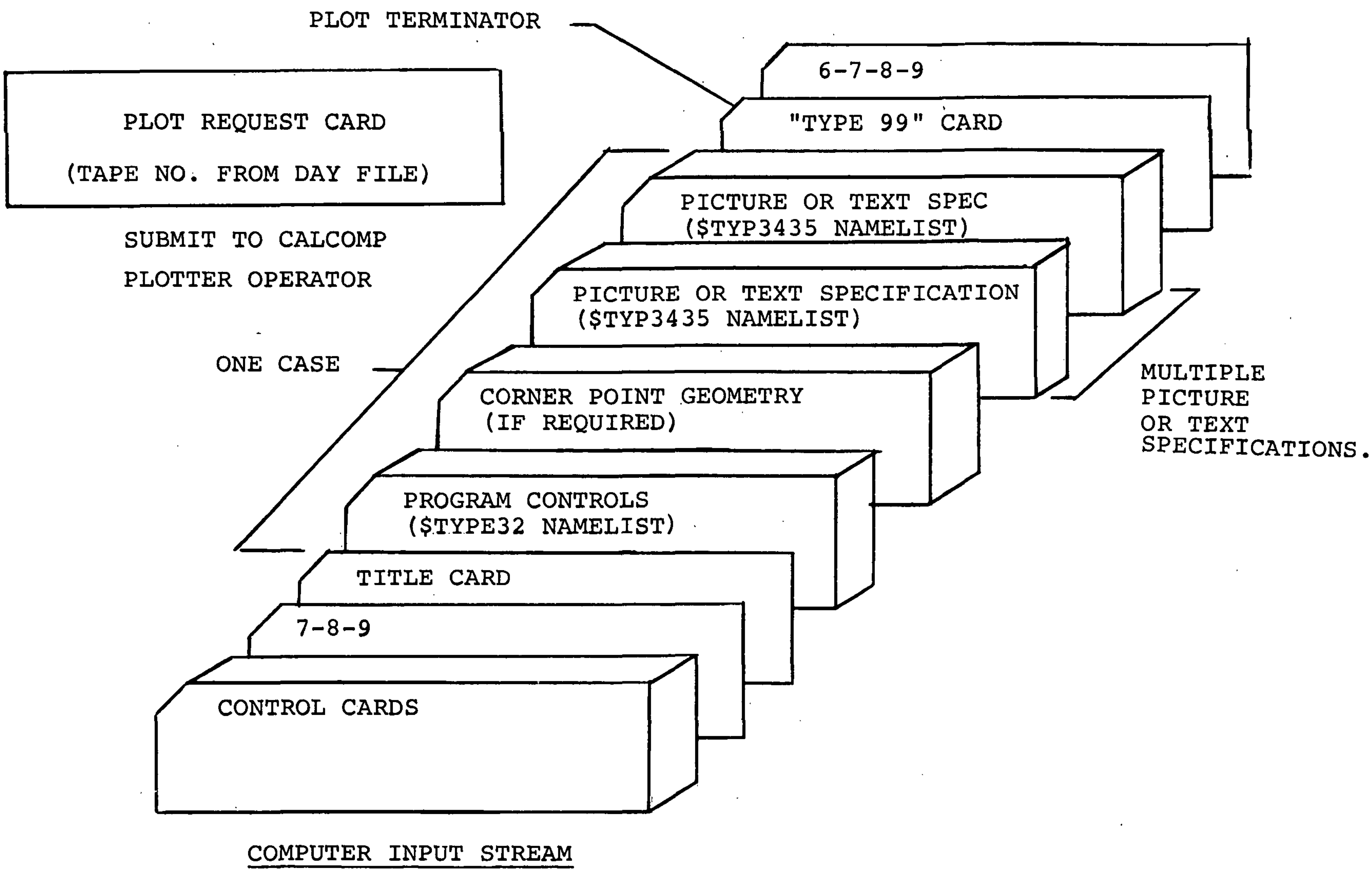


FIGURE 2 DECK SETUP FOR IMAGE.

```
JOB,1,20,35000,500. - - - -
USER. - - -
FETCH,A3647,SPR___,BINARY,,OIMAGE.
OIMAGE.
REQUEST,TAPE98,HI.CALTP,RIL, - - -
REWIND,CALTPE,TAPE99.
COPYBF,CALTPE,TAPE99.
UNLOAD,TAPE99.
7-8-9
(IMAGE CASE DATA)
6-7-8-9
```

REQUIRED ONLY IF PLOTS ARE REQUESTED

FIGURE 3A EXECUTION OF STORED PROGRAM

```
JOB,1,40,45000,800. - - - -
USER. - - -
FETCH,A3647,SPR___,SOURCE.
RUN,S,,,SCFILE.
LGO.
REQUEST,TAPE98,HL.CALTP,RIL, - - -
REWIND,CALTPE,TAPE99.
COPYBF,CALTPE,TAPE99.
UNLOAD,TAPE99.
7-8-9
(MODS TO SOURCE PROGRAM, IF ANY)
7-8-9
(IMAGE CASE DATA)
6-7-8-9
```

REQUIRED ONLY IF PLOTS ARE REQUESTED

FIGURE 3B COMPILE, LOAD AND EXECUTE

```
'EXECUTE IMAGE'
(IMAGE CASE DATA)
7-8-9
'EXECUTE PLOTSV'
7-8-9
```

REQUIRED ONLY IF PLOTS ARE REQUESTED

FIGURE 3C EXECUTE VAMP WITHIN ODIN SIMULATION

FIGURE 3 ILLUSTRATIONS OF CONTROL CARDS REQUIRED FOR IMAGE.

	CPU	FIELD LENGTH	O/S CALLS
EXECUTE ABSOLUTE BINARY PROGRAM	20	35000	500
COMPILE, LOAD AND EXECUTE	40	45000	800
EXECUTE WITHIN THE ODIN SYSTEM	30	56000*	600

*Minimum size for the executive system (reference 2).

The tabulated values above will vary with the complexity of the configuration (number of elements), the number of cases and the number of plots requested. A good rule to follow is to allow ample CPU and O/S calls in the first run, then use the day file results to estimate these parameters for similar configurations.

General Program Input

The IMAGE program input will typically consist of four types of input.

1. Title card in free field format.
2. Program controls ($TYPE32) in NAMELIST format.
3. Surface model data as formatted corner-point geometry.
4. Picture specifications ($TYP3435) in NAMELIST format.

The program controls and picture specifications are input in standard NAMELIST input format. Two NAMELIST reads are provided for this purpose as illustrated in figure 4.

The use of NAMELIST input was chosen for the following reasons:

1. It is a simple name oriented input easily understood by most computer users.
2. The format is standard and does not require relearning from program to program.
3. It is easily modified by the engineer or programmer when adding input variables to the program.

{TITLE (óne card - first 59 characters)

$TYPE32

[Geometry Scaling and Control Options]

$

[Element Data - only if unit 5 was specified by the data set above]

$TYP3435

[Picture Drawing or Text Options]

$

$TYP3435

[Picture Drawing or Text Options (if LAST=1]

$

{END OF IMAGE DATA col. 71 ——→ 99

FIGURE 4 ILLUSTRATION OF INPUT STREAM TO IMAGE FOR TWO VIEWS.

When a NAMELIST read is encountered in the program, the entire input file is scanned up to an end-of-file or a record with a dollar ($) in column 2 followed immediately by the NAMELIST name requested by the programs. Succeeding data items are read until a second dollar ($) is encountered signifying the end of the NAMELIST. Any data on the input file before the requested NAMELIST is found will be ignored. All data between the opening and closing dollar is interpreted by the NAMELIST input routine. The data item within the NAMELIST statement may be in any of three forms:

$V = C,$

$A = D_1, \ldots, D_j,$

$A(n) = D_1, \ldots, D_m,$

V is a variable name; C is a constant; A is an array name and n is an integer constant subscript, $D_1, \ldots D_m$, are simple constants or repeated constants of the form k*C, where k is the repetition factor. Constants may be real, integer, hollerith or logical. Hollerith constants are preceded by nH where n is the number of characters in the hollerith constant. Logical constants are of the form .TRUE. (or T) or .FALSE. (or F).

Data items and constants must be separated by commas. The number of constants, including repetitions given for an unscripted array must equal the number of elements in that array. For a subscripted array name, the number of constants need not be equal but may not exceed the number of array elements needed to fill the array. More than one card may be used for input data and arrays may be split between cards. All except the last record must end with a constant followed by a comma and no sequence numbers may appear. The first column of each record is ignored. The set of data items may consist of any subset of the variable names associated with the NAMELIST name and the name need not be any particular order. More details on the use of NAMELIST are available in any FORTRAN user's guide, but the above description should be sufficient for the operation of the IMAGE program.

The first list ($TYPE32) is used for specifying general data pertaining orientation, translation and scaling of the geometry. It also provides for setting flags pertaining to printing and geometry file control.

The $TYPE32 data set is followed by element geometry data if (described below) the INPUT file is specified in the above data set. Alternately the data may be read from the internal unit number 8. The geometry data is read, translated and scaled according to user instructions, then placed on a scratch unit (TAPE3) in binary format for use in the remainder of the program.

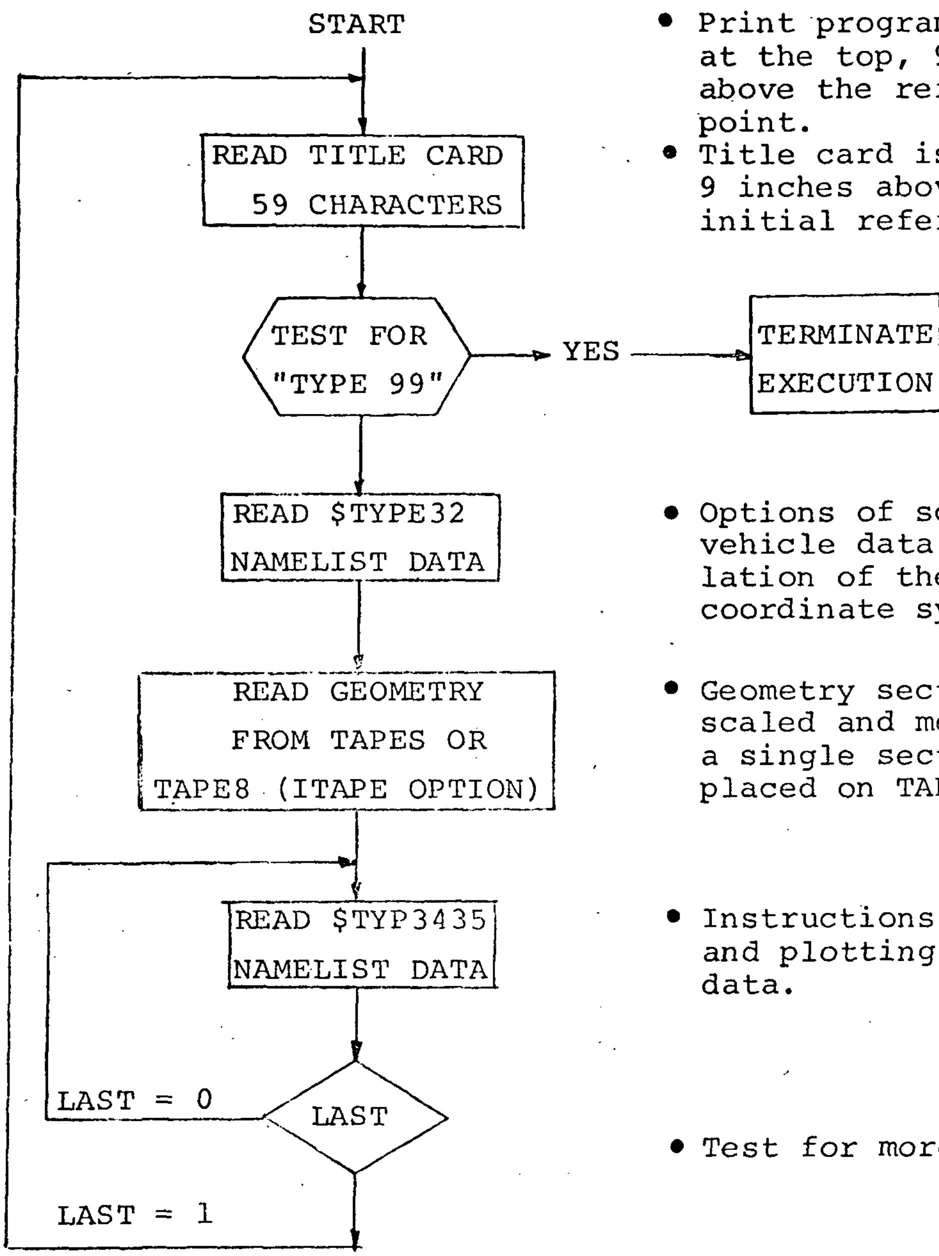
After geometry data (if any) the second NAMELIST ($TYPE3435) is input. This data set provides input for picture control options, etc., required to generate the desired pictures. Figure 4 illustrates an input stream to IMAGE. Any number of views may be specified by repeating the $TYP3435 data set. The last data set should specify LAST = 1.

The parameter, LAST, terminates the picture sequence for the current geometry which was temporarily placed on TAPE 3. The program logic returns to read a new TITLE card. The input flow is illustrated in figure 5. If a TITLE card is present in the input stream, additional geometry will be read from TAPE5 or TAPE8 and placed on the temporary storage file, TAPE3. A new sequence of pictures will be expected after the geometry is read. The execution is terminated by placement of a special (type 99) card in the input stream in place of a TITLE card. The special card must contain the integer, 99 in columns 71 and 72. The following paragraphs describe each input type in detail.

Title Card. - The TITLE card must be the first card in the input sequence. This card may contain from 1 to 59 characters (columns) of information which will be printed at the top of the frame, 9-inches above and 3-inches to the right of the initial reference point. A TITLE card must be present for each set of geometry which is to be processed by the program. Failure to supply this card will result in an input error. If no title is desired, a blank card must be inserted. The characters (99) in columns 71 and 72 of the TITLE card will cause normal termination of the program.

Program Controls. - The $TYPE32 data set is a NAMELIST input set consisting of input instructions for the geometry data, scaling, translation and orientation of the data, and printing instructions for the quadrilateral element characteristics. Figure 6 summarized the NAMELIST names and descriptions for the input set.

This section discusses each NAMELIST input in detail. Each paragraph is headed by the name and default value for the



- Print program header at the top, 9.5 inches above the reference point.
- Title card is printed 9 inches above the initial reference point.
- Options of scaling the vehicle data and translation of the vehicle coordinate system.
- Geometry sections are scaled and merged into a single section and placed on TAPE3.
- Instructions for scaling and plotting the picture data.
- Test for more geometry.

FIGURE 5 ILLUSTRATION OF INPUT FLOW LOGIC FOR IMAGE.

NAMELIST NAME	DEFAULT VALUE	DESCRIPTION
DELX	1.0	X-translation of scaled input data.
DELY	1.0	Y-translation of scaled input data.
DELZ	1.0	Z-translation of scaled input data.
IORIEN	0	Integer definition of a element orientation. = 0 Cross section input mode. = 1 Streamwise input mode. = 2 or 3 See text.
IREW8	0	Control integer for logical unit 8. file position. = 0 Rewind tape 8 before reading. = 1 Do not rewind 8 before reading.
ISTAT3	1	Number of vehicle components.
ITAPE	0	Geometry control integer. = 0 Geometry from unit 5. = 1 Geometry from unit 8 (coded). = 2 Geometry from unit 8 (binary).
PRINTS	0	Print flag. = 0 No printing. = 1 Print quadrilateral data.
XSC	0.0	X-scale factor.
YSC	0.0	Y-scale factor.
ZSC	0.0	Z-scale factor.

FIGURE 6 $TYPE32 NAMELIST INPUT SET.

variable listed. The heading is followed by a description of the input variable. If the default value is acceptable, the user need not define a value for it in the $TYPE32 NAMELIST data set.

PRINTS = 0 PRINTS is an integer variable which may have meaningful values of zero (0) or one (1). The variable controls the printing of detailed quadrilateral element characteristics.

0 Element characteristics will not be printed.

1 Element characteristics will be printed.

IORIEN = 0 IORIEN is an integer variable which defines the element orientation for the data. All vehicle components must have the same orientation.

0 Normal mode using cross sections.

1 Geometry is input in streamwise strips.

2 Geometry is input in streamwise strips. For each strip of elements, the first coordinate point in the right-hand strip of points is not used in the formation of the leading edge element but is ignored by the program.

3 Same as = 2 except the left-hand point is ignored in the formation of the leading edge elements.

Usually the data is described with IORIEN values of 0 or 1. Values of 2 or 3 will give correct pictures when the data-point slip methods of reference 1 are used to input the geometry data. Vehicle components with different values for IORIEN cannot be drawn correctly on a single picture frame.

The scale factors and associated translation increments described below are generally used to transform the configuration geometry to a more convenient form for plotting. The factors are frequently used to move the vehicle reference axis so that the vehicle center corresponds approximately to the coordinate system origin. For a vehicle with its nose at X = 0.0 this is accomplished by using a DELX value (see below) of about one-half of the vehicle length. This simplifies the selection of picture scales to keep the vehicle within the imaginary picture frame.

The original geometry data on tape 5 or tape 8 is not changed by the use of the scale factors. They are applied to the geometry stored on the temporary file TAPE3. The data is transformed using the following equations for X, Y and Z.

$$X_{new} = X_{input} \cdot (XSC) + DELX$$

$$Y_{new} = Y_{input} \cdot (YSC) + DELY$$

$$Z_{new} = Z_{input} \cdot (ZSC) + DELZ$$

The above scaling terms are defined as follows:

XSC = 1.0 Scale factor to be multiplied by X_{input}.

YSC = 1.0 Scale factor to be multiplied by Y_{input}.

ZSC = 1.0 Scale factor to be multiplied by Z_{input}.

DELX = 0.0 X increment to be added to $X_{input} \cdot (XSC)$.

DELY = 0.0 Y increment to be added to $Y_{input} \cdot (YSC)$.

DELZ = 0.0 Z increment to be added to $Z_{input} \cdot (ZSC)$.

Usually the configuration geometry consists of more than one component. In the data format of the IMAGE program, the data for each component is terminated with an integer flag referred to as a status flag as described under Surface Model Data (below). The merging of the geometric components is desirable for plotting purposes. Therefore, the geometric components are merged into a single component as the geometric data is being transferred to the temporary file, TAPE3. The user of the program must specify the number of components to be merged for each picture sequence. This specification is accomplished by the integer input variable ISTAT3.

ISTAT3 = 1 Integer variable number of vehicle components with Status = 3 in the vehicle geometry which the user wishes to plot as a unit. The program will count the number of Status = 3 in the geometry deck and when the count reaches this input value, the program will proceed to the plot options for the vehicle components which have just been read.

Several options are available to the user of the program for accessing geometric data. Alternate files may be employed and alternate formats may be specified. The input parameter for controlling the above options is the integer variable ITAPE.

ITAPE = 0 Geometry tape control integer variables with the following possible values:

= 0 Geometry data (type 3) will be read from Tape 5 (geometry data cards are loaded along with picture-data control cards).

= 1 Geometry data (Type 3 will be read from the geometry storage tape (Tape 8) in coded format.

= 2 Geometry data (Type 3) will be read from the geometry storage tape (Tape 8) in binary format.

The IMAGE program provides a flexible means of controlling the alternate geometry file, TAPE8. Multiple components may be stored on TAPE8. These components may be extracted in sequential groups or plotted individually. The input parameter for rewinding the geometry tape (TAPE8) is IREW8. Usually the file is rewound for the first sequence of pictures, then through the use of the input variable ISTAT3, additional groups of components can be extracted for plotting purposes.

IREW8 = 0 Integer variable to control the position of Tape 8 just before the geometry data are read from it.

= 0 Rewind Tape 8 and then read geometry data from it.

= 1 Do not rewind Tape 8, but start reading geometry data from it in its current position.

The integer, IREW8, permits the user to store more than one group of vehicle geometrics on the geometry tape, then collect and plot them by groups.

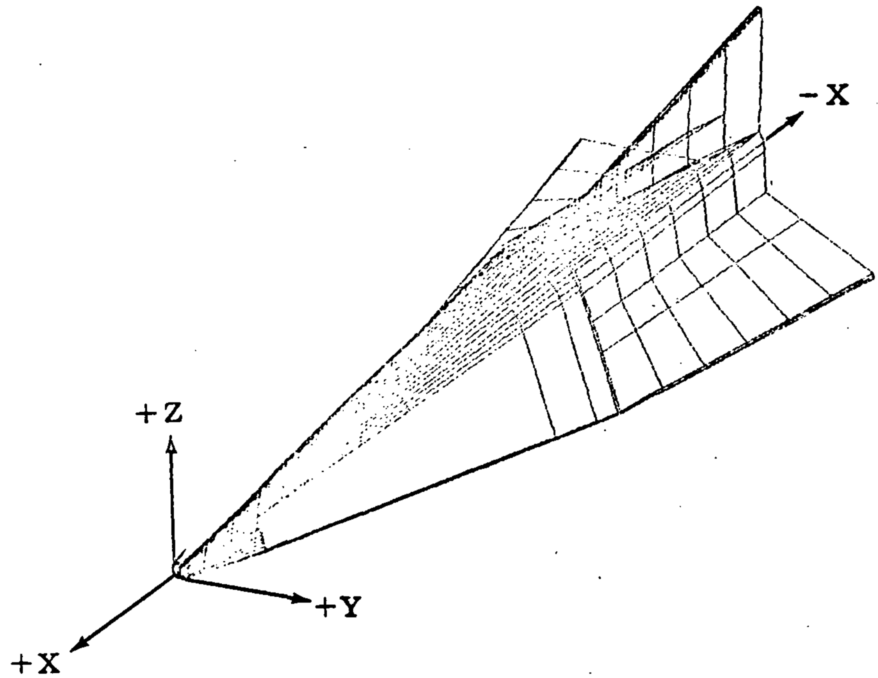
Surface Model Data. - The program accepts the geometry in element data form only. No simplified geometry techniques are employed. Several auxiliary programs are available which convert simplified geometry to element data format suitable for input to the IMAGE program. Reference 1 is an example of such a program. It generates element data using several mathematical surface generation options and includes an aircraft geometry option which provides a convenient means of generating aerodynamic surfaces.

The element data in IMAGE may be read from input or from an alternate unit. If the INPUT file is selected, the actual data cards are merged with the other input data as illustrated in figures 2 and 4.

If the alternate input unit (logical unit 8) is selected for element data input, the data may be read in coded (same as INPUT cards) or binary (fast read) mode. In order to use the binary mode, the data must have been written in binary mode. Reference 1 generates element data in coded format. The binary mode is very useful for improving efficiency particularily when the geometry must be regenerated many times for a problem solution. Regeneration of geometry is often desirable when studying the effects of geometric shapes. Regeneration will be required when optimizing geometric shape in an ODIN simulation.

Element Data Format. - The element data method uses a large number of surface coordinate points on the surface of the configuration. The points can be ordered around a station contour or in a streamwise manner. Each point consists of an X, Y, Z coordinate set and a status flag. Each card contains two points.

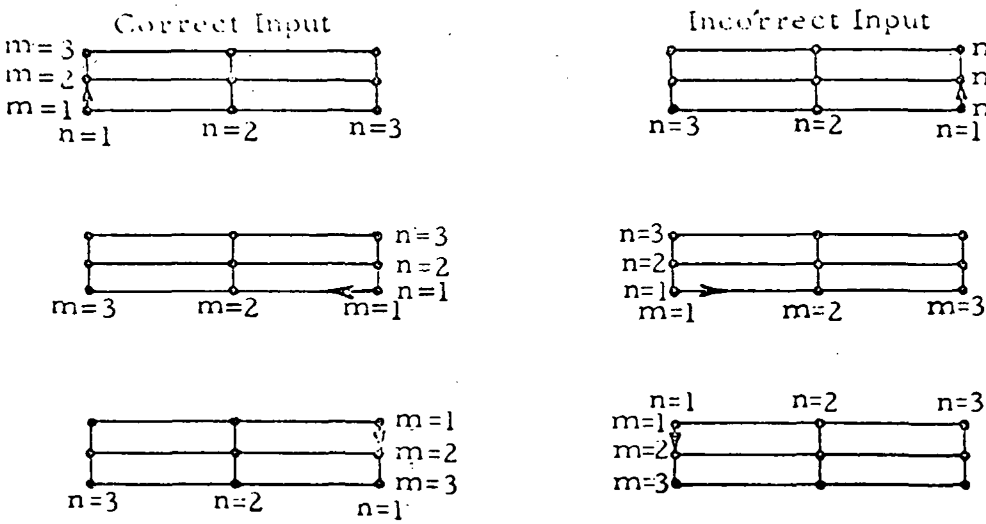
The coordinate system used for all the geometry data is shown in the figure below. For symmetrical vehicles it is standard practice to input the left side of the vehicle only. There is only one input method for the corner-point geometry and since any of the auxiliary geometry programs finally produce geometry data in surface-element form, it is important that the methods and nomenclature used with this method be clearly understood. It is, therefore, recommended that the input instructions for the surface-element method be studied before an attempt is made to use the method or write a geometry generation routine.



The geometric input data in this method include the coordinates of a large number of points on the vehicle surface. The input data are organized in a manner that permits the description of a vehicle on a component buildup basis. This gives increased flexibility in shape description and makes it possible to draw exploded views by physically separating the components of the vehicle. Because of possible changes in the surface contours of a configuration, or replacement of the entire component, it may also be desirable to divide the configuration into several components. This permits easy changes either in a manual or automated mode such as ODIN. Each component of a configuration is further divided into a number of sections each defined by a group of points in space. In practice, the surface coordinates are usually recorded from cross-section drawings of the vehicle in such a way that each point need be read only once (even though it may be a member of as many as four adjacent quadrilateral elements). Each point is defined by its three coordinates and a status flag that indicates whether it is the first point of a new section, a continuation of a group of points defining a section, the beginning of a new section, or the last point of the component. The program uses the status flags to determine how the input points are to be related to form the quadrilateral elements, and how the elements are combined to form a component.

The first question that the user asks when starting to load the element geometry is the order in which the surface points are entered. The basic rules to be followed are given below. The rules discussion will be followed by a discussion of a visual technique that many users will find helpful in determining the proper loading order.

For the purpose of organizing the input data for computation, each point is assigned a pair of integers, m and n. These integers are not actually input to the program (they are calculated internally) but their use in the following discussion will provide a better understanding of the input data organization. For each point, n identifies the "column" of points to which it belongs, and m identifies its position in the "column," i.e., the "row." The first point of a "column" always has m = 1. To insure that the program will compute outward normal vectors, the following condition for the order to input points must be satisfied. If an observer is located in the outside the component and is oriented so that locally he sees points on the surface with m values increasing upward, he must also see n values increasing toward the right. Strict adherence to this simple rule will always lead to a correct set of input geometry data. Examples of correct and incorrect input are shown in the sketches below. In these pictures the exterior of the configuration lies above the paper, and the interior of the configuration lies below the paper. The arrows indicate the order of reading the points.



Associated with each input point is an input quantity called its status. The first point of each new section has Status = 2. Except for the first n-line of a component, the first point of each n-line has Status 1. The last point of the component of the vehicle has Status 3. All other points have

Status = 0 (i.e., they may be left blank on the input sheet). The IMAGE program plots the picture according to components ending with a Status = 3.

The simple visual technique described below is helpful in determining the proper order of the input points:

1. First, assume that you are holding in your hand a small model of the vehicle shape. Many program users find it helpful to construct a small paper model to help in visualizing the geometry loading procedure. On this model draw lines to represent the elements to be loaded for a given vehicle section.

2. Next, decide which strips of elements are to constitute "columns" and which "rows." In most problems one of two procedures is selected - either a "column" of elements starts at the bottom of the shape and continues around to the top, roughly following vehicle cross-section lines, or a "column" is oriented so that it starts at the front part of the vehicle and runs aft toward the rear.

3. Hold the model out in front of you and rotate it until the columns are vertical with the first row of elements at the bottom. This procedure should be used regardless of what part of the vehicle is being loaded - the body, fin, inside of fin, etc. Always orientate the model so that you are looking at the section to be loaded (from the outside, looking at the surface) with the columns running vertical, and the rows running horizontal.

4. Now that you have the section being loaded oriented in front of you, with the columns vertical, apply the following cardinal geometry rule:

   > If a column of data points are loaded from the bottom to the top, then the next column of points (starting with a Status = 1) must be to the right.

All of the geometry input data for this geometry option are input on "Type 3" element data cards (an integer 3 in column 72). Each card contains the X, Y, Z coordinates and status flag for two points on the body surface. Every card in the element geometry deck must contain two surface points except the last card, which may have only the first surface point coordinates and status filled in. If a particular line of vehicle points is odd in number, then it is usually advisable to repeat the last point (a dummy point) so that the last card will have two sets of point data. This permits the shifting of configuration components without disrupting other components. A description of the input card for element data is shown in figure 7.

Picture Specifications. - The picture control data set is a NAMELIST input set called $TYP3435 which is summarized in figure 8. It controls the drawing of pictures and the printing of text. A typical deck setup will consist of several data sets of this type, one for each picture desired. The program will always try to read one data set. Additional views or text information may be generated or added by the inclusion of additional $TYP3435 data sets. The input integer controlling this function is LAST.

LAST = 0 Integer variable controlling the program flow logic after drawing a picture or printing text.

= 0 Return for $TYP3435 data set.

= 1 Return for new TITLE card (may result in program termination or the reading of more geometric data).

If the variable LAST is set to one (1), the reference axes system of the plotting device is moved to a new frame position specified by the user. The input variables controlling the axis translation are XMOVE and YMOVE.

XMOVE = 17.0 The translation of the reference axis system in the X-direction (in inches) following the last $TYP3435 data set.

YMOVE = 0.0 The translation of the reference axis system in the Y-direction (in inches) following the last $TYP3435 data set.

The values of XMOVE and YMOVE are made with respect to the original coordinate system reference specified upon execution of the program. The picture positioning parameters DXG and DYG described below have no affect on the XMOVE, YMOVE translation.

The program permits the user to specify the viewing angles with the three-axis system discussed in the early part of this report. The righthand rule is used for identifying the positive rotation angle of the vehicle geometry. The sequence of rotation is yaw, pitch and roll.

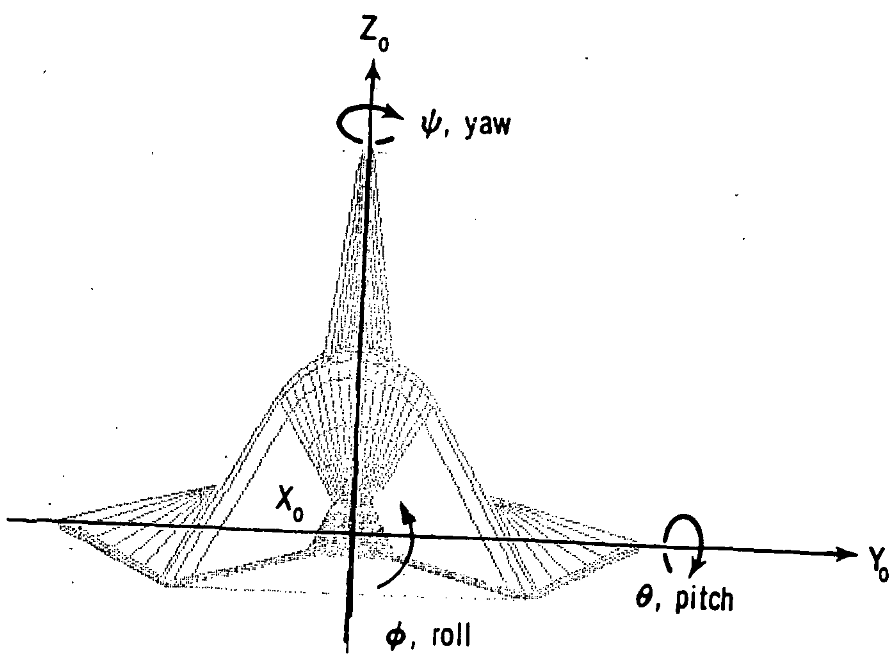
Column	Code	Explanation
1-10	X	X-coordinate of surface point (the value of X is written anywhere in this space with a decimal point and sign; usually input only if it is negative).
11-20	Y	Y-coordinate of surface point.
21-30	Z	Z-coordinate of surface point.
31	STAT	Status flag for the above set of coordinates (=2, 1, 0, or 3).
32-41	XX	X-coordinate of surface point.
42-51	YY	Y-coordinate of surface point.
52-61	ZZ	Z-coordinate of surface point.
62	STATT	Status flag for the above set of coordinates (=2, 1, 0, or 3).
66-68	CASE	Case number (right-justified integer)
69-70	SECT	Numbers or letters to identify the vehicle section. These must be legal machine characters.
72	TYPE	Card type number = 3.

FIGURE 7 ELEMENT DATA INPUT CARDS

NAMELIST NAME	DEFAULT VALUE	DESCRIPTION
DXG	0.0	Repositioning of X- and Y- reference point before plotting current picture - inches.
DYG	0.0	
HTEXT	0.14	Character height for text material - in.
IAREA	0.	Print control integer. = 0 No print. = 1 Print area of each section.
ICS	0.	Connectivity flag for quadrilaterals. = 0 Connect all four points. = 1 Connect points 1-2 and 3-4. = 2 Connect points 1-4 and 2-3. = 4 Do not connect points.
IQUAD	0.	Controls points to be drawn. = 0 Draw input points. = 1 Draw computed points on quadrilateral.
IREFL	1.	Reflected element flag. = 0 Draw input elements only. = 1 Draw reflected elements. = 2 Draw reflected elements (only one quadrant is input).
ISHAD	0.	Hidden line option. = 0 Do not plot hidden lines. = 1 Plot all lines.
LAST	0.	Program control flag. = 0 Return for $TYP3435 data set. = 1 Return for TITLE card (or Type 99 to terminate).

FIGURE 8A $TYP3435 NAMELIST INPUT SET.

NAMELIST NAME	DEFAULT VALUE	DESCRIPTION
PHI	0.	Roll angle (see below) - degrees.
PSI	0.	Yaw angle (see below) - degrees.
THETA	0.	Pitch angle (see below) - degrees.



SCAL	6.0	Frame size in inches. Geometry will be scaled to this dimension.
TEXT	F	Logical control variable for text input. = .TRUE. Text information will follow $TYP3435. = .FALSE. No text information will be input.

FIGURE 8B $TYP3435 NAMELIST INPUT SET. (CONTINUED)

NAMELIST NAME	DEFAULT VALUE	DESCRIPTION
XLG	*	Extreme forward X-dimension on input geometry - input units.
XRG	*	Extreme aft X-dimension on input geometry - input units.
YBG	*	Extreme bottom Y-dimension on input geometry - input units.
YTG	*	Extreme top Y-dimension on input geometry - input units.
XMOVE	17.0	Translation of X-reference after one complete case - inches.
YMOVE	0.0	Translation of Y-reference after one complete case - inches.

*Computed before. This input set is read but may be overridden (see text).

FIGURE 8C $TYP3435 NAMELIST INPUT SET (CONTINUED).

PSI = 0. Yaw angle in degrees measured with respect to the positive Z-axis of the reference geometry coordinate system.

THETA = 0. Pitch angle in degrees measured with respect to the positive Y-axis of the reference geometry coordinate system.

PHI = 0. Roll angle in degrees measured with respect to the positive X-axis of the reference geometry coordinate system. Note that the positive X-axis is usually toward the nose of the vehicle.

The technique used for plotting geometric data is to treat each quadrilateral element as a five-point 'curve' in the image plane. The IMAGE program provides the user with the option of specifying the points on each quadrilateral element which will be connected when plotting the elements. The integer variable which controls this option is ICS.

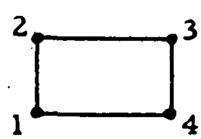
ICS = 0 Integer variable controlling the points to be connected.

= 0 Connect all 4-points of each element.

= 1 Connect points 1-2 and 3-4 (see diagram below)

= 2 Connect points 1-4 and 2-3.

= 3 Do not connect points with lines



Quadrilateral Element

Usually the geometric data plotted by the IMAGE program is symmetrical about the X-Z plane and data is generally provided for only one side of the configuration. The program provides the user the option of plotting only the input geometry, the reflected geometry or alternately plotting both the input geometry and the geometric reflection of the input geometry. The input integer controlling this option is IREFL.

IREFL = 1 Integer variable controlling the plotting of reflection-elements as follows:

= 0 Do not plot elements reflected to negative side of Y-axis.

= 1 Plot reflected elements.

= 2 Plot reflected elements (only one quadrant is input).

The plotting of all quadrilateral elements can often be distracting if not misleading with respect to appearance of the vehicle which the geometric data represents. The distraction is usually caused by the plotting of elements which face away from the viewer. The IMAGE program provides the user with the option of eliminating the above class of elements for the individual geometry components. The integer variable which controls this option is ISHAD. IMAGE does not provide any capability for omitting elements which face the viewer but are hidden by another component of the vehicle.

ISHAD = 0 Integer variable controlling the plotting of elements facing away from the viewer.

= 0 Do not plot elements that face away from the viewer (shadow elements).

= 1 Plot shadow elements (elements facing away from viewer).

The IMAGE program provides the user the option of printing the surface area characteristics of the vehicle components as follows:

IAREA = 0 Integer variable controlling the printing of section areas.

= 0 Do not print section areas.

= 1 Print out the area of each section.

Usually, the desired picture is represented by the actual corner points of the geometric data. However, for some types of analysis, it is desirable to view the corner points of the actual quadrilateral elements since the latter represents the data actually used by some of the technology programs which the IMAGE geometry supports. The IQUAD option permits the selection of the points to be drawn.

IQUAD = 0 Integer variable which selects the corner points to be drawn.

= 0 Draw input elements.

= 1 Draw picture using quadrilateral element corner points.

The program provides a flexible means of framing the picture data. Because of the flexibility provided, the user should exercise a certain degree of caution in setting up the data to assure the resulting picture will be plotted in the desired location. The following discussion of input variables is designed to help the unfamiliar user in setting up the data for positioning the picture sequences. Figure 9 shows the relationships among the variable input data discussed below.

SCAL = 6. Imaginary frame size in inches for the current picture. All data will be scaled to fit within the specified frame size according to the following relationships.

X-scale factor = (XRG-XLG)/SCAL

Starting X-value = XLG

Y-scale factor = (XRG-XLG)/SCAL

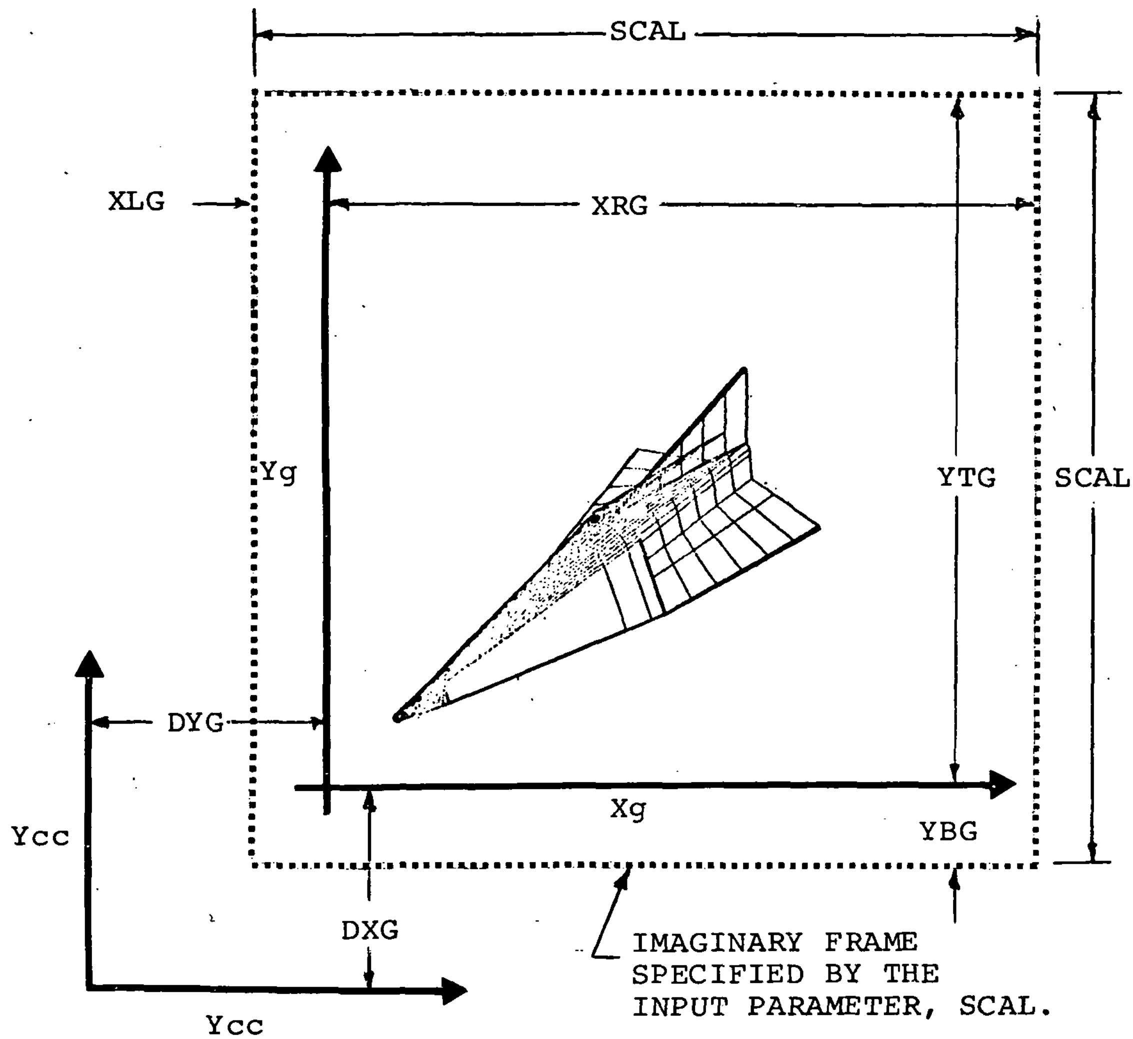
Starting Y-value = XLG

XRG and XLG are computed automatically by the IMAGE program. The coordinate axes used are the result of the initial geometric transformation described above. The relationships used for computing the above parameters internally are as follows:

XLG = min (all X-coordinates)

XRG = max (all X-coordinates)

The above equation generates a "square frame" scaled to the longitudinal configuration geometry dimensions. The "square frame" produces identical scale factors in the X and Y directions. The geometry is guaranteed to fit in the specified frame size. However, the data will not necessarily fit in the frame itself. The position of the picture depends upon the values of the viewing angles. For example, assume the geometry reference is at the nose of the vehicle. The imaginary frame under automatic scaling conditions would encompass the length of the vehicle. Further, the data would all be scaled to the value of the input variable, SCAL. For the above conditions, the position of the nose is controlled by the input variables DXG and DYG.



COORDINATE SYSTEM DEFINITIONS

Xcc, Ycc — Reference coordinates of the plotting device before the current picture is drawn. The initial values are usually specified by the user on the plot request card. Before the picture is drawn they are moved to the Xg, Yg system.

Xg, Yg — Reference axis of the geometric data after the initial transformation specified by the input parameters DELX, DELY, DELZ, XSC, YSC and ZSC.

FIGURE 9 ILLUSTRATION OF THE FRAMING TECHNIQUE EMPLOYED IN THE IMAGE PROGRAM.

Each picture is positioned within the limits of the plotting device using the variables DXG and DYG measured with respect to the previous position.

DXG = 0.	Repositioning of the reference Y-axis in inches for the plotting device before plotting the geometry data for the current picture.
DYG = 0	Repositioning of the reference X-axis in inches for the plotting device before plotting the geometry data for the current picture.

Once the reference axes for the current picture is described, the imaginary frame is defined by the variables YLG, XRG, YBG, YTG. Although computed automatically as described above, the values of XLG, YRG, YBG and YTG may be specified by the user in scaled geometry coordinates with respect to the translated vehicle geometry on TAPE3 as follows:

XLG = computed	Value of the left side of the imaginary frame (geometry scale).
XRG = computed	Value of the right side of the imaginary frame (geometry scale).
YBG = computed	Value of the bottom of the imaginary frame (geometry scale).
YTG = computed	Value of the top of the imaginary frame (geometry scale).

When specifying the above values, the user should remember the "square frame" is essential to the creation of an undistorted picture. Rectangular frames will produce pictures which appear for-shortened in the direction of the short side of the rectangle. For example:

XLG = -10., XRG = 20.,

YBG = -5., YTG = 25.,

would produce a square frame 30 inches on a side. In the above example, the data for the current picture would be scaled as follows:

Y - scale factor = (20.-(-10.))/10. = 3 units/inch

Y - scale factor = (25.-(5.))/10. = 3 units/inch

Starting X-value = -10.

Starting Y-value = -5.

The primary function of the IMAGE program is the generation of pictorial data. Equally important to a geometric description is a description of the vehicle characteristics. The IMAGE program provides the user the option of displaying text information along with the pictorial data. The logical variable controlling this option is TEXT.

TEXT = .FALSE.	Logical variable, if .TRUE., Text information will be read from cards following the $TYP3435 data set in free field format.

In the text option the input variables DXG and DYG position the reference coordinates at the beginning of the first line of text. This reference point will remain until altered by a new $TYP3435 data set. Any number of cards may be read as text. Each card represents a line of text. Lines of text may be skipped by placing a zero (0) in column 1 of the text card. This provides a convenient means of spacing the text information. The text input is terminated by placing the character (2) in column 1 of the last text card. The last card will not be printed.

The maximum number of cards (lines) of text is 53 cards. The character height of the text is controlled by the input variable HTEXT.

HTEXT = 0.14	Height of the characters in the text material.

Post Processing Instructions

The IMAGE program is designed to generate a file of plot commands for a 12-inch vertical height continuous roll paper such as CALCOMP. The file must be on a physical tape for plotting (see figure 3). The user establishes a reference point on the roll at the time the plot request is submitted to be plotted. Generally a one inch offset from the X-axis (Y=1) is adequate. The Y-offset is not applicable since a continuous roll of paper is generally used. A hard coded program header is printed 9.5 inches above the user established reference. All pictures should be scaled and positioned within the 9.5 inch limitation on vertical height, otherwise, the picture may overlay the program header. The horizontal placement of pictures is essentially unlimited.

Externally Generated Geometry

The IMAGE program provides the flexibility of reading the element data for plotting from the normal input device (INPUT) or from an alternate unit. The later capability is provided for the purpose of plotting geometry generated by another computer program.

The information may be stored temporarily or permanently on a system file and attached to the IMAGE execution job step by a method called file substitution.

File substitution is a CDC 6600 system capability which provides a correspondence between internal (logical units) files and external (system) files. The mechanism by which this correspondence is implemented is the "program card." The program card for the IMAGE program is:

```
PROGRAM IMAGE (INPUT, OUTPUT, TAPE5 = INPUT, TAPE6 =
               OUTPUT, TAPE3, TAPE 8)
```

In the above illustration the file, TAPE8 is the internal logical unit which may contain the geometric data to be plotted. The correspondence between the internal logical units, TAPE8 and the external file is established by the substitution of the TAPE8 parameter at execution time. For example, assume the data to be plotted was stored on an external file called DATA. The execution card for the IMAGE program would be:

```
EXECUTE (IMAGE,,,,,,,DATA)
```

During the above execution, the IMAGE program would read from the file called DATA each time the logical unit, TAPE 8, was read. The program card parameters are <u>positional</u>. Therefore, the six commas are essential for the proper use of file substitution, one for each of the other files on the program card.

Use of IMAGE within ODIN

The Optimal Design Integration (ODIN) system is a library of independent computer programs representing the analytical capabilities in a wide variety of technological disciplines. The IMAGE computer program is but a single member of the ODIN library. The sequence of execution of the individual programs is controlled by the executive program, DIALOG (reference 2) which also maintains a name-oriented data base of design information.

Each piece of information is stored by name. The data base forms a communication link among the programs in the library. When used within the ODIN system, IMAGE receives data from the data base before execution. Generally, the ODIN library programs provided information to be stored in the data base. However, IMAGE does not currently generate information of this category.

The actual transfer of information from the data base to IMAGE is performed by DIALOG through pre-processing of the data so the program is "unaware" that it is part of an analysis involving many programs. There are no special input requirements for using IMAGE within the ODIN system. A single control directive,

```
'EXECUTE IMAGE'
```

is required for the execution of the program. The delimiter (') is a 4-8 punch. The data which follows this directive is the normal input data described above. However, any data values may come from the data base by specifying the data base name on the input card (in lieu of the actual value).

```
$TYPE32
  :
  :
SCAL = 'SCALE',
  :
  :
$END
```

In the above illustration, the name SCALE is a data base name which may represent a scale factor for the entire configuration. The executive program, DIALOG, replaces the name and the associated delimiter 'SCALE' with the data base value for SCALE. Upon execution of IMAGE, the input component is photometrically scaled by the current scale factor in the data base. A procedure is also available for transferring data base arrays (see reference 2).

The illustration above applies to namelist input, but the procedure for extracting data base information is equally applicable to formatted input. The field width is specified to the position of the delimiters (') as illustrated below:

```
2579    3.426    'DEN'    7.29    - - -
```

DEN is assumed to be the name of a data base variable. The value of DEN will be placed (by DIALOG) on the input card in the most significant (E and F) format left justified in the specified field. If the data base variables were an integer, the value would be right justified in the field (I format).

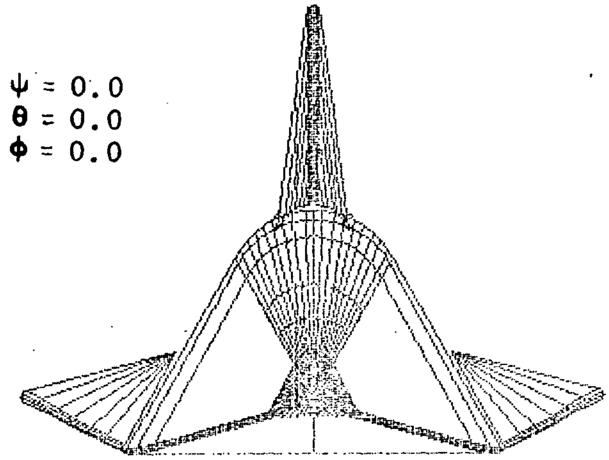
PROGRAM OUTPUT

The output from the IMAGE program is both printed output and plotted output. Geometric characteristics of the quadrilateral elements are available at the user's option. This data is presented through the normal output channels. The plotted data is in the form of a plot tape. The actual plots are obtained by separate submission of a plot request card. Upon submission of the request, the plot tape is processed on the CALCOMP plotting hardware. Pictures of the vehicle at pre-selected viewing angles, as illustrated in figure 10, may be generated on the CALCOMP device. Input errors can then be corrected before the data is submitted to another ODIN program for the aerodynamic or other technology calculations. Alternately, annotated report quality pictures (see figure 1) may be generated by combining the picture and text options in the program.

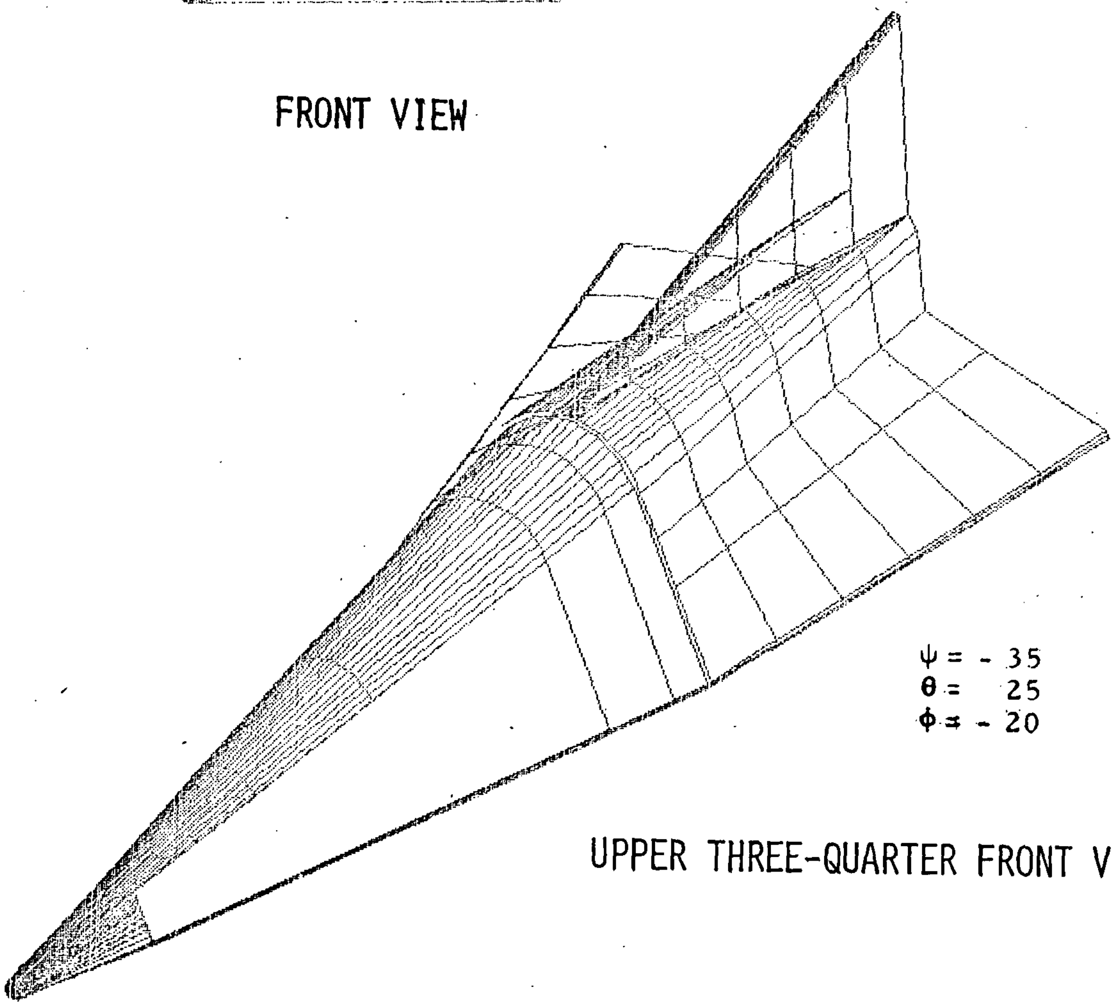
The IMAGE program can also be used to obtain a detailed print-out of the properties of each quadrilateral element of the vehicle as illustrated in figure 11. The normal output may also include the accumulated surface area as illustrated in figure 12 and the number of elements for each section of the vehicle. If no print options are specified, only the picture number is printed.

ODIN Output

The current IMAGE program generates no output for the ODIN data base. It has been used primarily for displaying geometric data to be used by other technology programs.

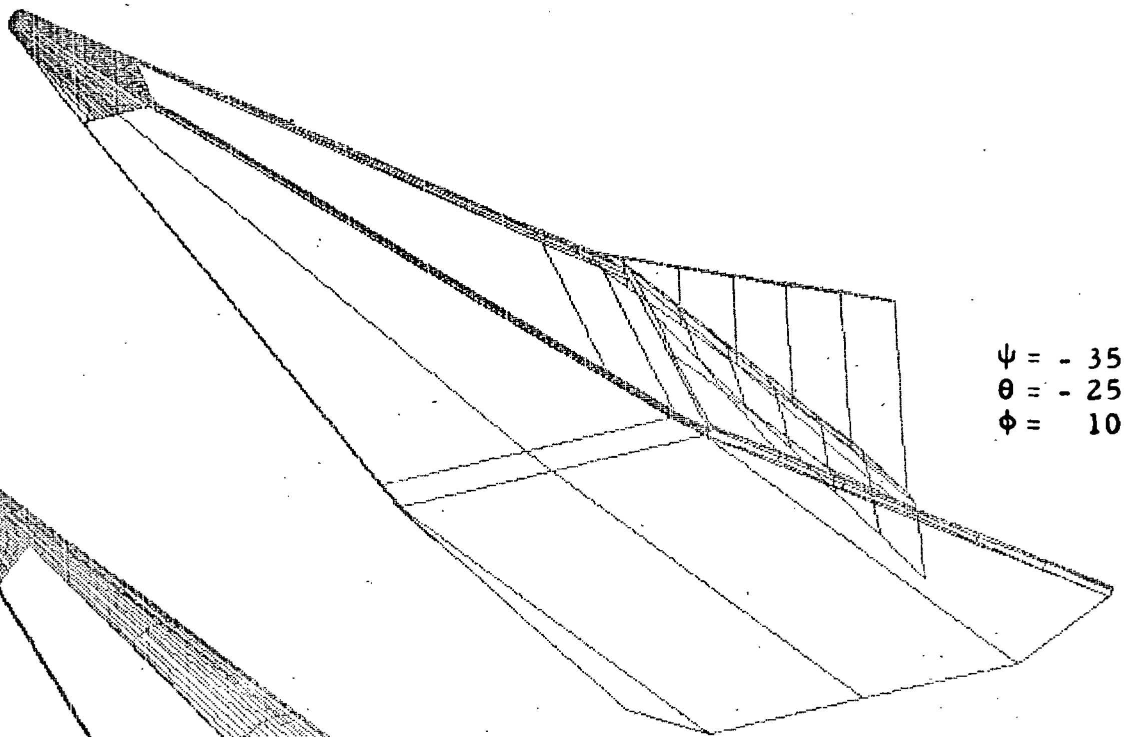


FRONT VIEW

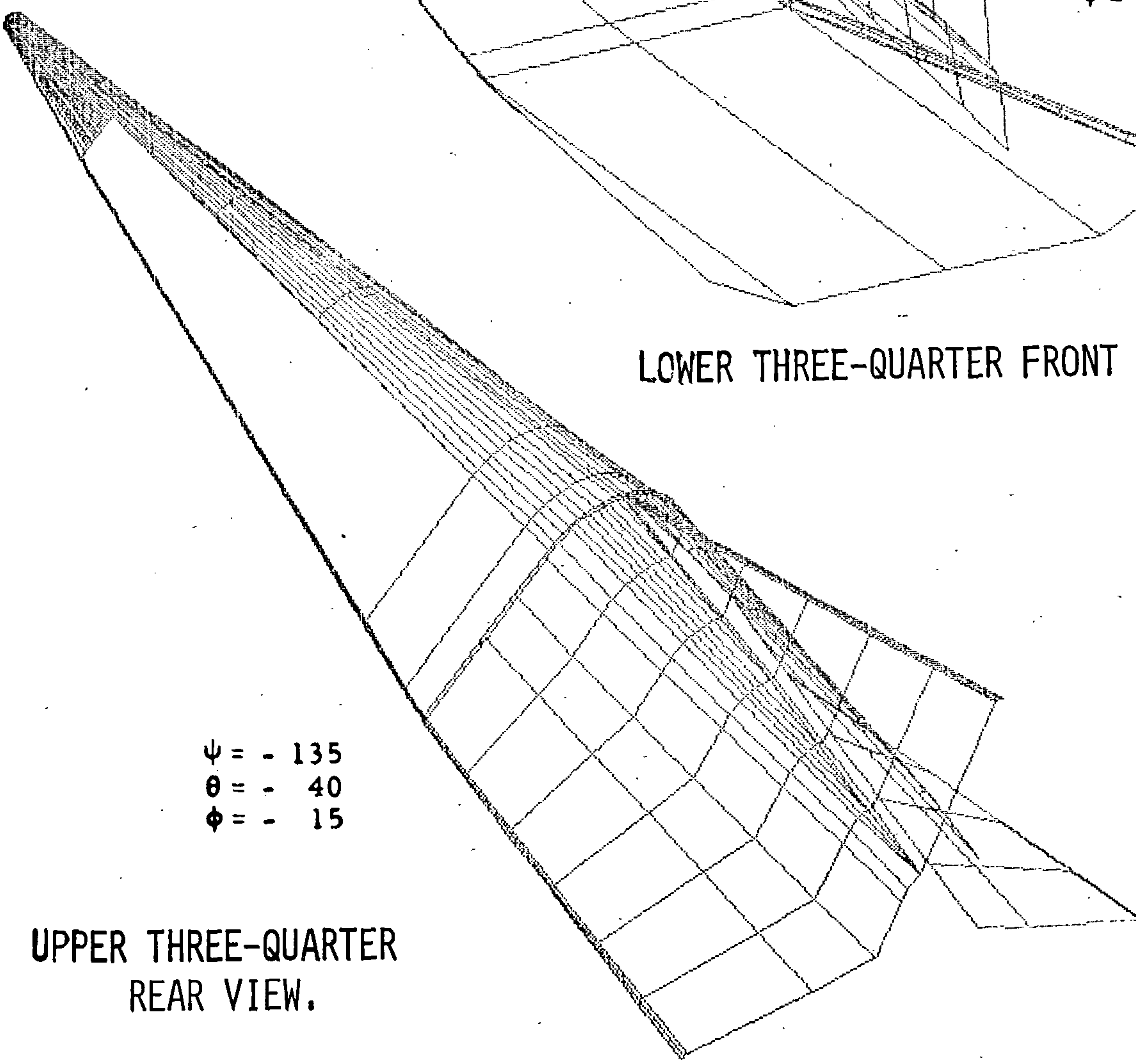


UPPER THREE-QUARTER FRONT VIEW

FIGURE 10A PICTURE OUTPUT FROM IMAGE.



LOWER THREE-QUARTER FRONT VIEW.



UPPER THREE-QUARTER REAR VIEW.

FIGURE 10B PICTURE OUTPUT FROM IMAGE. (CONTINUED)

QUADRILATERAL CHARACTERISTICS - PICTURE DRAWING PROGRAM

CASE 0 PAGE 0

INPUT SURFACE ELEMENT DATA

N	M	X Y Z	X Y Z	X Y Z	X Y Z	NX NY NZ	XCENT YCENT ZCENT	AREA DELTA V VOLUME
1	1	0.	0.	-83.00000E-04	-83.00000E-04	.992090	-55.33333E-04	76.52933E-05
		0.	0.	22.80000E-03	0.	.021691	76.00000E-04	12.61600E-08
		0.	0.	-62.60000E-03	-66.60000E-03	-.123639	-43.06667E-03	12.61600E-09
	2	0.	0.	-83.00000E-04	-83.00000E-04	.992088	-55.33333E-04	76.42868E-05
		0.	0.	42.80000E-03	22.80000E-03	.062987	21.86667E-03	10.52661E-07
		0.	0.	-51.00000E-03	-62.60000E-03	-.108598	-37.86667E-03	11.78821E-07
	3	0.	0.	-83.00000E-04	-83.00000E-04	.992088	-55.33333E-04	76.47813E-05
		0.	0.	57.70000E-03	42.80000E-03	.096047	33.50000E-03	24.60742E-07
		0.	0.	-33.30000E-03	-51.00000E-03	-.080853	-28.10000E-03	36.39564E-07
	4	0.	0.	-83.00000E-04	-83.00000E-04	.992093	-55.33333E-04	75.36178E-05
		0.	0.	65.60000E-03	57.70000E-03	.117932	41.10000E-03	37.01260E-07
		0.	0.	-11.60000E-03	-33.30000E-03	-.042934	-14.96667E-03	73.40824E-07
	5	0.	0.	-83.00000E-04	-83.00000E-04	.992091	-55.33333E-04	76.70267E-05
		0.	0.	65.60000E-03	65.60000E-03	.125524	43.73333E-03	42.10645E-07
		0.	0.	11.60000E-03	-11.60000E-03	0.000000	-13.87779E-18	11.55147E-06
	6	0.	0.	-83.00000E-04	-83.00000E-04	.992093	-55.33333E-04	76.36178E-05
		0.	0.	57.70000E-03	65.60000E-03	.117932	41.10000E-03	37.01260E-07
		0.	0.	33.30000E-03	11.60000E-03	.042934	14.96667E-03	15.25273E-06
	7	0.	0.	-83.00000E-04	-83.00000E-04	.992088	-55.33333E-04	76.47813E-05
		0.	0.	42.80000E-03	57.70000E-03	.096047	33.50000E-03	24.60742E-07
		0.	0.	51.00000E-03	33.30000E-03	.080853	28.10000E-03	17.71347E-06
	8	0.	0.	-83.00000E-04	-83.00000E-04	.992088	-55.33333E-04	76.42868E-05
		0.	0.	22.80000E-03	42.80000E-03	.062987	21.86667E-03	10.52661E-07
		0.	0.	62.60000E-03	51.00000E-03	.108598	37.86667E-03	18.76613E-06
	9	0.	0.	-83.00000E-04	-83.00000E-04	.992090	-55.33333E-04	76.52933E-05
		0.	0.	0.	22.80000E-03	.021691	76.00000E-04	12.61600E-08
		0.	0.	66.60000E-03	62.60000E-03	.123639	43.06667E-03	18.89229E-06
2	1	-83.00000E-04	-83.00000E-04	-33.30000E-03	-33.30000E-03	.917135	-22.07026E-03	25.88699E-04
		0.	22.80000E-03	42.80000E-03	0.	.068823	16.90834E-03	24.30574E-07
		-66.60000E-03	-62.60000E-03	-11.75000E-02	-12.50000E-02	-.392589	-95.80336E-03	21.32287E-06

FIGURE 11 ILLUSTRATION OF QUADRILATERAL ELEMENT PROPERTIES PRINTOUT FOR IMAGE.

TOTAL AREA OF INPUT ELEMENTS =	.1925	TOTAL NUMBER OF ELEMENTS =	45
TOTAL VOLUME OF INPUT ELEMENTS =	.016		
TOTAL AREA OF INPUT ELEMENTS =	2.1657	TOTAL NUMBER OF ELEMENTS =	54
TOTAL VOLUME OF INPUT ELEMENTS =	.093		
TOTAL AREA OF INPUT ELEMENTS =	4.9149	TOTAL NUMBER OF ELEMENTS =	69
TOTAL VOLUME OF INPUT ELEMENTS =	1.250		
TOTAL AREA OF INPUT ELEMENTS =	6.0621	TOTAL NUMBER OF ELEMENTS =	78
TOTAL VOLUME OF INPUT ELEMENTS =	1.348		
TOTAL AREA OF INPUT ELEMENTS =	41.3808	TOTAL NUMBER OF ELEMENTS =	79
TOTAL VOLUME OF INPUT ELEMENTS =	1.348		
TOTAL AREA OF INPUT ELEMENTS =	76.9257	TOTAL NUMBER OF ELEMENTS =	80
TOTAL VOLUME OF INPUT ELEMENTS =	52.711		
TOTAL AREA OF INPUT ELEMENTS =	89.7474	TOTAL NUMBER OF ELEMENTS =	82
TOTAL VOLUME OF INPUT ELEMENTS =	81.172		
TOTAL AREA OF INPUT ELEMENTS =	105.4111	TOTAL NUMBER OF ELEMENTS =	109
TOTAL VOLUME OF INPUT ELEMENTS =	86.445		
TOTAL AREA OF INPUT ELEMENTS =	111.0823	TOTAL NUMBER OF ELEMENTS =	127
TOTAL VOLUME OF INPUT ELEMENTS =	89.399		
TOTAL AREA OF INPUT ELEMENTS =	160.1581	TOTAL NUMBER OF ELEMENTS =	130
TOTAL VOLUME OF INPUT ELEMENTS =	89.399		
TOTAL AREA OF INPUT ELEMENTS =	174.4336	TOTAL NUMBER OF ELEMENTS =	131
TOTAL VOLUME OF INPUT ELEMENTS =	118.473		
TOTAL AREA OF INPUT ELEMENTS =	204.8862	TOTAL NUMBER OF ELEMENTS =	137
TOTAL VOLUME OF INPUT ELEMENTS =	179.496		
TOTAL AREA OF INPUT ELEMENTS =	219.9242	TOTAL NUMBER OF ELEMENTS =	143
TOTAL VOLUME OF INPUT ELEMENTS =	198.834		
TOTAL AREA OF INPUT ELEMENTS =	244.2043	TOTAL NUMBER OF ELEMENTS =	161
TOTAL VOLUME OF INPUT ELEMENTS =	232.146		
TOTAL AREA OF INPUT ELEMENTS =	254.0524	TOTAL NUMBER OF ELEMENTS =	215
TOTAL VOLUME OF INPUT ELEMENTS =	235.955		
TOTAL AREA OF INPUT ELEMENTS =	257.2163	TOTAL NUMBER OF ELEMENTS =	230
TOTAL VOLUME OF INPUT ELEMENTS =	240.742		
TOTAL AREA OF INPUT ELEMENTS =	260.4069	TOTAL NUMBER OF ELEMENTS =	248
TOTAL VOLUME OF INPUT ELEMENTS =	253.090		
TOTAL AREA OF INPUT ELEMENTS =	286.6821	TOTAL NUMBER OF ELEMENTS =	254
TOTAL VOLUME OF INPUT ELEMENTS =	261.139		
TOTAL AREA OF INPUT ELEMENTS =	288.2921	TOTAL NUMBER OF ELEMENTS =	272
TOTAL VOLUME OF INPUT ELEMENTS =	261.191		

FIGURE 12 ILLUSTRATION OF ACCUMULATED SURFACE AREA PRINTOUT FOR IMAGE.

SAMPLE CASES

Usually the most difficult aspect of using any computer program for the first time is mental inertia involved. During the initial learning period, the user gains the necessary confidence required to obtain useful results from the program. The learning period varies with the size and complexity of the particular computer program and the individual involved. The IMAGE program is small and therefore, the learning period should be short.

This section presents four example problems designed to help the first-time user in overcoming the initial mental inertia. Once familiar with the program input, the user will find that providing data to the IMAGE program is not unlike providing data for a configuration layout.

Three-View, Nose Right

The first example is a three-view of a suborbital maneuvering vehicle. The input data for this example is illustrated in figure 12. Also illustrated in figure 13 is an inset drawing resulting from the input data shown. The drawing is reduced several fold from the original 11 x 17 inch drawing.

Each line of information in figure 13 represents a "card" of input data except as noted in the following discussion. The first card is the title card. This hollerith information appears just below the program heading as illustrated in the inset drawing. The information presented can be up to 59 characters of the TITLE card. The next card is the entire $TYPE32 data set. Note that no data is entered in the data set for this example. Therefore, the program default values are used. By omitting the data entries in this section, the following input data is implied:

PRINTS = 0, (no printing of quadrilateral characters)

IORIEN = 0, (normal cross section input)

ISTAT3 = 1, (one vehicle section)

ITAPE = 0, (geometric input data in UNIT 5)

IREW8 = 0, (rewind UNIT 8 before reading - no affect)

XSC = 1.0, (X - scale factor)

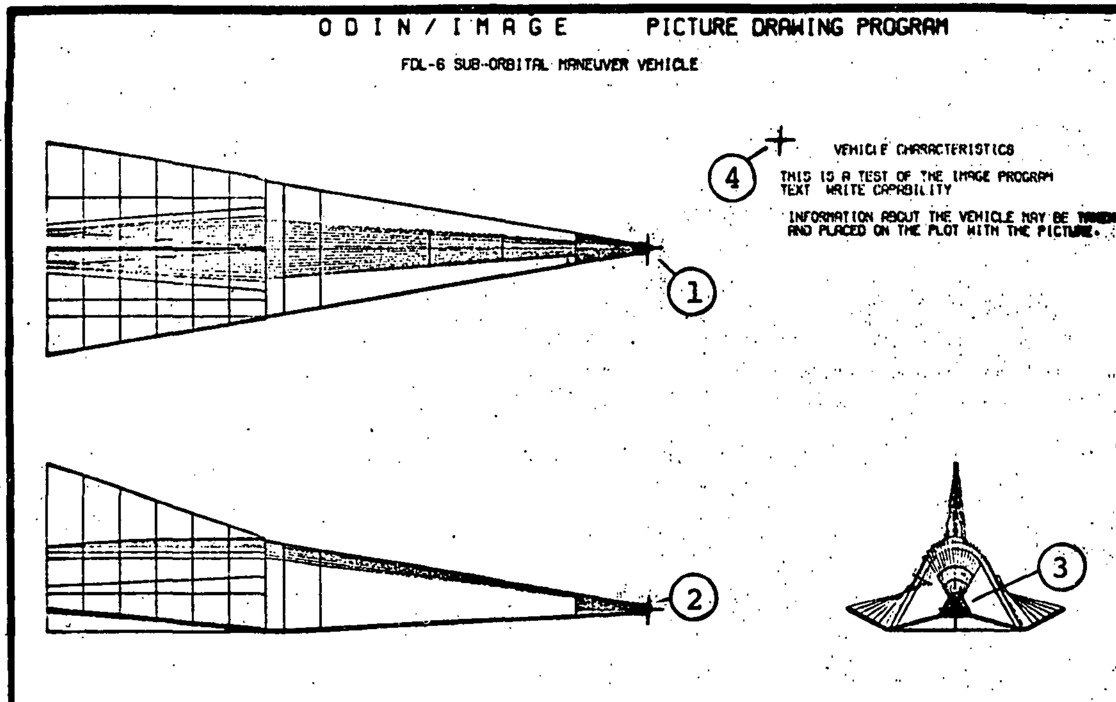
```
FDL-6 SUB-ORBITAL MANEUVER VEHICLE
$TYPE32   $

    [GEOMETRY DATA]

$TYP3435
PSI      = 90..
PHI      = 90..
DXG      = 6.,
DYG      = 6..
SCAL     = 10..
$
$TYP3435
PHI      = 0.,
DXG      = 0.,
DYG      = -5..
$
$TYP3435
PSI      = 0..
DXG      = 3.,
DYG      = 0.,
$
$TYP3435
TEXT     = .TRUE..
DXG      = -2..
DYG      =  7..
LAST     = 1,
$
        VEHICLE CHARACTERISTICS
0
 THIS IS A TEST OF THE IMAGE PROGRAM.
 TEXT  WRITE CAPABILITY
0
  INFORMATION ABOUT THE VEHICLE MAY BE TAKEN FROM THE DATA BASE
  AND PLACED ON THE PLOT WITH THE PICTURE.
2
   END OF PICTURE PHASE
```

Braced groups in the listing, top to bottom: 1, 2, 3, 4

Actual Text



+ DXG, DYG Reference Points. Illustration Assumes the Vehicle Reference Axes are at the Nose.

FIGURE 13   ILLUSTRATION OF INPUT DATA FOR THREE-VIEW DRAWING - NOSE RIGHT

YSC = 1.0, (Y - scale factor)

ZSC = 1.0, (Z - scale factor)

DELX = 0., (X - translation)

DELY = 0., (Y - translation)

DELZ = 0., (Z - translation)

Even though no data is placed in the $TYPE32 data set, the empty data set shown must be present. The minimum information that must appear in the dummy data set includes the name, the opening $ and closing $ as follows:

$TYPE32 $

At least one space must appear after the NAMELIST name.

The next set of data is the geometric input data indicated by:

GEOMETRY DATA

The actual data is omitted from the figure for clarity of presentation. Appendix A is a listing of the actual data used in this and other examples.

The next series of data sets are the $TYP3435 data sets. These data control the position and orientation of a sequence of pictures representing the data described above. The text option is also illustrated. The correspondence between the $TYP3435 data sets and the pictures (text) is indicated by the circled numbers. The crosshairs on the individual drawings refer to the location of the reference system at the start of the individual picture (text). These locations are controlled by DXG and DYG.

The first $TYP3435 data set defines a plan view orientation with the reference axes located 6 inches to the right and 6 inches above the initial reference system of the plotting device. The data is to be scaled from the longest X-dimension to fit within a 10-inch imaginary frame. All other data in this data set will be the default values as follows:

THETA = 0 (pitch angle in degrees)

ICS = 0 (connect all four points)

IREFL = 0	(draw reflected plane)
ISHAD = 0	(omit drawing rear facing elements)
IAREA = 0	(no section areas will be printed)
IQUAD = 0	(actual corner points will be drawn)
LAST = 0	(this is not the last picture)
XLG = computed	(left side of frame)
XRG = computed	(right side of frame)
YBG = computed	(bottom of frame)
YTG = computed	(top of frame)
TEXT = .FALSE.	(this is a picture option, not text)
XMOVE = 17.0	(move plot device 17 inches in the X-direction after completion of the present picture series)
YMOVE = 0.0	(plot device does not move in the Y-direction after the current picture series)
HTEXT = 0.14	(character height for textual information)

The second $TYP3435 data set defines a profile view of the vehicle. This view is to be located 5 inches below the plan view (described above). Note that all data remains unchanged between data set definitions. Only those variables requiring change from the previous set need be reset by the user.

The third $TYP3435 data set defines a front view of the vehicle. This view is to be located 3 inches to the right of the profile view.

The fourth $TYP3435 data set defines a text option, a series of text cards which will be read from cards in 80 column format. Only five input variables have meaning to the text option.

TEXT	(activates text option)
DXG	(X-movement of the plot device reference to the start of the first line of text)

DYG (Y-movement of the plot device reference to the start of the first line of text)

LAST (indicates the last \$TYP3435 data set in the current series)

HTEXT (character height for textual information)

The actual text immediately follows the \$TYP3435 data set which activates the text option. The first column of each text card is reserved for print control as follows:

0 - skip a line

2 - terminate the text option

In the illustration, the start of the first line of text is positioned 7 inches above and 2 inches to the left of the previous (front view) reference and the characters are 0.14 inches high.

The parameter LAST is set to 1 in the text option indicating that current set to be the last \$TYP3435 data set in the series. The program flow logic will return for a new title card. Then since the next (title) card has the characters "99" in columns 71 and 72, the IMAGE program execution is terminated.

Three-View, Nose Left

The second example is a three view of a shuttle orbiter-type vehicle. The input data for this example is illustrated in figure 13. The drawing resulting from the data is inset into the figure. Except for the geometric data employed, the primary difference between this example and the previous one is the orientation of the views. This example uses a nose left orientation and a rear view. The previous example used a nose right orientation and a front view. Comparison of figures 12 and 13 will illustrate the differences between the data definition for the two types of vehicle orientation.

The geometric data for this example was generated by a special computer program called PANEL (reference 3). Therefore, the \$TYPE32 data set specifies the data source to be external by the parameter.

ITAPE = 1

This parameter specifies the data is to be read from the internal file, UNIT 8. The correspondence between UNIT 8 and the external file which contains the actual data is established through the sixth parameter on the program card. The external source of data is accessed by the IMAGE execution control card through file substitution of the parameter with the external file name as follows:

```
OIMAGE,,,,,GEOM.
```

In the above illustration the file, GEOM, contains the geometric data for the IMAGE program. Of course, the geometric data must have been placed on the GEOM file by the PANEL program in a similar manner as that illustrated above.

```
OPANEL,,,,,GEOM.
```

In the case of the PANEL program, the geometry unit, GEOM, happens to be the fifth file parameter and must have been substituted accordingly at the time PANEL was executed.

The $TYPE32 data set also specifies a translation of the original data in GEOM (TAPE8) 1200 inches forward and 700 inches down. The reason for the translation is to reposition the reference axis system for plotting purposes. The original data for the orbiter generated in the PANEL program was referenced to the coordinate system of the boost vehicle upon which the orbiter was mounted. The translated reference system is coincidental with the nose of the orbiter as indicated by the inset to figure 14.

The picture sequence and text option which are defined by the $TYP3435 data sets are similar, except for orientation, to the sequence described in the previous example. Therefore, the reader is referred to that section for discussion of the input data.

Oblique Views

The third example shown in figure 15 illustrates a series of oblique views of the suborbital maneuvering vehicle of example 1. The geometric data is read from the normal input device and not translated as was the data of example 2. The actual data is shown in Appendix A.

Each $TYP3435 data set refers to a numbered view. The correspondence of the individual data set with the view is indicated in the figure. The significant difference between these data sets and the data sets of the other examples are:

```
* PARAMETRIC CUBIC BODY SECTION
$TYPE32  ITAPE=1,
 DELX=1200.,
 DELZ=-700.,
$
$TYP3435
PSI     = -90.,
PHI     = -90.,
DXG     = 0.,
DYG     = 6.,
SCAL    = 10.,
$
$TYP3435
PHI     = 0.,
DXG     = 0.,
DYG     = -5.,
$
$TYP3435
PSI     = 180.,
DXG     =  7.,
DYG     = 0.,
$
$TYP3435
TEXT    = .TRUE.,
LAST    = 1.
DXG     = -2.,
DYG     =  7.,
$
        VEHICLE CHARACTERISTICS

THIS IS A TEST OF THE IMAGE PROGRAM
TEXT  WRITE CAPABILITY

 INFORMATION ABOUT THE VEHICLE MAY BE
 TAKEN FROM THE DATA BASE
 AND PLACED ON THE PLOT WITH THE PICTURE.
2
  END OF PICTURE PHASE
```

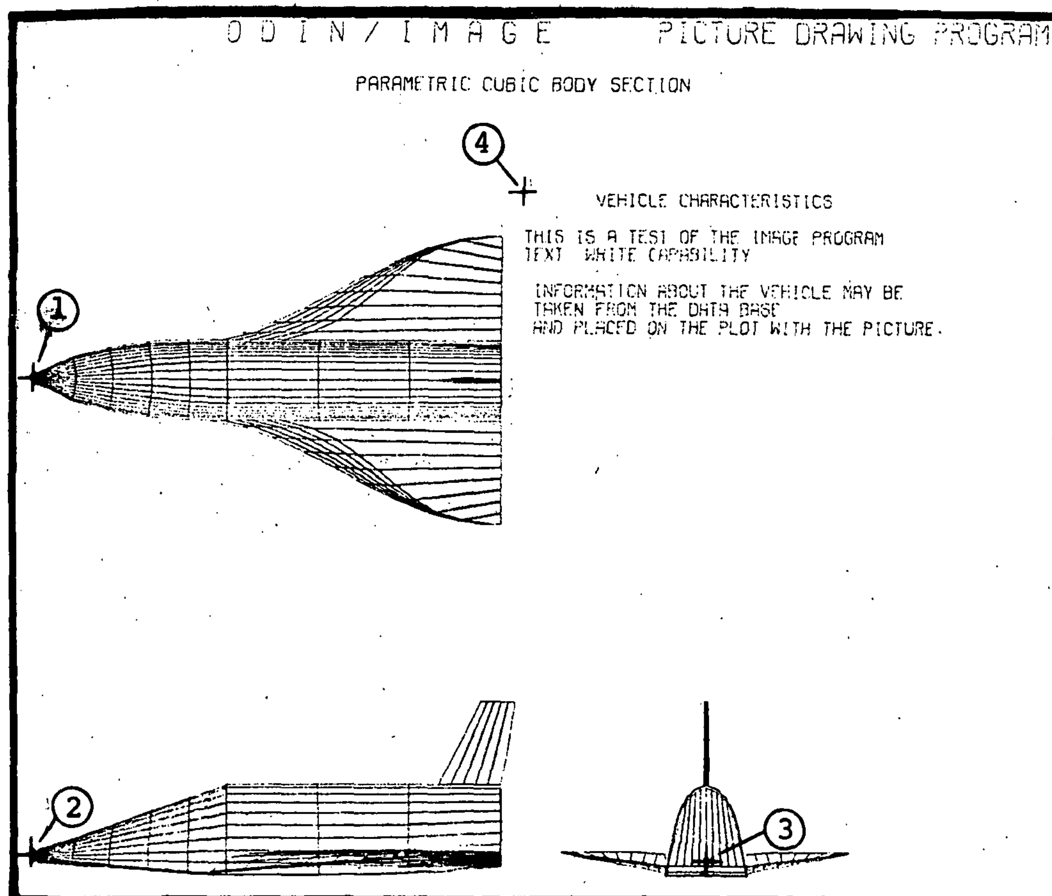
1

2

3

4

ACTUAL TEXT



Col. 71-72 → 99

FIGURE 14   ILLUSTRATION OF INPUT DATA FOR THREE-VIEW DRAWING - NOSE LEFT.

```
FDL-6 SUB-ORBITAL MANEUVER VEHICLE
$TYPE32 .$

   [GEOMETRIC DATA]

$TYP3435
 PSI = -35..
THETA = 25..
 PHI = -20..
 DXG = 1..
 DYG = 1..               1
 XLG=0..
 XRG=33..
 YBG=0..
 YTG=33..
SCAL    = 10..
$
$TYP3435
 PSI = 0..
THETA = 0..
 PHI = 0..               2
DXG      = 1.5.
 DYG = 5..
$
$TYP3435
 PSI = -35..
THETA = -25..
 PHI = 10..              3
 DXG = 6..
 DYG = 2..
$
$TYP3435
 PSI = -135..
THETA = -40..
 PHI =  -15..            4
 DYG = -2..
 DXG = 0..
 LAST = 1.
$
```

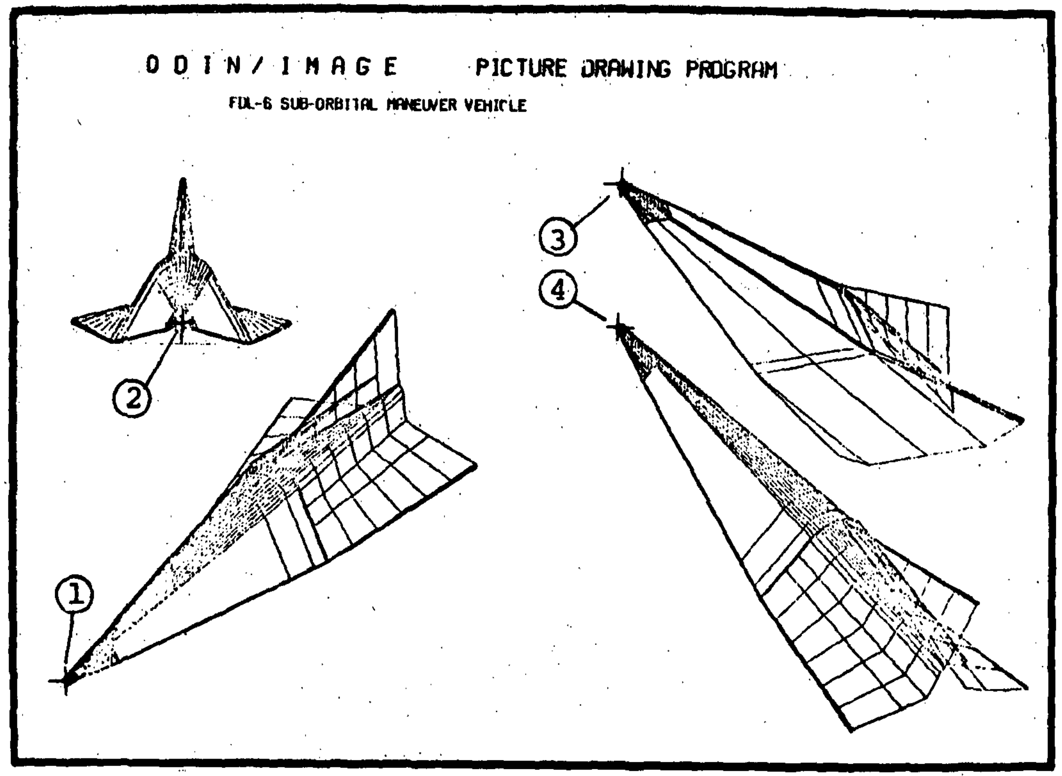


FIGURE 15   ILLUSTRATION OF INPUT DATA FOR A SEQUENCE OF OBLIQUE VIEWS.

1. Differences in viewing angles.

2. Different locations for the views.

3. The absence of text in this example.

4. The frame dimensions XLG, XRG, YBG and YTG are read in rather than computed by the program.

Notice the frame dimensions are established so that the frame size in the X-direction is the same as the frame size in the Y-direction.

$$XRG - XLG = YTG - YBG$$

The above criteria establishes a "square frame" which is essential to an undistorted picture.

ODIN Input Example

This example is an illustration of a three-view drawing of a shuttle orbiter recently studied at NASA Langley Research Center using the ODIN procedure to simulate certain aspects of the design process. Figure 16 is an illustration of the data setup for IMAGE within the ODIN framework. This figure portrays a set of data describing a three view, nose left with vehicle characteristics printed on the picture. The resulting picture is inset in the figure. Both the picture definition data and the optional text information is augmented by data base variable names. The variable names are data base interfaces and therefore delimited by (≠). For details of the interface language, the reader is referred to reference 2.

The primary differences between this example and the previous one illustrated in figure 13 are:

1. The present example is an illustration of the program being executed within the ODIN system. The illustrated data was extracted directly from an ODIN simulation.

2. The data is generated by the program of reference 1 which was also executed in the ODIN simulation.

3. The frame size is input from the data base and is based upon the current fuselage length (XLFUS from the data base).

```
#EXECUTE IMAGE#
#ADD DM=XLFUS/SCAL#
ORBITER DESIGN  25K  P/L       1  INCH= #DM     #  INCHES
$TYPE32  ITAPE=1,  ISTAT3=3,  $
$TYP3435
PSI    = -90.,
PHI    = -90.,
 DXG=0.,
 DYG=6.,
 XRG=#XLFUS #,
 XLG=0.,
 YBG=0.,
 YTG=#XLFUS #,
SCAL   = #SCAL #,
$
$TYP3435
PHI    = 0.,
DXG    = 0.,
DYG    = -5.,
$
$TYP3435
PSI    = 180.,
 DXG=12.,
DYG    = 0.,
$
$TYP3435
TEXT   = .TRUE.,
LAST   = 1,
 DXG=-2.5,
 DYG=7.5,
$
  VEHICLE CHARACTERISTICS
0
 1.  MASS PROPERTIES
0
           LANDED WEIGHT                                    #WLAND #   LB
           ENTRY WEIGHT                                     #WENTRY#   LB
           DESIGN CG, SUBSONIC,  25K P/L                    #XCGIN #   FT
           DESIGN CG HYPERSONIC,  13K P/L                   #XCGHYP#   FT
0
 2.  GEOMETRY
0
#ADD DM=XLFUS/12.#
           BODY LENGTH                                      #DM    #   FT
           TOTAL WING AREA                                  #STOTAL#   SQFT
           CHORD, THEO ROOT                                 #CROOT #   FT
           CHORD, TIP                                       #CTIP  #   FT
           ASPECT RATIO                                     #AR    #
           LEADING EDGE SWEEP ANGLE                         #SWTLE #   DEG
           TRAILING EDGE SWEEP ANGLE                        #SWTTE #   DEG
#ADD DM=CRE+CTE*SPEXP/2.#
           ELEVON AREA, TOTAL                               #DM    #   SQFT
           EXPOSED WING LOCATION                            #XOF   #   FT
0
 3.  AERODYNAMIC CHARACTERISTICS
0
           DESIGN TRIM LIFT COEFFICIENT                     #CLT(8)#
           DESIGN MIMIMUM LANDING SPEED                     #VMD   #   KNOTS
           MAX TRIM ALPHA HYPERSONIC DESIGN COND            #ALTRIM#   DEG
2
 END OF PICTURE PHASE
```

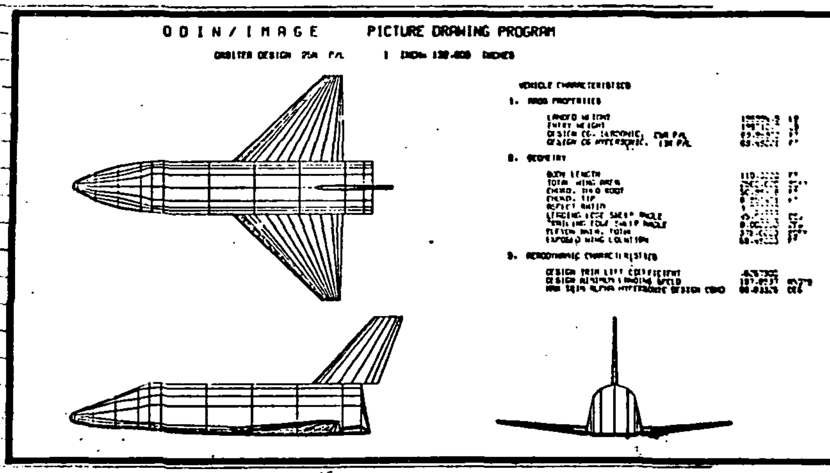


FIGURE 16 ILLUSTRATION OF INPUT DATA FOR IMAGE WHEN USED IN ODIN

4. The text option display actual vehicle characteristics from the data base for the current vehicle. The delimited (≠) names denote the data base names for the indicated quantities.

Each line of information in figure 16 represents a card of input data. The first card illustrated is the DIALOG control directive.

≠EXECUTE IMAGE≠

The control directive is input information to the DIALOG executive system and accomplishes the following functions:

1. Retrieves the IMAGE program from permanent storage at the beginning of the simulation.

2. Executes the IMAGE program providing the necessary file substitution to pass geometry file to the program from the program of reference 1.

3. Provides for the return of control to the DIALOG executive system after completion of the execution of IMAGE.

The second card is an ADD command which is part of the communication language in the DIALOG executive system described in reference 2. The function of this card,

≠ADD DM = XLFUS/SCAL≠

is to compute a scale of the drawing (DM) to be used on the picture title card. In the illustration the picture scale is 1 to 132 inches. The card illustrated above is "removed" from the input stream by DIALOG and is therefore not read by the IMAGE program.

The third card is the TITLE card. This card is generally the first card (when IMAGE is used as an independent program). However, when used with the DIALOG executive system, the TITLE card may not be the first card due to interface requirements with other programs and with the ODIN data base. However, after the input data is preprocessed by DIALOG, the modified input file will be identical in format to the previous examples. No delimited information will appear.

The $TYPE32 data resets the parameter,

ITAPE = 1

which specifies the geometric data will be obtained from an external source. The DIALOG executive system automatically maintains the proper correspondence between the IMAGE data file and the data files of the program of reference 1. The user need not be concerned with file substitution when using IMAGE with the ODIN system.

Another parameter set in $TYPE32 is:

ISTAT3=3

This variable specifies that three sections of data on the geometry file will be plotted. Actually more data resided on the geometry file for the example simulation but the first three sections (WING, BODY and VERTICAL TAIL) were the only ones of interest for plotting purposes.

The next series of data sets ($TYP3435) control the position and orientation of a sequence of pictures and text. The first set in the series also defines the frame size and dimensions. Frame size is referenced to the fuselage length, XLFUS and the data is scaled to the parameter, SCAL. Both of these parameters come from the data base as indicated in figure 16.

The last $TYP3435 data set activates the text option. The actual text cards follow the $TYP3435 data set. These cards are modified by data base information as indicated by the delimited variable names. Each delimited name is replaced by the current value of the variable from data base. DIALOG performs this replacement function before the data is read by IMAGE. In this manner, the current vehicle characteristics are extracted from the data base and printed with the geometric representation of the vehicle.

CONCLUDING REMARKS

The IMAGE program lends support to many technology programs using the same geometry definition. The surface definition is based on the quadrilateral elements to describe the surface of an arbitrary configuration. In the IMAGE program, this geometry is converted to a picture-like drawing of the configuration for inspection. The computer program is important as a stand alone program or when used within the ODIN (for Optimal Design Integration) system. The use of IMAGE is threefold:

1. It provides a visual check on the geometric data input for both the IMAGE program and other technology programs using the data.

2. It provides for monitoring of geometric perturbations particularly when used within the ODIN system where geometric perturbations are constantly taking place.

3. It provides a convenient means of reporting on current geometric characteristics of a vehicle under study.

The advant of the ODIN system as a means of performing design analysis has resulted in greater emphasis on the development of geometric definitions as a separate and distinct technology area. The IMAGE program represents the first step towards the expansion of this technology, and is expected to have wide applications in the future use of the ODIN system.

REFERENCES

1. Gentry, A.: Hypersonic Arbitrary-Body Aerodynamic Computer Program, Mark III Version, Volume I, User's Manual, Douglas Report, DAC 61552, April, 1968.

2. Glatt, C. R., Hague, D. S. and Watson, D. A.: DIALOG: An Executive Computer Program for Linking Independent Programs. National Aeronautics and Space Administration Contractor Report CR-2296. Washington D. C. 1973.

3. Hague, D. S. and Glatt, C. R.: Optimal Design Integration of Military Flight Vehicles - ODIN/MFV. AFFDL-TR-72-132. December 1972.

APPENDIX A

LISTING OF TEXT DATA FOR THE IMAGE PROGRAM

The following pages present a list of the element data for the suborbital maneuvering vehicle test data discussed in connection with the use of the IMAGE program.

0.0000	0.0000	0.00002	0.0000	0.0000	0.00000	-0NOSE 3AERO
0.0000	0.0000	0.00000	0.0000	0.0000	0.00000	-0NOSE 3AERO
0.0000	0.0000	0.00000	0.0000	0.0000	0.00000	-0NOSE 3AERO
0.0000	0.0000	0.00000	0.0000	0.0000	0.00000	-0NOSE 3AERO
0.0000	0.0000	0.00000	0.0000	0.0000	0.00000	-0NOSE 3AERO
-.0083	0.0000	-.06661	-.0083	.0228	-.06260	-0NOSE 3AERO
-.0083	.0428	-.05100	-.0083	.0577	-.03330	-0NOSE 3AERO
-.0083	.0656	-.01160	-.0083	.0656	.01160	-0NOSE 3AERO
-.0083	.0577	.03330	-.0083	.0428	.05100	-0NOSE 3AERO
-.0083	.0228	.06260	-.0083	-.0000	.06660	-0NOSE 3AERO
-.0333	0.0000	-.12501	-.0333	.0428	-.11750	-0NOSE 3AERO
-.0333	.0803	-.09580	-.0333	.1083	-.06250	-0NOSE 3AERO
-.0333	.1231	-.02170	-.0333	.1231	.02170	-0NOSE 3AERO
-.0333	.1083	.06250	-.0333	.0803	.09580	-0NOSE 3AERO
-.0333	.0428	.11750	-.0333	-.0000	.12500	-0NOSE 3AERO
-.0750	0.0000	-.17901	-.0750	.0612	-.16820	-0NOSE 3AERO
-.0750	.1151	-.13710	-.0750	.1550	-.08950	-0NOSE 3AERO
-.0750	.1763	-.03110	-.0750	.1763	.03110	-0NOSE 3AERO
-.0750	.1550	.08950	-.0750	.1151	.13710	-0NOSE 3AERO
-.0750	.0612	.16820	-.0750	-.0000	.17900	-0NOSE 3AERO
-.1625	0.0000	-.23601	-.1625	.0807	-.22180	-0NOSE 3AERO
-.1625	.1517	-.18080	-.1625	.2044	-.11800	-0NOSE 3AERO
-.1625	.2324	-.04100	-.1625	.2324	.04100	-0NOSE 3AERO
-.1625	.2044	.11800	-.1625	.1517	.18080	-0NOSE 3AERO
-.1625	.0807	.22180	-.1625	-.0000	.23600	-0NOSE 3AERO
-.2500	0.0000	-.25001	-.2500	.0855	-.23490	-0NOSE 3AERO
-.2500	.1607	-.19150	-.2500	.2165	-.12500	-0NOSE 3AERO
-.2500	.2462	-.04340	-.2500	.2462	.04340	-0NOSE 3AERO
-.2500	.2165	.12500	-.2500	.1607	.19150	-0NOSE 3AERO
-.2500	.0855	.23490	-.2500	-.0000	.25000	-0NOSE 3AERO
-.25	0.0	-.25	2-.25	.02	-.249	NFBR 3AERO
-.25	.05	-.245	-.25	.08	-.238	NFBR 3AERO
-.25	.12	-.22	-.25	.15	-.20	NFBR 3AERO
-.25	.17	-.195	-.25	.19	-.16	NFBR 3AERO
-.25	.215	-.14	-.25	.22	-.12	NFBR 3AERO
-4.	0.0	-.45	1-4.	.04	-.45	NFBR 3AERO
-4.	.09	-.45	-4.	.15	-.45	NFBR 3AERO
-4.	.24	-.45	-4.	.34	-.45	NFBR 3AERO
-4.	.41	-.45	-4.	.53	-.45	NFBR 3AERO

-4.	.66	-.45	-4.	.82	-.45	NFBB	3AERO
-.25	.23	-.06	2-.25	.24	-.06	NFBT	3AERO
-.25	.245	-.035	-.25	.248	-.02	NFBT	3AERO
-.25	.250	0.0	-.25	.248	.02	NFBT	3AERO
-.25	.245	.045	-.25	.243	.06	NFBT	3AERO
-.25	.235	.09	-.25	.22	.12	NFBT	3AERO
-.25	.21	.14	-.25	.19	.165	NFBT	3AERO
-.25	.19	.17	-.25	.165	.19	NFBT	3AERO
-.25	.15	.215	-.25	.13	.22	NFBT	3AERO
-4.	.89	-.34	1-4.	.83	-.22	NFBT	3AERO
-4.	.78	-.11	-4.	.75	-.06	NFBT	3AERO
-4.	.72	0.0	-4.	.70	.04	NFBT	3AERO
-4.	.66	.12	-4.	.63	.17	NFBT	3AERO
-4.	.60	.24	-4.	.56	.31	NFBT	3AERO
-4.	.54	.36	-4.	.51	.42	NFBT	3AERO
-4.	.48	.49	-4.	.45	.54	NFBT	3AERO
-4.	.42	.60	-4.	.39	.67	NFBT	3AERO
-.25	.13	.22	2-.25	.12	.225	NFBT	3AERO
-.25	.10	.235	-.25	.09	.24	NFBT	3AERO
-.25	.075	.243	-.25	.06	.244	NFBT	3AERO
-.25	.05	.245	-.25	.035	.247	NFBT	3AERO
-.25	.02	.249	-.25	0.0	.250	NFBT	3AERO
-4.	.39	.67	1-4.	.36	.72	NFBT	3AERO
-4.	.32	.76	-4.	.29	.79	NFBT	3AERO
-4.	.24	.83	-4.	.20	.85	NFBT	3AERO
-4.	.155	.87	-4.	.11	.89	NFBT	3AERO
-4.	.05	.895	-4.	0.0	.90	NFBT	3AERO
-4.	0.0	-.45	2-4.	.82	-.45	FBB	3AERO
-20.	0.0	-1.2	1-20.	3.59	-1.2	FBB	3AERO
-4.	.89	-.34	2-4.	.39	.67	FBS	3AERO
-18.	3.29	-.99	1-18.	1.41	2.44	FBS	3AERO
-18.	3.29	-.99	2-18.	1.41	2.44	FBS	3AERO
-20.	3.645	-1.1	1-20.	1.58	2.74	FBS	3AERO
-21.	3.83	-1.1	1-21.	1.64	2.84	FBS	3AERO
-4.	.39	.67	2-4.	.36	.72	FBT	3AERO
-4.	.32	.76	-4.	.29	.79	FBT	3AERO
-4.	.24	.83	-4.	.20	.85	FBT	3AERO
-4.	.155	.87	-4.	.11	.89	FBT	3AERO
-4.	.05	.895	-4.	0.0	.90	FBT	3AERO

-8.	.72	1.23	1-8.	.66	1.30	FBT	3AERO
-8.	.59	1.38	-8.	.52	1.43	FBT	3AERO
-8.	.43	1.49	-8.	.36	1.52	FBT	3AERO
-8.	.275	1.56	-8.	.19	1.58	FBT	3AERO
-8.	.19	1.595	-8.	0.0	1.60	FBT	3AERO
-12.	1.0	1.74	1-12.	.93	1.84	FBT	3AERO
-12.	.93	1.95	-12.	.74	2.04	FBT	3AERO
-12.	.62	2.12	-12.	.52	2.18	FBT	3AERO
-12.	.39	2.24	-12.	.25	2.27	FBT	3AERO
-12.	.11	2.295	-12.	0.0	2.3	FBT	3AERO
-18.	1.4	2.44	1-18.	1.31	2.6	FBT	3AERO
-18.	1.19	2.77	-18.	1.06	2.89	FBT	3AERO
-18.	.88	3.03	-18.	.74	3.115	FBT	3AERO
-18.	.56	3.2	-18.	.38	3.25	FBT	3AERO
-18.	.15	3.29	-18.	0.0	3.3	FBT	3AERO
-18.	1.4	2.44	2-18.	1.31	2.6	FBT	3AERO
-18.	1.18	2.77	-18.	1.06	2.89	FBT	3AERO
-18.	.88	3.03	-18.	.74	3.115	FBT	3AERO
-18.	.56	3.2	-18.	.38	3.25	FBT	3AERO
-18.	.15	3.29	-18.	0.0	3.3	FBT	3AERO
-20.	1.58	2.74	1-20.	1.47	2.905	FBT	3AERO
-20.	1.31	3.09	-20.	1.17	3.22	FBT	3AERO
-20.	.98	3.36	-20.	.82	3.45	FBT	3AERO
-20.	.62	3.54	-20.	.42	3.60	FBT	3AERO
-20.	.16	3.638	-20.	0.0	3.64	FBT	3AERO
-21.	1.64	2.64	1-21.	1.53	3.02	FBT	3AERO
-21.	1.37	3.23	-21.	1.23	3.37	FBT	3AERO
-21.	1.03	3.53	-21.	.86	3.63	FBT	3AERO
-21.	.65	3.725	-21.	.44	3.79	FBT	3AERO
-21.	.17	3.84	-21.	0.0	3.85	FBT	3AERO
-19.99	0.0	-1.1995	2-19.99	3.58	-1.1995	ABB	3AERO
-20.	0.0	-1.2	1-20.	3.58	-1.2	ABB	3AERO
-21.	0.0	-1.2	1-21.	3.78	-1.2	ABB	3AERO
-33.	0.0	-1.2	1-33.	3.78	-1.2	ABB	3AERO
-21.	3.78	-1.2	2-21.	3.78	-1.2	CSS	3AERO
-33.	3.78	-1.2	1-33.	5.90	-.12	CSS	3AERO
-21.	3.90	-1.01	2-21.	2.84	.90	ABO	3AERO
-23.	4.24	-.81	1-23.	2.84	.90	ABO	3AERO
-25.	4.56	-.63	1-25.	2.84	.90	ABO	3AERO

-27.	4.92	-.43	1-27.	2.84	.90	ABO	3AERO
-29.	5.24	-.29	1-29.	2.84	.90	ABO	3AERO
-31.	5.58	-.055	1-31.	2.84	.90	ABO	3AERO
-33.	5.89	.13	1-33.	2.84	.90	ABO	3AERO
-21.	2.84	.90	2-21.	2.33	1.87	ABO	3AERO
-23.	2.84	.90	1-23.	2.16	1.76	ABO	3AERO
-25.	2.84	.90	1-25.	2.0	1.67	ABO	3AERO
-27.	2.84	.90	1-27.	1.83	1.58	ABO	3AERO
-29.	2.84	.90	1-29.	1.66	1.48	ABO	3AERO
-31.	2.84	.90	1-31.	1.50	1.40	ABO	3AERO
-33.	2.84	.90	1-33.	1.33	1.32	ABO	3AERO
-21.	2.33	1.67	2-21.	1.82	2.8	ABST	3AERO
-21.	1.59	3.15	-21.	1.33	3.44	ABST	3AERO
-23.	2.16	1.76	1-23.	1.7	2.8	ABST	3AERO
-23.	1.51	3.15	-23.	1.27	3.44	ABST	3AERO
-25.	2.0	1.67	1-25.	1.58	2.8	ABST	3AERO
-25.	1.42	3.15	-25.	1.20	3.44	ABST	3AERO
-27.	1.83	1.58	1-27.	1.45	2.8	ABST	3AERO
-27.	1.32	3.15	-27.	1.12	3.44	ABST	3AERO
-29.	1.66	1.48	1-29.	1.3	2.8	ABST	3AERO
-29.	1.19	3.15	-29.	1.02	3.44	ABST	3AERO
-31.	1.50	1.40	1-31.	1.15	2.8	ABST	3AERO
-31.	1.07	3.15	-31.	0.93	3.44	ABST	3AERO
-33.	1.33	1.32	1-33.	1.05	2.8	ABST	3AERO
-33.	0.98	3.15	-33.	0.74	3.44	ABST	3AERO
-21.	1.33	3.44	2-21.	1.2	3.55	ABST	3AERO
-21.	1.05	3.65	-21.	0.90	3.73	ABST	3AERO
-21.	0.75	3.80	-21.	0.60	3.86	ABST	3AERO
-21.	0.45	3.90	-21.	0.30	3.93	ABST	3AERO
-21.	0.15	3.95	-21.	0.0	3.96	ABST	3AERO
-23.	1.27	3.44	1-23.	1.13	3.55	ABST	3AERO
-23.	0.99	3.65	-23.	0.85	3.73	ABST	3AERO
-23.	0.72	3.8	-23.	0.57	3.86	ABST	3AERO
-23.	0.44	3.9	-23.	0.23	3.93	ABST	3AERO
-23.	0.23	3.93	-23.	0.20	3.93	ABST	3AERO
-25.	1.20	3.44	1-25.	1.08	3.55	ABST	3AERO
-25.	0.95	3.65	-25.	0.82	3.73	ABST	3AERO
-25.	0.67	3.8	-25.	0.53	3.86	ABST	3AERO
-25.	0.37	3.9	-25.	0.37	3.9	ABST	3AERO

-25.	0.37	3.9	-25.	0.29	3.9	ABST 3AERO
-27.	1.12	3.44	1-27.	1.02	3.55	ABST 3AERO
-27.	0.93	3.65	-27.	0.82	3.73	ABST 3AERO
-27.	0.68	3.8	-27.	0.49	3.86	ABST 3AERO
-27.	0.49	3.86	-27.	0.49	3.86	ABST 3AERO
-27.	0.49	3.86	-27.	0.39	3.86	ABST 3AERO
-29.	1.02	3.44	1-29.	0.93	3.55	ABST 3AERO
-29.	0.83	3.65	-29.	0.72	3.72	ABST 3AERO
-29.	0.63	3.78	-29.	0.63	3.78	ABST 3AERO
-29.	0.63	3.78	-29.	0.63	3.78	ABST 3AERO
-29.	0.63	3.78	-29.	0.49	3.78	ABST 3AERO
-31.	0.93	3.44	1-31.	0.86	3.55	ABST 3AERO
-31.	0.77	3.64	-31.	0.77	3.64	ABST 3AERO
-31.	0.77	3.64	-31.	0.77	3.64	ABST 3AERO
-31.	0.77	3.64	-31.	0.77	3.64	ABST 3AERO
-31.	0.77	3.64	-31.	0.61	3.64	ABST 3AERO
-33.	0.74	3.44	1-33.	0.74	3.44	ABST 3AERO
-33.	0.74	3.44	-33.	0.74	3.44	ABST 3AERO
-33.	0.74	3.44	-33.	0.74	3.44	ABST 3AERO
-33.	0.74	3.44	-33.	0.74	3.44	ABST 3AERO
-33.	0.74	3.44	-33.	0.74	3.44	ABST 3AERO
-0.2500	0.2191	-0.12042	-0.2500	0.2225	-0.11400	BLE 3AERO
-0.2500	0.2257	-0.10750	-0.2500	0.2287	-0.10100	BLE 3AERO
-0.2500	0.2315	-0.09433	-0.2500	0.2342	-0.08760	BLE 3AERO
-4.0000	0.8200	-0.45831	-4.0000	0.8519	-0.45200	BLE 3AERO
-4.0000	0.8789	-0.43390	-4.0000	0.8970	-0.40690	BLE 3AERO
-4.0000	0.9033	-0.37500	-4.0000	0.8970	-0.34310	BLE 3AERO
-20.0000	3.5800	-1.20001	-20.0000	3.6119	-1.19370	BLE 3AERO
-20.0000	3.6389	-1.17560	-20.0000	3.6570	-1.14860	BLE 3AERO
-20.0000	3.6633	-1.11670	-20.0000	3.6570	-1.0848	BLE 3AERO
-21.0000	3.7600	-1.20001	-21.0000	3.7919	-1.19370	BLE 3AERO
-21.0000	3.8189	-1.17560	-21.0000	3.8370	-1.14860	BLE 3AERO
-21.0000	3.8433	-1.11670	-21.0000	3.8370	-1.08480	BLE 3AERO
-21.	3.78	-1.2	2-21.	3.93	-1.12	BLE 3AERO
-21.	3.93	-1.05	-21.	3.90	-1.01	BLE 3AERO
-23.	4.21	-1.01	1-23.	4.28	-.92	BLE 3AERO
-23.	4.29	-.86	-23.	4.24	-.91	BLE 3AERO
-25.	4.55	-.82	1-25.	4.62	-.76	BLE 3AERO
-25.	4.62	-.67	-25.	4.56	-.63	BLE 3AERO

-27.	4.92	-.63	1-27.	4.96	-.56	BLE	3AERO
-27.	4.95	-.49	-27.	4.92	-.43	BLE	3AERO
-29.	5.24	-.46	1-29.	5.31	-.37	BLE	3AERO
-29.	5.30	-.3[illegible]	-29.	5.24	-.25	BLE	3AERO
-31.	5.57	-.28	1-31.	5.66	-.18	BLE	3AERO
-31.	5.65	-.11	-31.	5.58	-.055	BLE	3AERO
-33.	5.91	-.11	1-33.	6.00	0.0	BLE	3AERO
-33.	5.98	.08	-33.	5.89	.13	BLE	3AERO
-21.	0.1	3.95	2-21.	0.1	3.95	VFN	3AERO
-23.	0.20	3.93	1-23.	0.1	4.6	VFN	3AERO
-25.	0.29	3.9	1-25.	0.11	5.29	VFN	3AERO
-27.	0.39	3.86	1-27.	0.11	5.95	VFN	3AERO
-29.	0.49	3.78	1-29.	0.11	6.64	VFN	3AERO
-31.	0.61	3.64	1-31.	0.11	7.34	VFN	3AERO
-33.	0.74	3.44	1-33.	0.11	8.03	VFN	3AERO
-21.	0.1	3.95	2-21.	0.07	3.95	FNLE	3AERO
-21.	0.04	3.96	-21.	0.0	3.96	FNLE	3AERO
-23.	0.1	4.6	1-23.	0.07	4.63	FNLE	3AERO
-23.	0.04	4.64	-23.	0.0	4.65	FNLE	3AERO
-25.	0.11	5.29	1-25.	0.07	5.33	FNLE	3AERO
-25.	0.04	5.34	-25.	0.0	5.35	FNLE	3AERO
-27.	0.11	5.95	1-27.	0.07	6.0	FNLE	3AERO
-27.	0.04	6.01	-27.	0.0	6.02	FNLE	3AERO
-29.	0.11	6.64	1-29.	0.07	6.69	FNLE	3AERO
-29.	0.04	6.71	-29.	0.0	6.72	FNLE	3AERO
-31.	0.11	7.34	1-31.	0.07	7.39	FNLE	3AERO
-31.	0.04	7.41	-31.	0.0	7.43	FNLE	3AERO
-33.	0.11	8.03	1-33.	0.07	8.05	FNLE	3AERO
-33.	0.04	8.07	-33.	0.0	8.08	FNLE	3AERO