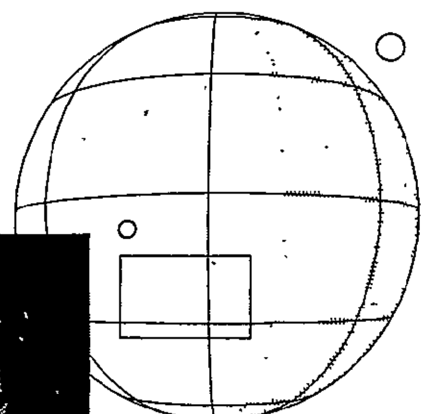
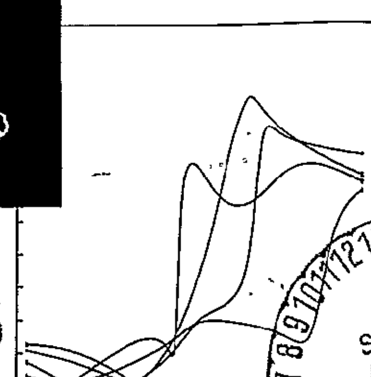
outer planet data presentation computer program

MODEL DESCRIPTION REPORT





MARTIN MARIETTA
DENVER DIVISION

MCR-71-181 Volume I

MCR-71-181

Contract JPL 953058

OUTER PLANET DATA PRESENTATION COMPUTER PROGRAM

Model Description Report

July 1971

Volume 1

Approved

Stephen J. Ducsai
Program Manager

MARTIN MARIETTA CORPORATION
DENVER DIVISION
P. O. Box 179
Denver, Colorado 80201

FOREWORD

This report has been prepared in accordance with requirements of Contract JPL 953058 to present a description of a computer program resulting from a four-month development effort performed for the Jet Propulsion Laboratory by the Martin Marietta Corporation, Denver Division. The report is submitted in two volumes. Volume I is a description of the program and its operation, Volume II contains the appendixes, which include sample data cases to aid initial input setup and program checkout.

This work was performed for the Jet Propulsion Laboratory, California [illegible] sponsored by the National A[illegible] and Space Administration under Contract NAS7-100.

CONTENTS

		Page
Foreword		ii
Contents		iii thru vi
1.	Introduction	1-1 and 1-2
2.	Summary	2-1 thru 2-14
3.	Program Description	3-1
	3.1 General Program Logic	3-1
	3.2 MAIN Program Logic and Flow Diagram	3-8
	3.3 Computer Translation Information	3-26 thru 3-29
4.	Model Symbol Dictionary	4-1
	4.1 Common Blocks	4-1
	4.2 Model Inputs	4-1
	4.3 Tape Files	4-1 thru 4-10
5.	Model Operating Procedures	5-1
	5.1 Comment Cards Contained in Program	5-1
	5.2 Special Imaging OPtions	5-1
	5.3 Example of Computer Input Cards	5-5 thru 5-14
6.	Model Subroutines	6-1
	6.1 Subroutine Classification	6-1
	6.2 Subroutines Supplied by JPL	6-1
	6.3 Major Subroutines Developed under Contract	6-13

6.4 Minor Subroutines Developed under Contract 6-117

6.5 Plotting Conversion Subroutines . . . . . . 6-130 thru 6-132

Figure

2-1 Jupiter Encounter 2-6

2-2 Uranus Encounter 2-7

2-3 Cone and Clock Angles versus Time 2-7

2-4 Satellite Disk Size versus 1 Degree Field of View 2-8

2-5 Closest Approach to Jupiter 2-8

2-6 Number of Pictures Taken 2-10

2-7 Resolution versus Time 2-10

2-8 Resolution versus Time 2-11

2-9 Multiple Satellite Capability 2-11

2-10 Bow Shock and Magnetopause 2-12

2-11 Magnetic Flux Vector - Cone Angle 2-12

2-12 Magnetic Flux Vector - Clock Angle 2-13

2-13 Detectability of Surface Field Strength 2-13

3-1 Simplified Program Logic 3-3

3-2 MAIN Subroutine Flow Logic 3-13

6-1 CACL Flow Diagram 6-22

6-2 FIELDS Flow Diagram 6-32

6-3 FOLDER Flow Diagram 6-39

6-4 GEOPLT Flow Diagram 6-44

6-5 INPUT Flow Diagram 6-49

6-6 MRLC Flow Diagram 6-54

6-7 OTJAZ Flow Diagram 6-63

6-8 OTPLT Flow Diagram 6-71

6-9 PIC2 Flow Diagram 6-77

6-10 PICIN Flow Diagram 6-82

6-11 PSATP Flow Diagram 6-86
6-12 RESO Flow Diagram 6-92
6-13 SMEAR Flow Diagram 6-100
6-14 TCON Flow Diagram 6-105
6-15 TVCHAR Flow Diagram 6-112
6-16 XLIMB Flow Diagram 6-116

Table

2-1 Output Tables 2-4
2-2 Graphic Output 2-4
3-1 Simplified Block Diagram of Presentation Model Operation 3-2
3-2 MAIN Subroutine Source Listing 3-9
3-3 Univac 1108/CDC 6500 Comparisons 3-27
4-1 Definition of Variables 4-2
4-2 Input Variable Identification - Common XZ (97) 4-6
4-3 Input Variable Identification - TV Characteristics 4-9
4-4 Input Variable Identification - Fields and Particles 4-9
4-5 Variable to Tape Identification 4-10
5-1 Comment Cards 5-2
5-2 User Instructions 5-6
6-1 CACL Listing 6-16
6-2 FIELDS Listing 6-26
6-3 FOLDER Listing 6-37
6-4 GEOPLT Listing 6-41
6-5 INPUT Listing 6-46
6-6 MRLC Listing 6-52
6-7 OTJAZ Listing 6-57
6-8 OTPLT Listing 6-66
6-9 PIC2 Listing 6-75

6-10 PICIN Listing 6-79
6-11 PSATP Listing 6-84
6-12 RESO Listing 6-88
6-13 SMEAR Listing 6-94
6-14 TCON Listing 6-103
6-15 TVCHAR Listing 6-107
6-16 XLIMB Listing 6-115
6-17 CIRC Listing 6-118
6-18 FPCHAR Listing 6-120
6-19 IRCHAR Listing 6-121
6-20 NOCHAR Listing 6-122
6-21 OUTS Listing 6-125
6-22 SCROT Listing 6-128
6-23 UVCHAR Listing 6-129

1. INTRODUCTION

This report is a description of "The Outer Planet Data Presentation Computer Program." The program provides visibility into the acquisition of science data for alternate missions; i.e., it could be termed a "Preflight Science Mission Performance Program." The objectives of the computer program were to provide a user with the capability to rapidly evaluate and manipulate mission variables, which include options on:

- Instrument characteristics;
- Instrument pointing and scan;
- Spacecraft;
- Data sample rate;
- Flyby and orbiter trajectories;
- Natural satellites.

Even though these numerous options were established for flexible performance analyses, ease of operation was specified to be of primary importance. Thus, liberal use was made of built-in mission variable data (which can be overridden), minimum number of mandatory inputs, multiple cases per submittal, and graphic plus tabular data presentation.

The program concept originated from the "1975 Venus Multiprobe Mission Study" performed under JPL Contract 952534 in early 1970. This study used "science value functions" to aid in properly allocating mission resources between competitive instruments, descent profiles, planet targeting, and number and configuration of entry probes. For the present purpose the broader mission objectives, mission mode variables and multiple encounters make a subjective value function more difficult to achieve. An alternative conceptual approach envisioned the development of summary science data returned from an entire multiple planet/instrument mission. The first step in this approach is to develop a core program that presents the general mission geometry and instrument data return. The essence of this core program is to provide simple parametric input with proper modeling to enable the user to compare instruments, trajectories, and systems through relevant quantitative data. This became the design criteria for the program described herein.

To accomplish the above, the scope of the development was defined to include construction of the EXECUTIVE (main or core) routine, geometry presentation, TV system modeling, and other instruments as time allowed. The approach consisted of using a series of existing JPL "building blocks" and programming in the common UNIVAC-1108/CDC-6500 computer language area. The use of the building blocks results in a program that has much of its capability being continuously maintained, used, and improved by JPL in conjunction with other associated efforts. These building blocks consist of approximately 70 subroutines, operational on the JPL UNIVAC computer, and are used for vector manipulation, trajectory generation, natural satellite ephemerides, plotting, and other basic data manipulations.

The program was developed on Martin Marietta's CDC 6500 computer. Thus, the common areas in the CDC-6500 UNIVAC-1108 language were used to minimize conversion to the JPL machine (Section 3.3).

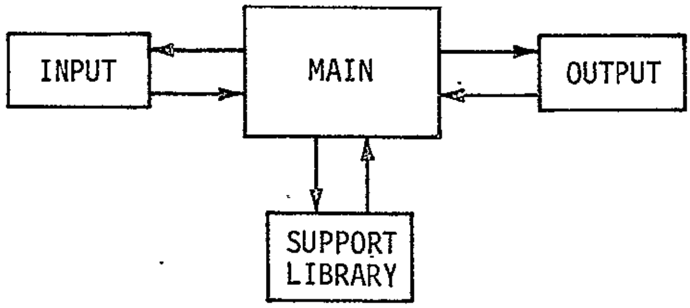
The program is described herein in terms of summary, general overview, flow diagrams, symbol dictionary, operating procedures, and sample cases. Management's interest would most probably include the Introduction, Summary, and the first part of Program Description (Sections 1, 2, 3, and 3.1). These portions of the report yield a general knowledge with the program and its purpose, application, flexibilities, and limitations. The user, at a minimum, should add Model Operating Procedures (Section 5.0), and Appendix A, Volume II. The remaining sections deal with in-depth details and are recommended for the computer-oriented user and programming analyst.

Special recognition is given to the Martin Marietta personnel listed below for their contributions to the development of the program and program documentation.

Bargar, A.	Stoffel, R.
Bullard, S.	Swain, D.
Bynum, M.	Vairin, C.
Carney, P.	Ziehm, R.
Hanson, B.	Froechtenigt, J.
Howard, E.	

2. SUMMARY

The program contains three major sections of responsibility and one support area, as shown below.



The MAIN section governs the total overview and logic flow through the program, i.e., spacecraft, system, instrument, and program control options. MAIN uses the data provided by INPUT for control and initialization, the LIBRARY of subroutines for performing the calculations, and the OUTPUT section for data presentation.

The program is basically for near planet operations (within a planet's sphere of influence). Thus, the trajectory data are obtained via the JPL subroutine ORBIT, a conic propagator. The natural satellite data are supplied from the JPL subroutines JOVSAT, SATSAT, NEPSAT, and URASAT for Jupiter, Saturn, Neptune, and Uranus, respectively. This enables rapid calculation of geometric parameters for flyby or orbiter type missions. The program is capable of processing, with a single input to the computer, multiple encounters, alternate orbits, instrument changes, etc.

To ensure ease of familiarization and use, numerous canned (or default) values were built into the program. The canned values are used by the program if values are not specified by the user; i.e., built in input is used by default. It is suggested that this capability, in conjunction with the minimum input option, be used as a first run to initialize the program. A small number of runs with additional inputs enable the user to rapidly become familiar with the program options, input, and output.

The minimum input option consists of those mandatory inputs that *must* be specified by the user to operate the program and obtain output. The total number of mandatory inputs is 13; nine data, and four "flags." The inputs are as follows:

INPUT CODE NUMBER	INPUT DEFINITION	UNITS
02	Planet number of interest	--
04	Encounter time	Yr/Mo/Day
06	Flag to specify trajectory input vector	--
07 ↓ 12	Trajectory data input	Option Dependent
23	Instrument type desired	--
96	Multiple case indicator	--
98	End of data case flag	--
99	Flag to terminate run	--

With the previous minimum input run in mind, the following options are available through simple flags (alternate values in identical input code numbers) and input data:

1) Spacecraft Identification Options -
   - TOPS - Earth oriented;
   - TOPS - Sun oriented;
   - PIONEER - Spin stabilized.
2) Instrument Pointing Options -
   - Alignment with the radius vector (with automatic repositioning, when the radius vector falls behind the terminator, to the center of the lit crescent);
   - Instrument pointing varied by input cone and clock angles for each measurement.
3) Measurement Timing Options -
   - Linearly spaced intervals set by time between measurements, and a start and stop time;

- Interval determined by the cube of the time from encounter;
- Timing determined by user's input time for cone and clock angles.

4) Output Intervals -
   - Plotting can be output for each measurement or for specified measurement increments;
   - Printing can be output for each measurement or for specified measurement increments, and does not have to be equal to the plot increments.
5) Programmed Instrument Options -
   - Canned values for the characteristics of six different TV systems are included;
   - Canned values for the characteristics of a magnetometer and magnetic fields of Jupiter and Saturn are included;
   - User inputs are available for definition of TV and magnetic field;
   - Inputs are also available to define magnetic pole vector, bow shock, and magnetopause.
6) Trajectory Data Input -
   - Encounter may be either a calendar date or a modified Julian date;
   - Trajectory elements can be input as a classical set, in terms of the arrival asymptote, or in terms of the velocity and position vector at encounter;
   - Orbit variations.

Several options include multiple canned variables, e.g., spacecraft identification and TV systems.

The output, as indicated previously, consists of both tabular (Table 2-1) and graphic (Table 2-2) displays. The tabular data consists of the following three categories:

- One time per case;
- If required;
- Case results.

Table 2-1 Output Tables

1. Input data listing
1. XZ (I) values in storage
1. Instrument characteristics
1. Planet data
1. Computations for minimum picture taking interval
1. Picture number and time
2. Exposure time check and reset
3. Picture taking parameters and quality
3. Satellite view data
3. Resolution data for satellites
3. Smear data for satellites

Note: 1. Printed one time per case
2. Printed if required
3. Printed at specified time increments

Table 2-2 Graphic Output

- Planet and satellite orbits, spacecraft flight path, and vectors to Earth and sun (A)
- Flight path around planet (A)
- Magnetic field
  Cone and clock angles of magnetic flux vectors (F)
- Field of view versus time (planet and satellite) (T)
- Path of planet in spacecraft cone and clock coordinates (size and orientation as seen from S/C) (O)
- Picture resolution at center for (planet, satellite) (O)
- Smear due to spacecraft and target motions (number of pixels smeared) (O)
- Number of pictures versus resolution level (O)
- Range from satellite versus time
  Phase, cone, and clock angles of satellite versus time (O)
- Earth-satellite-spacecraft angle versus time
  Satellite full angle versus time (O)
- Range to planet and satellite versus time (O)
- Planet phase, cone, and clock angle versus time (O)
- Satellite phase, cone, and clock angle versus time (O)

A - An instrument　　T - TV only
F - Fields and particles only　　O - Optional

When the instrument is a TV camera approximately 20 values of the instrument characteristics and the minimum picture taking intervals comprise the data that are calculated in the program instrument and spacecraft data system modeling. The "If Required" tabular print occurs when the TV EXPOSURE and EXPOSURE-TIME are adjusted; the checking and adjustments are accomplished automatically, and the event is printed. The user specifies a desired imaging sample rate, and the program performs the following tests and adjusts the sample rate to the maximum permissible if the tests fail:

- Compares a bit rate requirement versus combination of bulk storage and bit rate capability;
- Compares desired sample rate versus minimum camera frame read time;
- If it is a spinning spacecraft, desired sample rate is compared with rotational rate.

Any required adjustments in the sample rate are printed when they occur and an accumulated number of changes are indicated.

Again for the TV instrument the "Case Results," in general, conform to the plotted data in digital format except for a small number of instrument system parameters such as:

- Exposure Time for Camera (millisec);
- Power Density at Lens (ergs/sq-cm/sec);
- Power Density on Sensor (ergs/sq-cm/sec);
- Exposure (ergs/sq cm).

The Case Results are best illustrated by the graphic data. Since all uses of data could not be included, some examples are discussed in both the geometry and data return areas.

Figure 2-1 illustrates the general encounter overview of Jupiter from a typical 1977 Earth-Jupiter-Saturn-Pluto mission. The "ticks" on the trajectory and satellite orbit plots are 1 hour apart. The satellite names are printed when the trajectory is at its lowest altitude with respect to Jupiter. Thus, at least the following can be gleaned from this single picture:

- Relative time and distance spent at near encounter;
- Potential satellite encounters;
- Potential simultaneous satellite/planet imaging;

- Potential satellite occultations;
- Time spent in possible planet occultation.

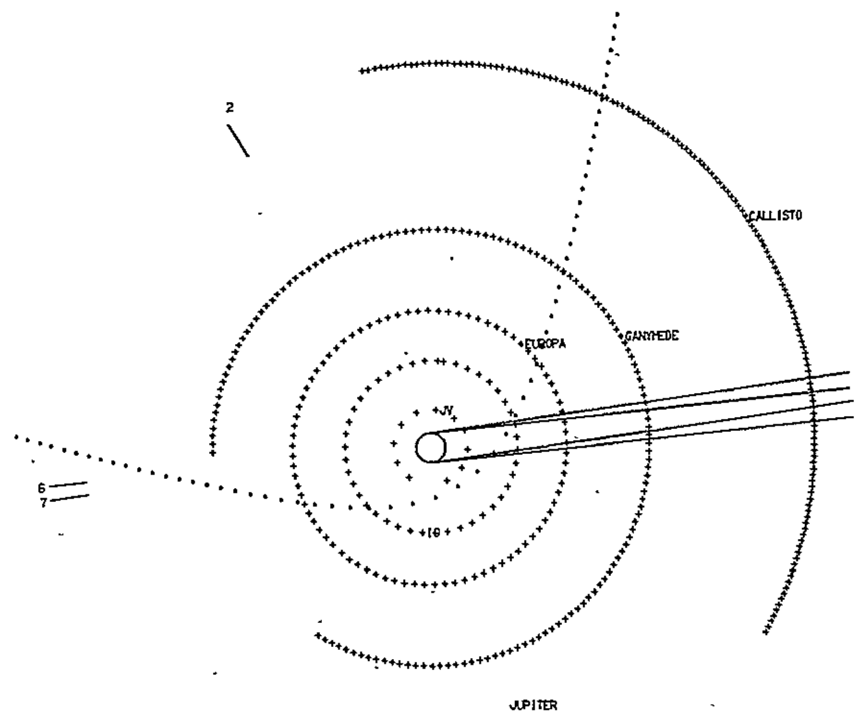


Fig. 2-1 Jupiter Encounter

Figure 2-2 illustrates a Uranus encounter that emphasizes the ecliptic inclination of the rotational axis, and the difficulty in establishing multiple satellite encounters. Figure 2-3, cone and clock angles versus time (which can be correlated with tab output) is associated with the previous Jupiter encounter. The satellite (Europa) was called as a special option. Thus, one can see the similarity in the required scan platform commands. This type of figure can also be used to define desired scan platform limits, determine the effect of specific platform limits, or define scan rates. Figure 2-4 indicates the relative size of the satellite disk to the 1 degree field of view on the planet. Figure 2-5 clearly demonstrates the closest approach to the satellites of Jupiter.

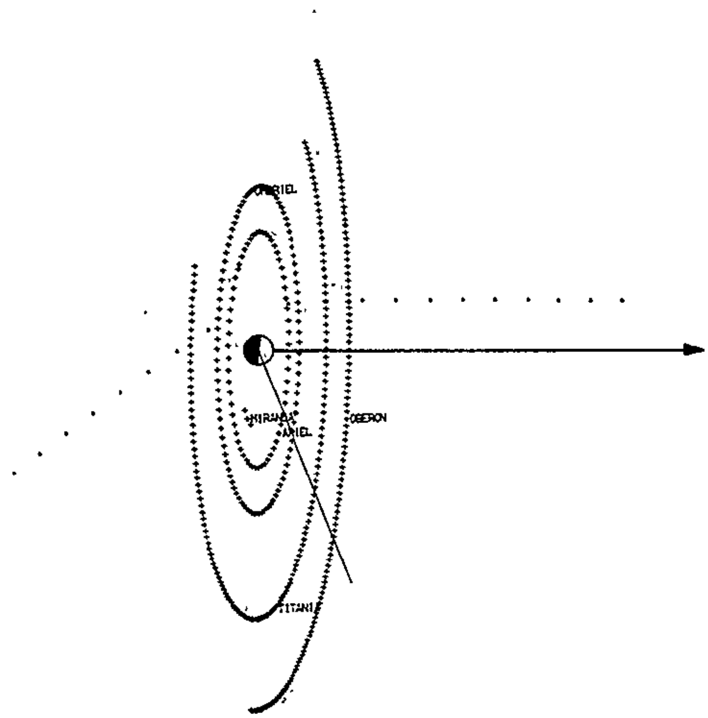


Fig. 2-2 Uranus Encounter

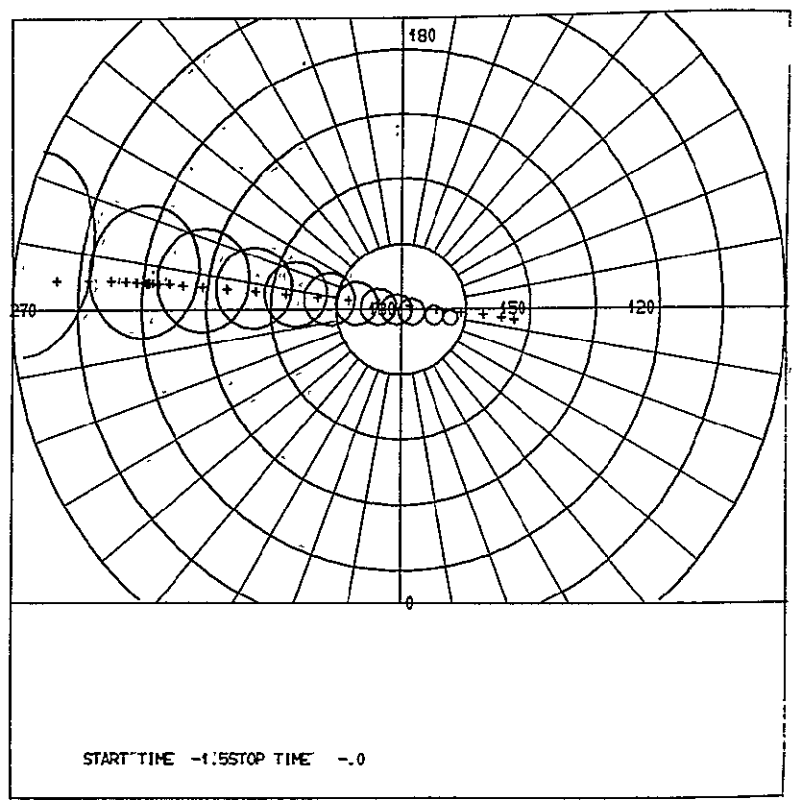


Fig. 2-3 Cone and Clock Angles versus Time

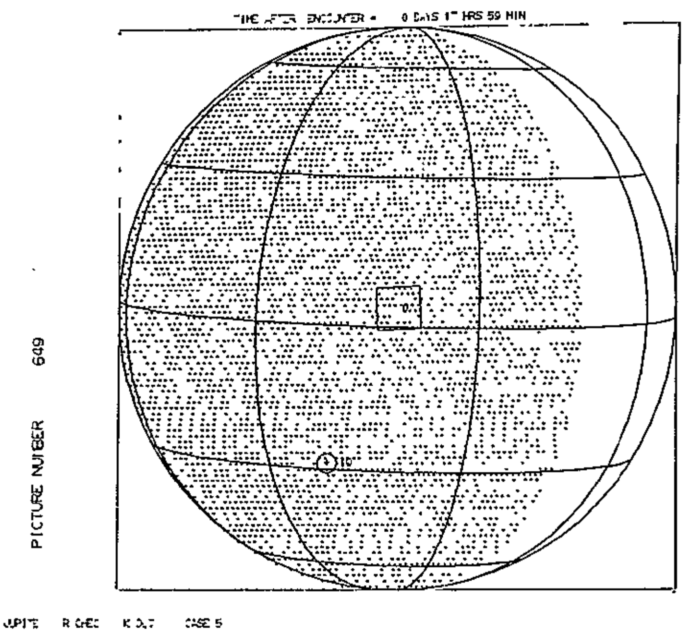


Fig. 2-4 Satellite Disk Size versus 1 Degree Field of View

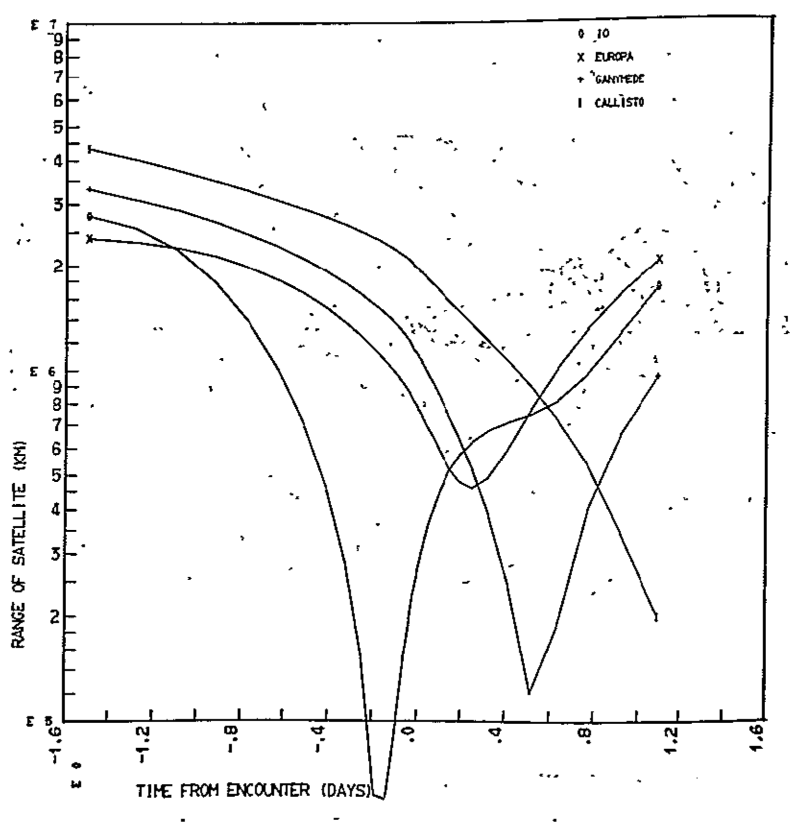


Fig. 2-5 Closest Approach to Jupiter

Figure 2-6 illustrates the number of pictures taken with a resolution greater than three reference values. Direct comparisons between trajectories, spacecraft systems, and TV systems can be made with this summary plot. Figures 2-7 and 2-8 display the resolution versus time for the primary planet and satellites. Timing sequences can be made and trajectory comparisons and mission mode tradeoffs can be evaluated. Figure 2-9 indicates multiple spacecraft capability comparisons through overlays. The spinning spacecraft system yields 135 pictures with resolution greater than 300 km, and the TOPS spacecraft yields 19,200 pictures with resolution greater than 300 km. The maximum planet and satellite resolution is an order of magnitude greater on TOPS than with the spinning spacecraft system. Figure 2-10 contains a view of the trajectory with respect to bow shock and magnetopause. The characteristics of these two phenomena can be altered by input. The trajectory time "ticks" are 1 hour apart, which can be used for timing evaluations. Figures 2-11 and 2-12 are the cone and clock angles of the magnetic flux vector, which can be used to develop or evaluate particle pointing requirements versus time and trajectory. Figure 2-13 shows the surface field strength detectability of a user-specified instrument, which is a parameter that can be used in various direct tradeoffs.

With the general concept in mind, the following improvements are suggested:

1) Expanded Imaging Data -
   - Planet coverage (fraction of planet covered with resolutions better than 300, 100, 50, 10 km per pixel);
   - Number of pictures of specified features (e.g., red spot, latitude bands) with resolutions better than 300, 100, 50, 10 km;
   - Required scan sequence to track satellites when resolution is better than X km.
2) Add a full complement of instruments;
3) Construct a bookkeeping data system;
4) Power requirement bookkeeping;
5) Add an outer loop for summarizing multiple case data;
6) Develop probe trajectory capability with attendant data link;

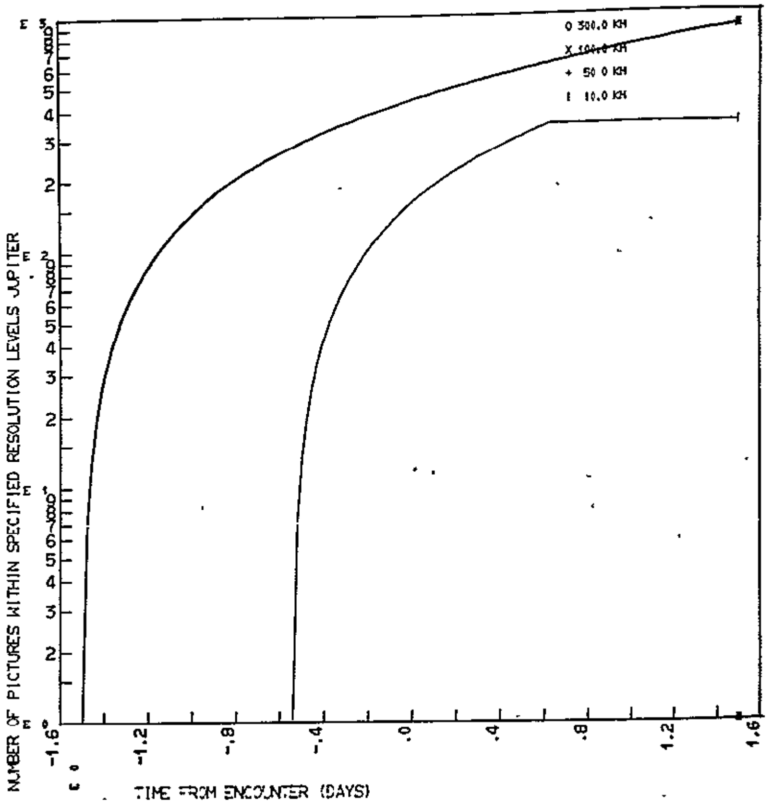


Fig. 2-6 Number of Pictures Taken (resolution greater than stated reference values)

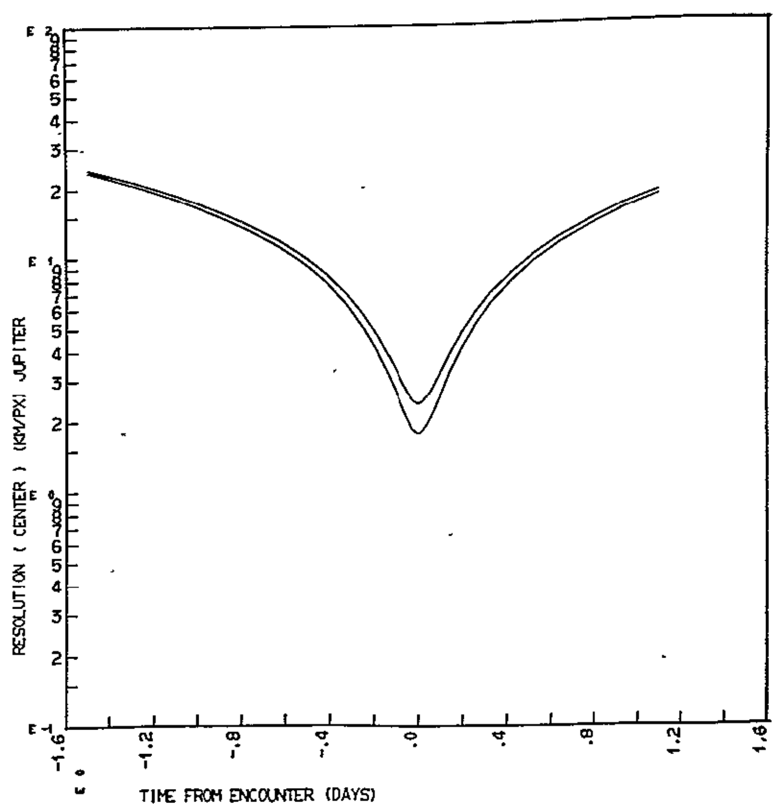


Fig. 2-7 Resolution versus Time

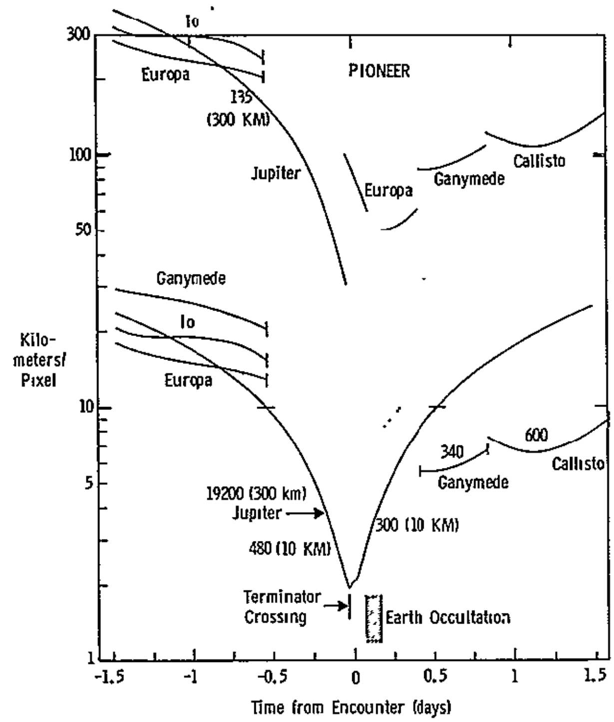


Fig. 2-8 Resolution versus Time

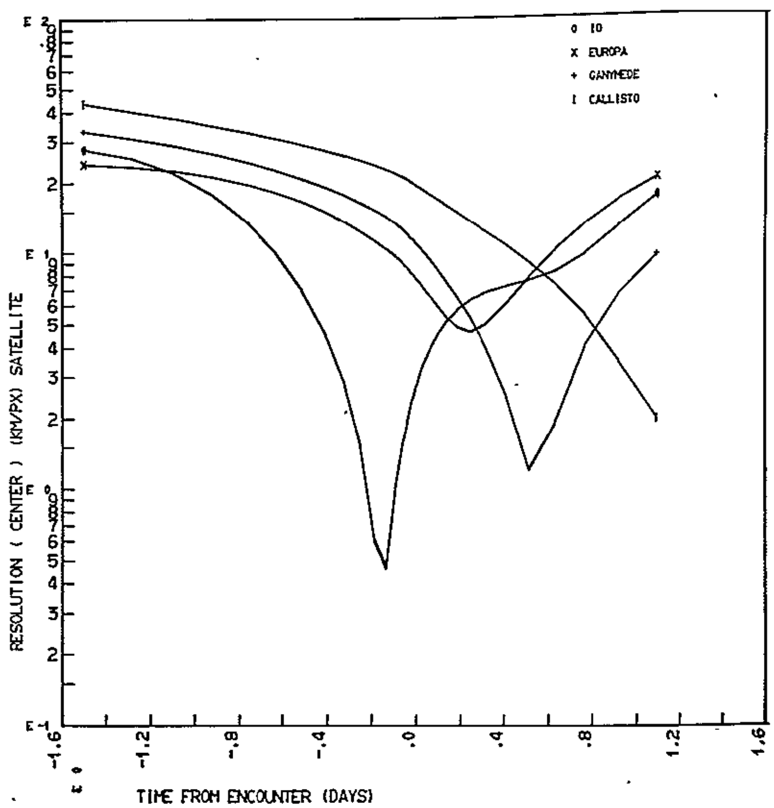


Fig. 2-9 Multiple Spacecraft Capability.

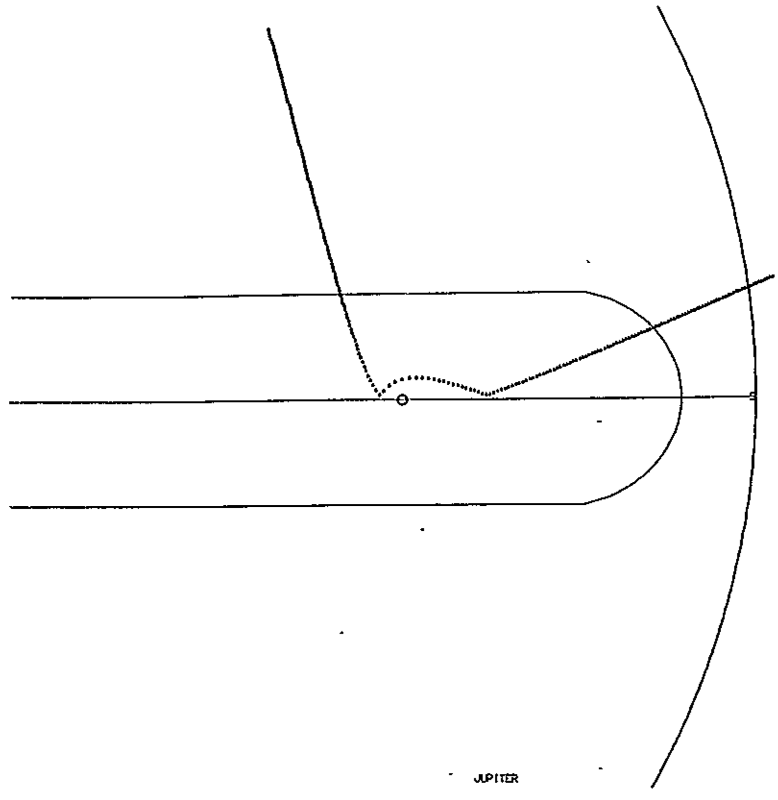


Fig. 2-10 Bow Shock and Magnetopause

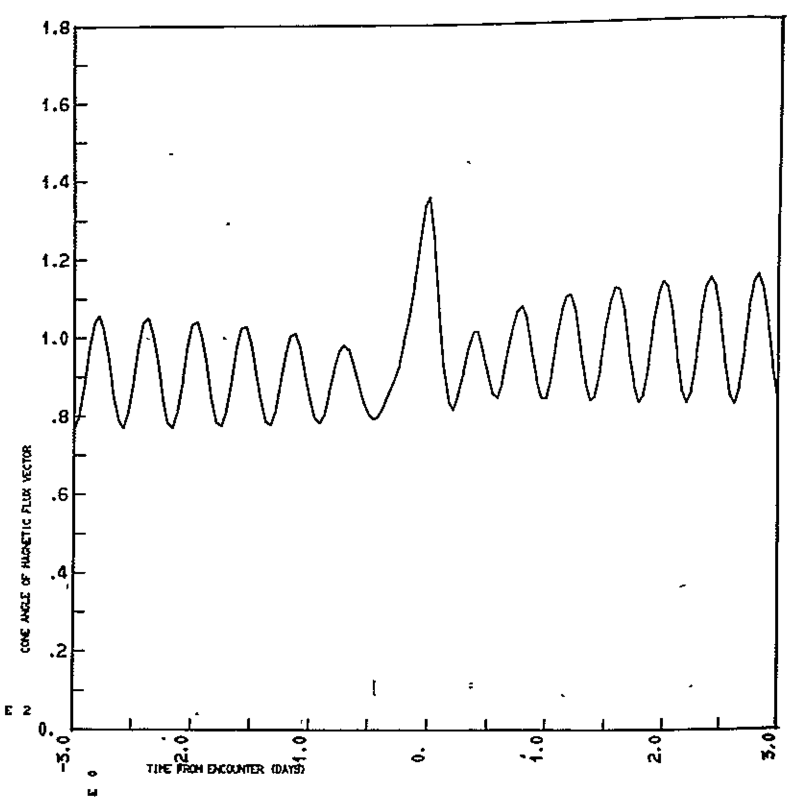


Fig. 2-11 Magnetic Flux Vector - Cone Angle

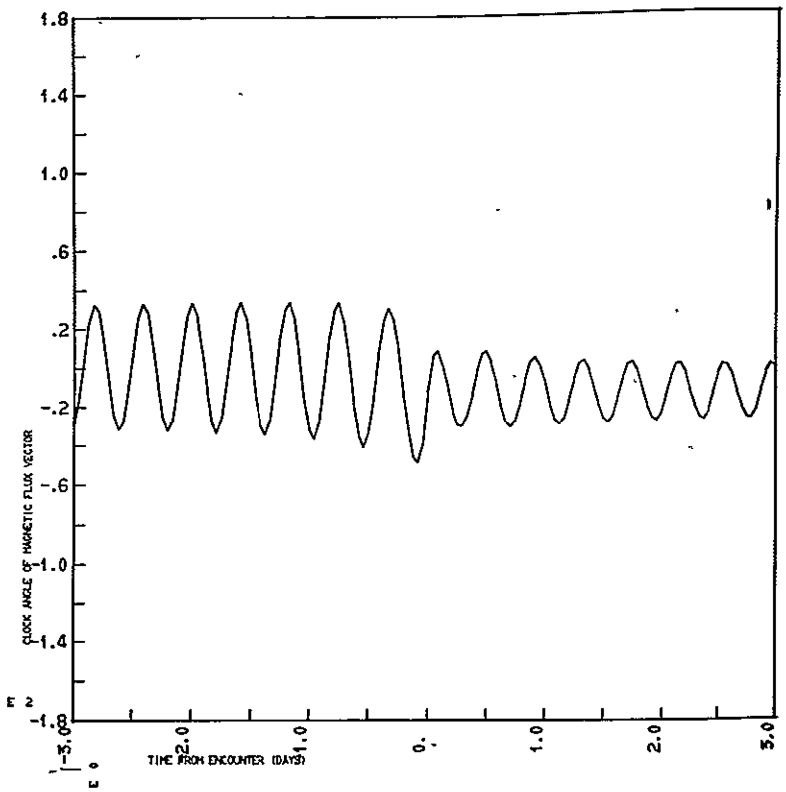


Fig. 2-12 Magnetic Flux Vector - Clock Angle

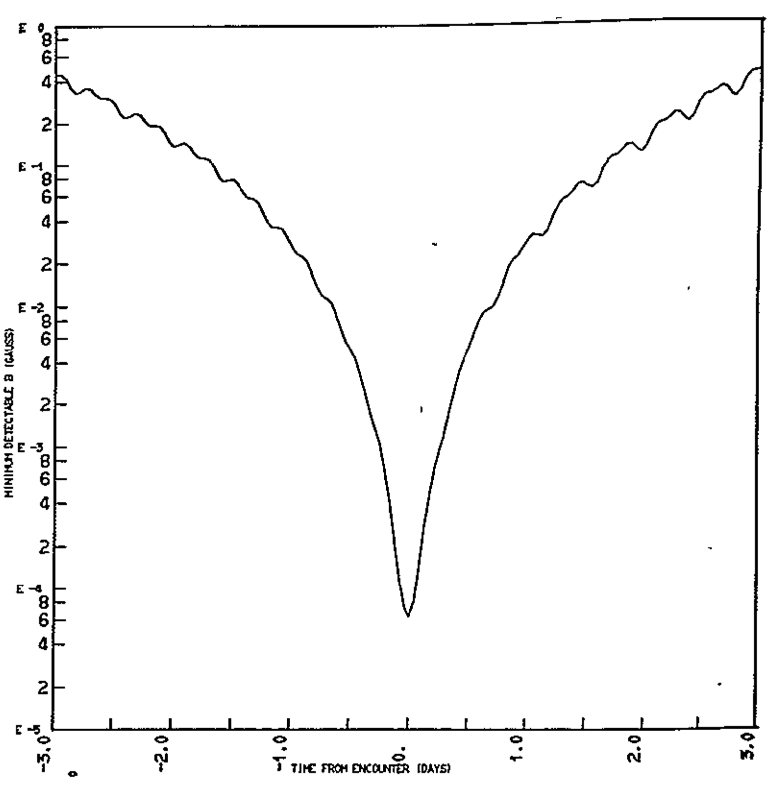


Fig. 2-13 Detectability of Surface Field Strength

7) Expand instrument design capability;

8) Incorporate iterative loop for maximizing desired parameters through alternate trajectories and/or instrument characteristics;

9) Incorporate a more accurate trajectory propagator for selected usage.

In conclusion, the program represents a powerful tool. However, it represents a small step in automating the "Science Mission Performance" evaluation capability.

3. PROGRAM DESCRIPTION

The Outer Planet Data Presentation Computer Program consists of an executive driver program (MAIN) and some 105 subroutines that perform many different functions. Some of the subroutines perform specific calculations dealing with a given science instrument; others are vector manipulators; several produce one or more plotted outputs; two are concerned with the reading of input cards and the tabulation of output data (Section 6). This section describes how these many subprograms fit together.

The primary function of the model is to display, in graphic and tabular form, the science data return that can be expected from a given instrument when carried past a planet by a spacecraft on a particular orbit or trajectory. This display provides a measure of the data's quantity, as well as its quality. In addition, a limited number of design parameters are computed, such as TV exposure times and lens f/stop. Table 3-1 shows the relationship between the inputs, the program subroutines, and the various outputs.

3.1 General Program Logic (Overview) - The model may be divided into three sections. In the first section, the model is initialized, input data is read in, a trajectory overview is plotted, and a timing sequence for the measurements to be taken by the instrument is determined. The second section performs the calculations, plots, or data tabulations associated with each measurement of interest; the third section provides summary output plots of the calculations already performed.

Figure 3-1 displays these functions in a slightly more detailed fashion. As shown by the figure, the route through the model is prescribed by the type of instrument carried by the spacecraft, and five alternatives are shown whereas there are several other obvious instrument possibilities. During the course of the contract, the effort was directed toward a rather complete treatment of the TV imaging instrument. It was the intention during the contract to provide the capability of adding other instruments in future developments.

Block 1 (Initialize Program) rewinds the output tapes and sets counters to zero. Block 2 (Read Input Data) reads the mission input data (Section 5).

Table 3-1 Simplified Block Diagram of Presentation Model Operation

INPUT DATA	DATA PRESENTATION MODEL	OUTPUT DATA
Planet Number • Specified Satellite Number • Encounter Date • Instrument Characteristics • Model Operation Flags and Intervals - Instrument Pointing Mode; Measurements Start and Stop Times; Measurements per Calculation; Calculations per Plot; Calculations per Printout; Planet/Satellite Run Flag; Minimum Switch Range; Inhibit Switch to Satellite Start and Stop Times; Summary Plot Flags.	• Executive Driver - MAIN • Instrument Characteristics and Properties Routines • Planet, Satellite, and Spacecraft, Position Subroutines • Vector Utility Subroutines • Plotting Subroutines • Tabulation Subroutines	• Tabulation of Input Data • Plots and Tabulations Relative to Individual Measurements • Plots Showing Overview • Plots Summarizing Expected Instrument Data Parameters versus Mission Time

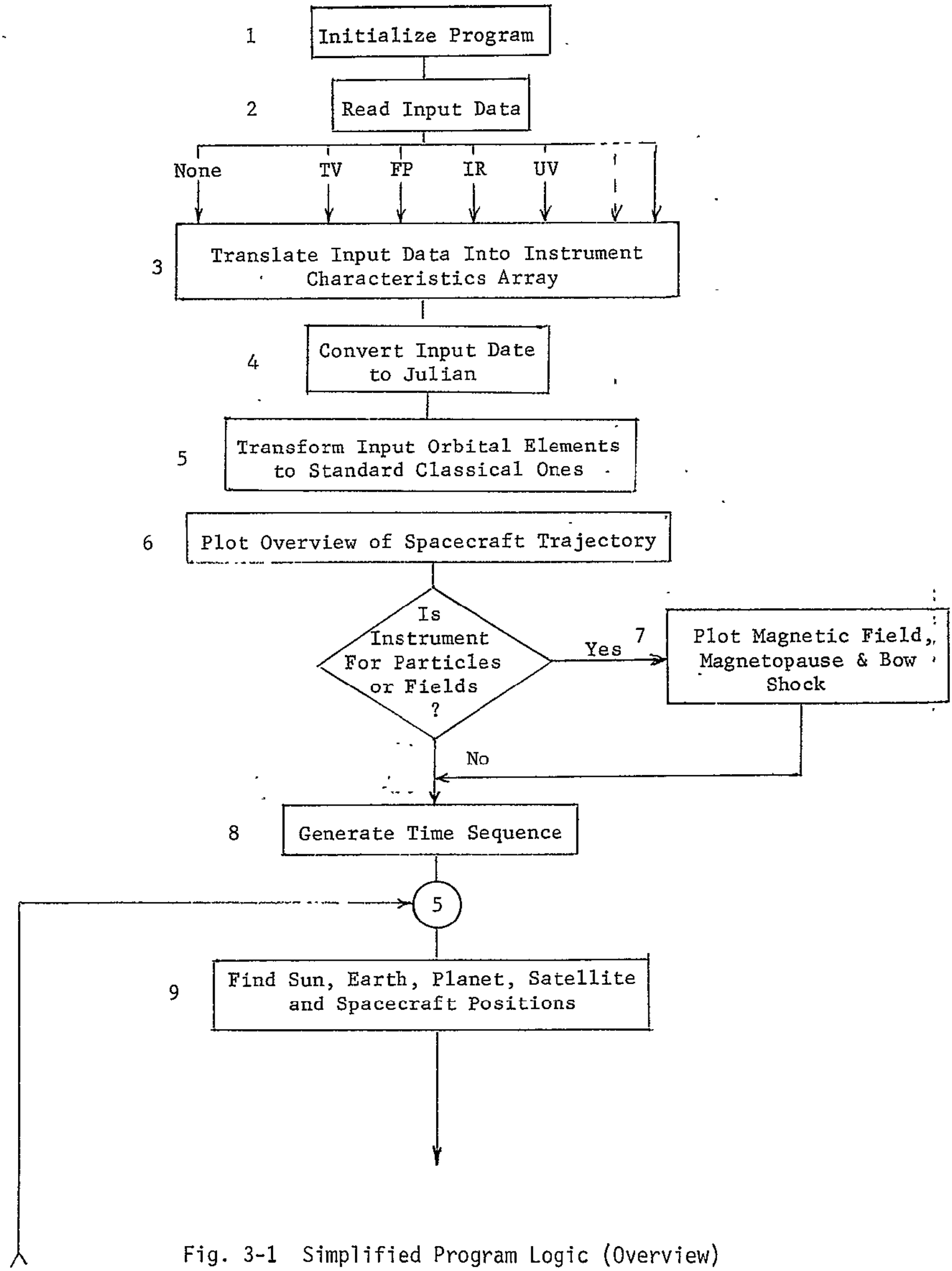


Fig. 3-1 Simplified Program Logic (Overview)

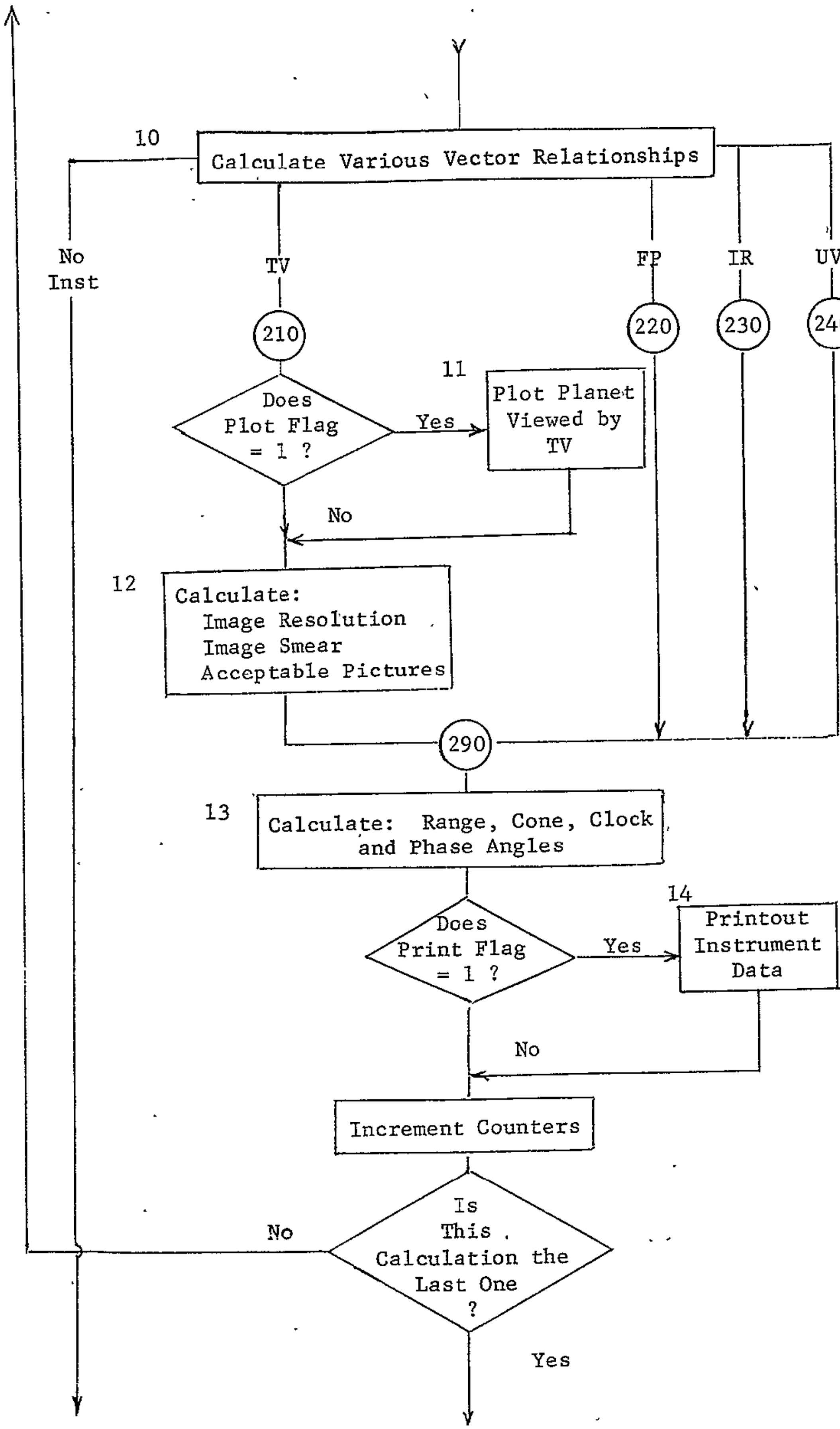

10
Calculate Various Vector Relationships
No Inst
TV
FP
IR
UV
210
220
230
240
11
Does Plot Flag = 1 ?
Yes
Plot Planet Viewed by TV
No
12
Calculate:
Image Resolution
Image Smear
Acceptable Pictures
290
13
Calculate: Range, Cone, Clock and Phase Angles
14
Does Print Flag = 1 ?
Yes
Printout Instrument Data
No
Increment Counters
Is This Calculation the Last One ?
No
Yes

Fig. 3-1 (cont)

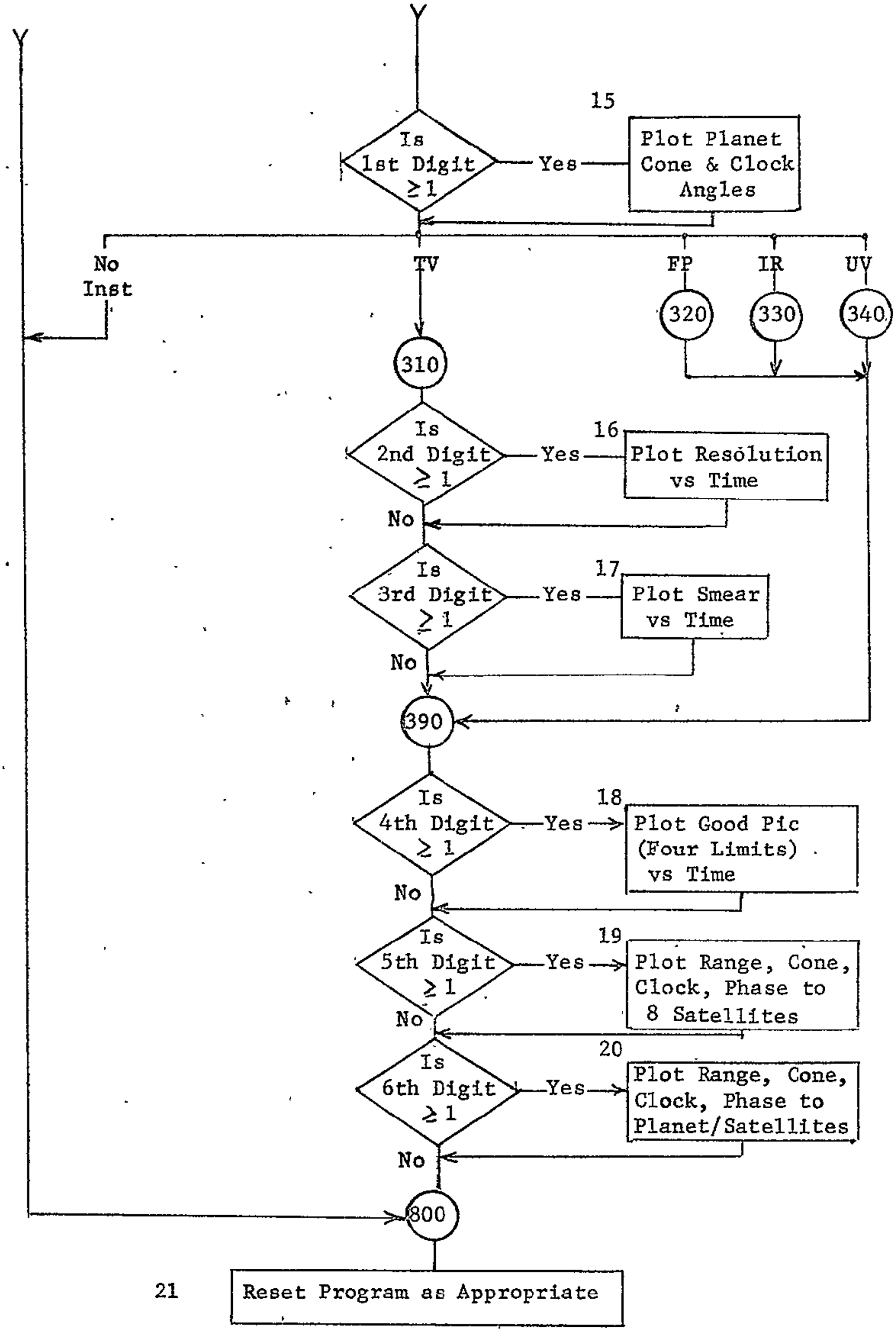

15
Is 1st Digit ≥1
Yes
Plot Planet Cone & Clock Angles
No Inst
TV
FP
IR
UV
320
330
340
310
16
Is 2nd Digit ≥ 1
Yes
Plot Resolution vs Time
No
17
Is 3rd Digit ≥ 1
Yes
Plot Smear vs Time
No
390
18
Is 4th Digit ≥ 1
Yes
Plot Good Pic (Four Limits) vs Time
No
19
Is 5th Digit ≥ 1
Yes
Plot Range, Cone, Clock, Phase to 8 Satellites
No
20
Is 6th Digit ≥ 1
Yes
Plot Range, Cone, Clock, Phase to Planet/Satellites
No
800
21
Reset Program as Appropriate

Fig. 3-1 (concl)

Once the input data has been read, the next set of functions is indicated by Block 3, which translates a given portion of the input or canned numbers into specific meanings with regard to the mission's instrument. For example, input card 36 with a value of XZ(36) is translated to a variable P(6), which is the focal length of the TV lens or is the magnetopause range if the instrument is measuring fields.

Block 4 converts the input calendar date into a Julian date, while Block 5 provides manipulations to convert the input trajectory elements (cards 07 through 12) into a classical set, which consists of the semimajor axis, the semilatus rectum, the argument of the ascending node, the longitude of periapsis, the inclination of the orbit, and the orbit eccentricity.

In Block 6, the relationship of the spacecraft trajectory, the position of the planet's satellites, the direction of the sun (marked with a 7), the direction of the Earth (marked with a 6), and the antiperiapsis direction (marked with a 2) can be seen. The position of each satellite at the time of the spacecraft's periapsis is marked with the name of the satellite, which enables tracing the relative hourly positions of the spacecraft and satellite. Similarly, when the instrument measures particles or fields, the orbital position of the spacecraft is plotted with respect to the planet's magnetic lines of force, and a second plot shows spacecraft position with respect to the planet's magnetopause and bow shock line. These plots are shown in Block 7.

In Block 8, the time sequence for the measurements conducted by the instrument is generated. This sequence can take three forms: a linear one dictated by the inputs of start, stop, and delta time for the instrument; a time sequence that provides an optimum plot of the cone and clock angles of the planet during encounter; or a set of measurement times input by the user.

Starting with Block 9 and extending through Block 14, the calculations, printouts, and plots are performed with regard to a specific measurement of the instrument and correspond to one of the sequence times generated in Block 8. Block 9 calculates many vector relationships between the sun, Earth, planet, spacecraft, instrument pointing, and the various satellites of the planet, which are used in other portions of the program.

The program then takes one of the five possible paths that are instrument dependent. In the case of TV, the flow is to Block 11 with the plot flag equal to 1, as it will every N^{th} calculation, N being an optional input. With no input, N is set to 1 and Block 11, which plots the view as seen by the TV, is entered on each calculation.

Block 12 contains the subroutines required to calculate the resolution of the image, the image smear, and the number of pictures that have a resolution better than four input levels. The results of these calculations are sequentially stored on tapes for later readout, to produce summary type plots. In Block 13, further geometric parameters describing the range and angular relationships of the planet and its satellites are stored on tape.

When the print flag equals 1 (every M^{th} calculation), suitable output data are tabulated for that measurement (Block 14). With this tabulation, the program returns to statement number 5 to consider the next measurement.

When all measurements have been processed, the program is governed by an input six digit number stored by card 90. When no card 90 is read, all six digits are set to zero by default and no further plotting is accomplished. If any of the digits are set to 1 by card 90, the plotting Blocks 15 through 20 are called on to produce the desired plots, as shown below:

1 1 1 1 1 1

- 6th digit (1) → Plot planet cone and clock angle position in Block 15
- 5th digit (1) → Plot TV resolution plots (Block 16)
- 4th digit (1) → Plot TV image smear plots (Block 17)
- 3rd digit (1) → Plot number of pictures having resolutions better than four levels
- 2nd digit (1) → Plot range, cone, clock, phase, and half angles for eight possible satellites as well as time to switch
- 1st digit (1) → Plot range, cone, clock and phase angles to planet and specified satellite.

After the last summary plot is completed, the program terminates or recycles with new data (as indicated in Block 21 and governed by input cards 96, 98, and 99).

3.2 MAIN Program Logic and Flow Diagram - MAIN calls for the basic set of input data to be entered into computer storage. It then manipulates the main thread of the calculations by calling up other subroutines as required to obtain a complete set of data points, which are then stored on tapes, to be used in generating plots and output tables.

Table 3-2 presents the MAIN Fortran statements; Figure 3-2 presents the MAIN logic flow. Since MAIN calls other subroutines to perform various calculations, a brief statement of the operation performed by the called subroutine is contained in the figure. In many cases a subroutine called by MAIN will call other subroutines. The flow diagram illustrates second and third levels of subroutine operations.

Table 3-2 MAIN Subroutine Source Listing

```
      PROGRAM MAIN(INPUT,OUTPUT,FILMPL,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7,
     *TAPE8,TAPE9,TAPE10,TAPE11,TAPE12)
C
C                                                        MAIN-2
C
      COMMON/HEDING/TITLE(13)
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      REAL JDE(3),JDV
      COMMON/JDAYS/JDV,TT
      COMMON /PLCONS/GMS,GMPL(12),RS,RPL(12),SNAME(2),PLNAME(2,12),
     X SPV(3),PLPV(3,12),SVE(3),PLVE(3,12),SROT,PLROT(12),SOBL,PLOBL(12)
      COMMON/REVER/ROTW(3)
      COMMON/CHARAC/P(60)
      COMMON/FOROUT/DUMMYB(13),IPCN,DUMMYC(6)
      COMMON/CELEST/C(3),S(3),SN(3)
      COMMON/STUFF/XN(3),DUMMF(110)
      COMMON/STUF1/XSAT(3)
      COMMON/GEOM/XE(3),XP(3)
      COMMON/CAM/CCONE(200),CCLOCK(200)
      COMMON/TEST/ITEST(97)
      COMMON/FLAGS/IFLT,IFLPL,IFLPR
      COMMON/TCONS/TCONS(200)
      COMMON /TRA/TRACE,DUMMYA(13)
      COMMON/XZ/XZ(97) /VAR/VAR(97)
      COMMON/FLYBY/DUM1(6),X(3),DUM2(9)
      DATA C/-.06034330,.23724160,-.969957469/
      FLAG(A,B)=AMOD(A,B)
C
C         INITIALIZATION AND ONE TIME CALCULATIONS
C
 1000 CONTINUE
      REWIND 7
      REWIND 8
      REWIND 9
      REWIND 10
      REWIND 11
      REWIND 12
      RPD=3.14159/180.0
       I=1
      NREC=1
      NCB=0
      IFLST=0
      CALL INIT280
      CALL PLANET(5HNOTRA)
      CALL INRUT(NP)
      IF(XZ(85).NE.0) IFLST=1
      DATE=XZ(4)
      IP=XZ(2)
```

Table 3-2 (cont)

```
      IVS=XZ(3)
      NSC=XZ(18)
      CALL CPLANN(IP)
      CALL SCROT(ROTW)
C
C  RETRIEVE INSTRUMENT CHARACTERISTICS
C
      INST=1
      IF(ITEST(22).NE.0) INST=XZ(22)+0.01
      INSTR=INST/100 + 1
      GO TO (10,11,12,13,14),INSTR
   10 CALL NOCHAR(INST)
      GO TO 20
   11 CALL TVCHAR(INST)
      GO TO 20
   12 CALL FPCHAR(INST)
      GO TO 20
   13 CALL IRCHAR(INST)
      GO TO 20
   14 CALL UVCHAR(INST)
C
C  SET UP TIME SEQUENCING FOR CALCULATIONS
C
   20 CONTINUE
 2    IF(DATE.GE.0) CALL CALNDR(JDE,2HJD,DATE,5HCAL19)
      IF(DATE.LT.0) CALL CALNDR(JDE,2HJD,-DATE,5HMODJD)
      DAYERS=DATE-INT(DATE)
      JDE=JDE+DAYERS
      CALL ORBIN(XZ(5),3)
      SAVE=JDV
      JDV=JDE
      CALL PSATP
      JDV=JDE
      IF(INSTR.EQ.3) CALL FIELDS
      JDV=SAVE
      IF(INSTR.EQ.3) GO TO 800
      CALL TCON(NP)
      INTPL=1
      INTPR=1
      IF(ITEST(81).EQ.1) INTPL=XZ(81)
      IF(ITEST(82).EQ.1) INTPR=XZ(82)
      RAPER=RPL(IP)
      IF(ITEST(85).EQ.1) CALL PSCONS(RAPER,6HRADIUS,IVS,IP)
      NPC=P(39)
C
C        START LOOP
C
 5    JDV=JDE+TCONS(I)
      IFLPL=0
      IFLPR=0
      IF((FLOAT((I-1)/INTPL)).EQ.(FLOAT(I)-1.0)/FLOAT(INTPL)) IFLPL=1
      IF((FLOAT((I-1)/INTPR)).EQ.(FLOAT(I)-1.0)/FLOAT(INTPR)) IFLPR=1
```

Table 3-2 (cont)

```
      CALL TCONV(TT,3HSEC,TCONS(I),3HDAY)
      IF(NSC.GE.0) CALL PLPOS(XE,JDV,3)
       CALL PLPOS(XP,JDV,IP)
 8    IF(IVS.GT.0) CALL PLASAT(XSAT,JDV,IVS,IP)
      CALL ORBPOS(X,TT)
      CALL VEQUAL(XN,X)
      TX=TT/86400.
      IPCN=(I-1)*NPC+1
      TMAGX=ABSV(X)
      TANGLE=ASIN(RAPER/TMAGX)*RPD
      IF(ITEST(83).EQ.1.AND.TMAGX.GT.XZ(83)) IFLPL=IFLPR=0
      IF(ITEST(84).EQ.1.AND.TANGL.LT.XZ(84)) IFLPL=IFLPR=0
  50   CONTINUE
      CALL VCOMB(S,XP,-1.0,X,-1.0)
      CALL CACL
      GO TO (800,210,220,230,240),INSTR
  210 CONTINUE
C
C        TV ROUTINES
C
      IF(IFLPL.EQ.1) CALL PIC2(P(44),P(45))
      CALL RESO(4HCALC)
      CALL SMEAR(4HCALC)
      CALL MRLC(4HCALC)
      GO TO 290
  220 CONTINUE
C
C        FP ROUTINES
C
      GO TO 290
  230 CONTINUE
C
C        IR ROUTINES
C
      GO TO 290
  240 CONTINUE
C
C        UV ROUTINES
C
  290 CONTINUE
      CALL OTJAZ(4HCALC)
      IF(IFLPR.EQ.1) CALL OUTS
       I=I+1
      NREC=NREC+1
      IF(I.LT.NP+1) GO TO 5
C
C        END LOOP
```

Table 3-2 (concl)

```
      ZOUT=XZ(90)
      IF(FLAG(ZOUT,10.).GT.0.) CALL GEOPLT
      CALL FRAME
      GO TO (800,310,320,330,340),INSTR
  310 CONTINUE
C
C        TV ROUTINES
C
      IF(FLAG(ZOUT,100.).GE.10.)  CALL RESO(4HPLOT)
      IF(FLAG(ZOUT,1000.).GE.100.) CALL SMEAR(4HPLOT)
      GO TO 390
  320 CONTINUE
C
C        FP ROUTINES
C
      GO TO 390
  330 CONTINUE
C
C        IR ROUTINES
C
      GO TO 390
  340 CONTINUE
C
C        UV ROUTINES
  390 CONTINUE
      IF(FLAG(ZOUT,10000.).GE.1000.) CALL MRLC(4HPLOT)
      IF(FLAG(ZOUT,100000.).GE.10000.) CALL OTJAZ(4HPLOT)
      IF(FLAG(ZOUT,1000000.).GE.100000.) CALL OTPLT
  800 CONTINUE
      IF(ITEST(96).EQ.0) GO TO 1000
      IF(XZ(96).EQ.0.0) GO TO 1000
      IF(XZ(96).NE.1.0) CALL STERM
      DO 2000 I=1,96
      ITEST(I)=0
      XZ(I)=0.0
 2000 CONTINUE
      IFLST=0
      NSC=0
      NCB=0
      NCODE=0
      IFLT=0
      IFLPL=1
      IFLPR=1
      GO TO 1000
  900 FORMAT(3E25.8)
      END
```

```
      PROGRAM MAIN(INPUT,OUTPUT,FILMPL,TAPE5=INPUT,TAPE6=OUTPUT,TAPE7,
     *TAPE8,TAPE9,TAPE10,TAPE11,TAPE12)
C
C                                                          MAIN-2
C
      COMMON/HEDING/TITLE(13)
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      REAL JDE(3),JDV
      COMMON/JDAYS/JDV,TT
      COMMON /PLCONS/GMS,GMPL(12),RS,RPL(12),SNAME(2),PLNAME(2,12),
     X SPV(3),PLPV(3,12),SVE(3),PLVE(3,12),SROT,PLROT(12),SOBL,PLOBL(12)
      COMMON/REVER/ROTW(3)
      COMMON/CHARAC/P(60)
      COMMON/FOROUT/DUMMYB(13),IPCN,DUMMYC(6)
      COMMON/CELEST/C(3),S(3),SN(3)
      COMMON/STUFF/XN(3),DUMMF(110)
      COMMON/STUF1/XSAT(3)
      COMMON/GEOM/XE(3),XP(3)
      COMMON/CAM/CCONE(200),CCLOCK(200)
      COMMON/TEST/ITEST(97)
      COMMON/FLAGS/IFLT,IFLPL,IFLPR
      COMMON/TCONS/TCONS(200)
      COMMON /TRA/TRACE,DUMMYA(13)
      COMMON/XZ/XZ(97) /VAR/VAR(97)
      COMMON/FLYBY/DUM1(6),X(3),DUM2(9)
      DATA C/-.06034330,.23724150,-.96957469/
      FLAG(A,B)=AMOD(A,B)
```

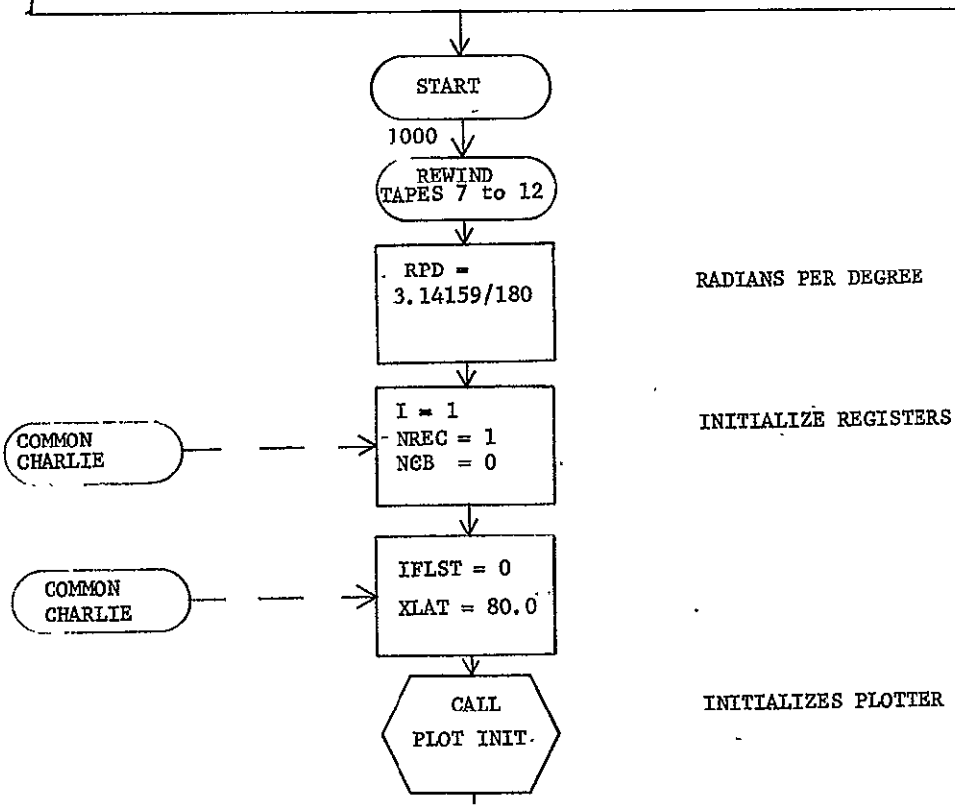
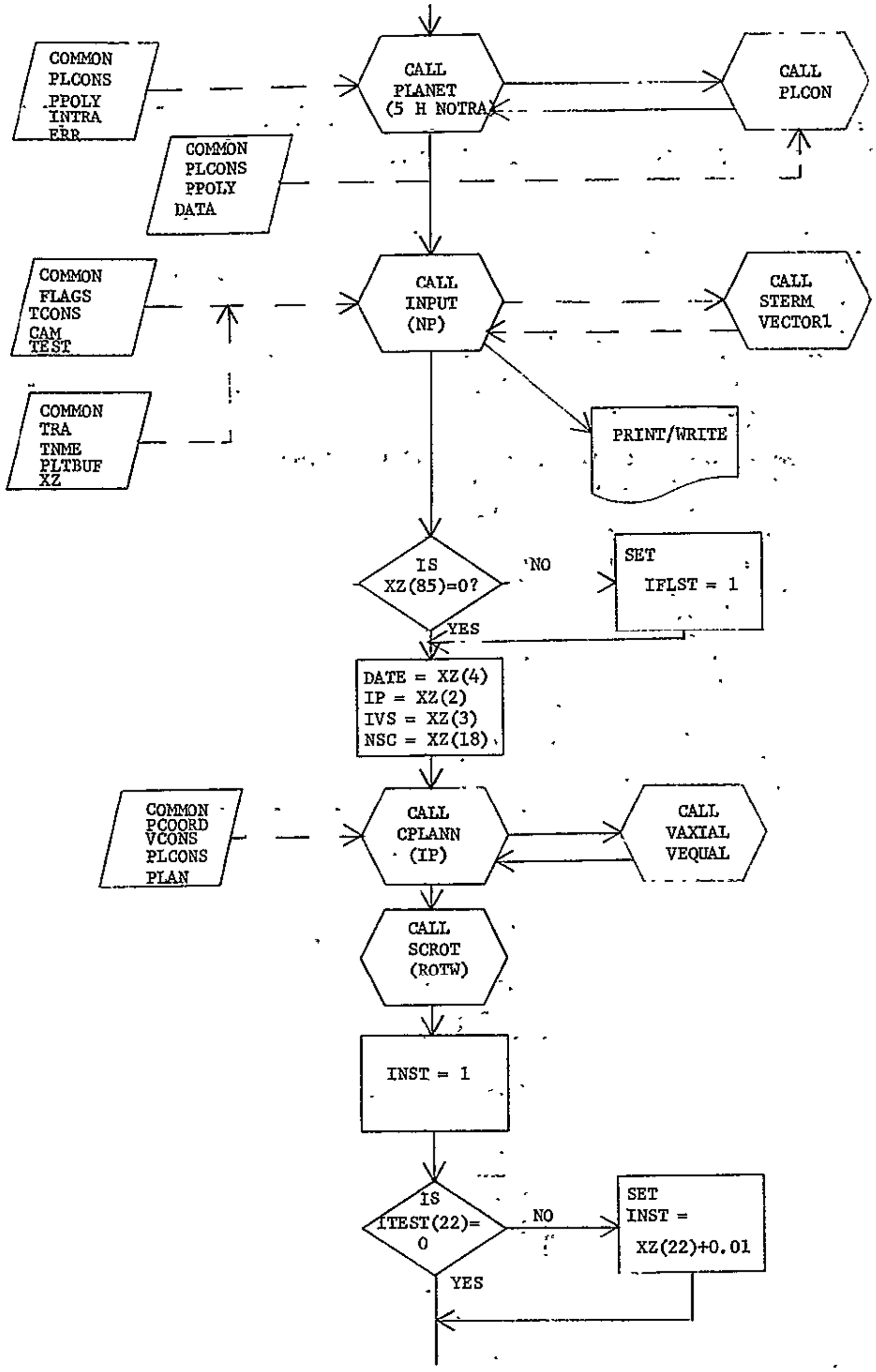


Fig. 3-2 MAIN Subroutine Flow Logic



PLANET - COMPUTES POSITIONS, VELOCITIES, ACCELERATIONS AND ROTATION OF THE SUN AND PLANETS.

PLCON - CONTAINS ALL THE PLANETARY CONSTANTS FOR PLANETS

INPUT - READS IN AND OUTPUTS DATA RELATIVE TO BASE BEING RUN.

STERM - PROVIDES AN ALTERNATE METHOD TO TERMINATE RUN.

VECTOR1 - A SET OF SUBROUTINES THAT PERFORM VECTOR MANIPULATIONS.

PRINT/WRITE - OUTPUTS LIST OF INPUTS

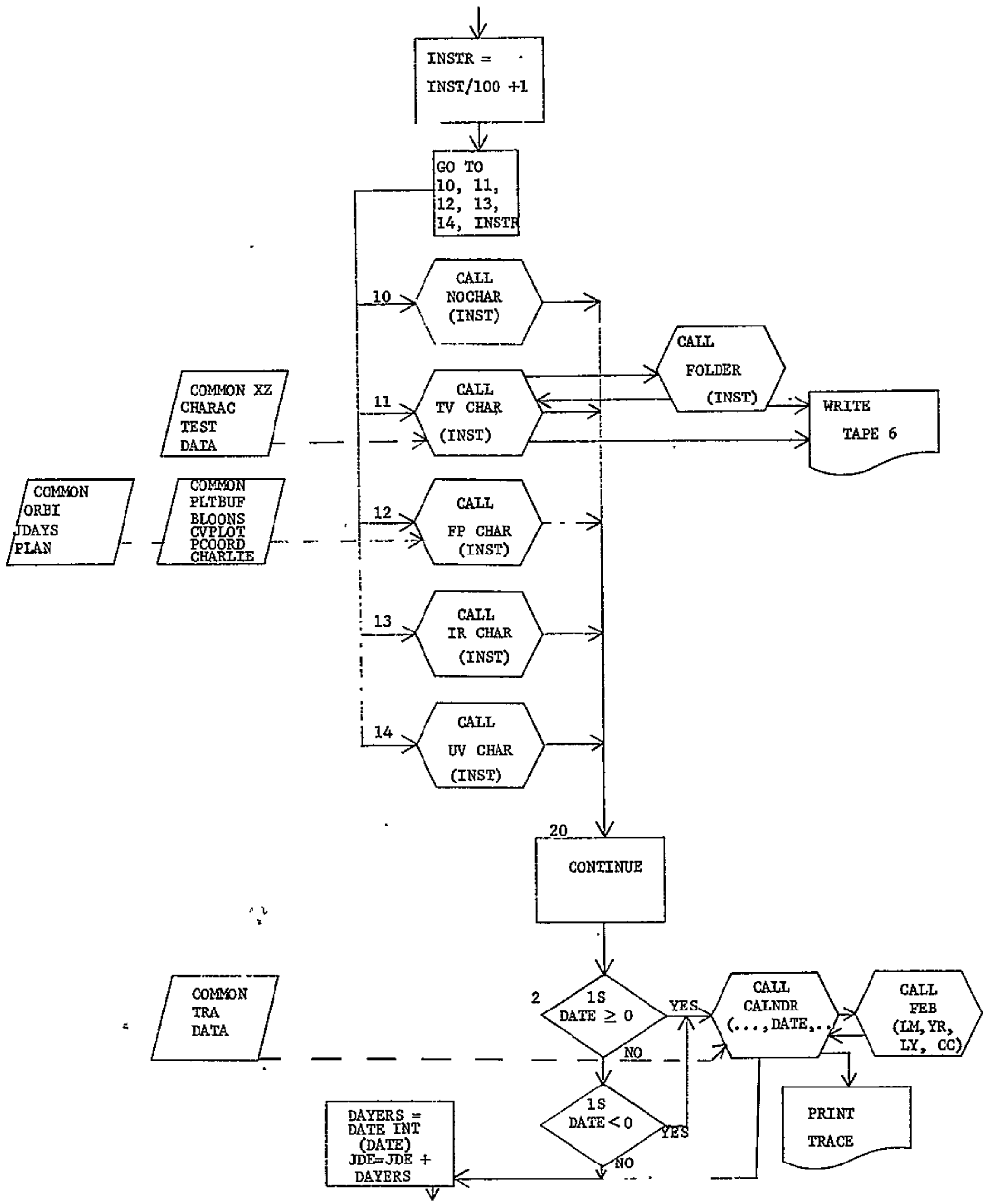
CPLANN - TRANSFORMS POSITION AND VELOCITY VECTORS

VAXIAL - GENERATES UNIT VECTORS ALONG AXES

VEQUAL - SETS ONE VECTOR EQUAL TO ANOTHER

SCROT - PROVIDES SPACECRAFT ROTATION RATES FROM INPUT VALUES.

Fig. 3-2 (cont)



NOCHAR - OPTION FOR NO INSTRUMENTATION

TVCHAR - TO PROVIDE A MEANING TO THE BLOCK OF INPUT VARIABLES ASSIGNED TO THE INSTRUMENT. WHENEVER SPECIFIC INFORMATION IS NOT SUPPLIED BY DATA INPUTS, CANNED VALUES DESCRIBING SIX ALTERNATE CAMERAS ARE PROVIDED. A SECOND BLOCK OF TV VARIABLES ARE COMPUTED BY THE SUBROUTINE AND THE TOTAL SET OF VARIABLES IS PRINTED OUT.

FOLDER - CALCULATES AND WRITES ON TAPE 6 ALBEDO AND ENERGY CHARACTERISTICS

WRITE - WRITES PICTURE CHARACTERISTICS ON TAPE 6

FPCHAR - TO ASSIGN A BLOCK OF INPUT VARIABLES TO VARIOUS FIELDS AND PARTICLES.

IRCHAR - TO ASSIGN A BLOCK OF INPUT VARIABLES TO VARIOUS INFRA RED IMAGERS

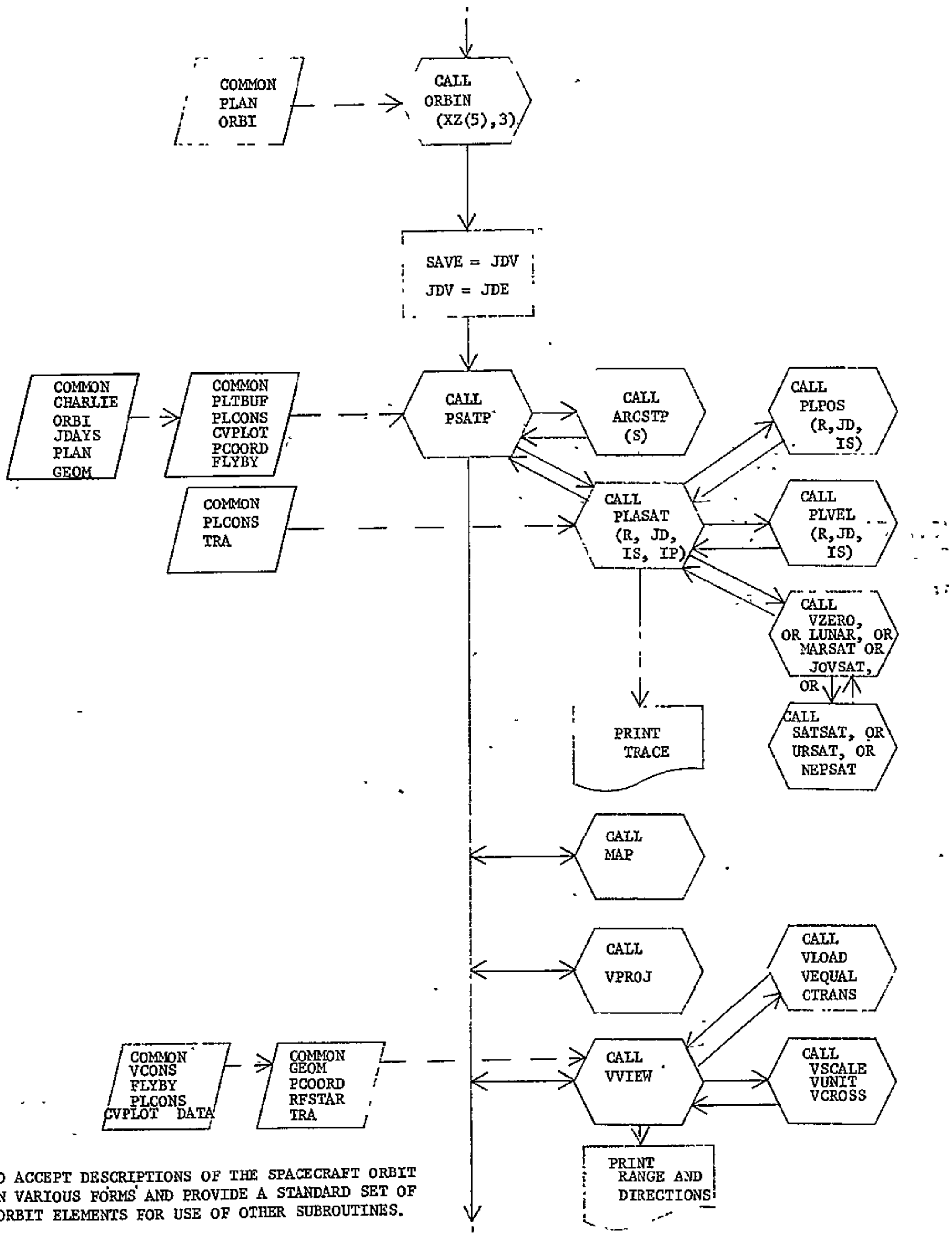
UVCHAR - TO ASSIGN A BLOCK OF INPUT VARIABLES TO VARIOUS ULTRA VIOLET IMAGERS

CALNDR - CONVERTS BETWEEN JULIAN, MODIFIED JULIAN, AND CALENDAR DATES.

FEB - DETERMINES LENGTH OF FEBRUARY

PRINT - PRINTS TRACE FOR DEBUGGING.

Fig. 3-2 (cont)



ORBIN - TO ACCEPT DESCRIPTIONS OF THE SPACECRAFT ORBIT IN VARIOUS FORMS AND PROVIDE A STANDARD SET OF ORBIT ELEMENTS FOR USE OF OTHER SUBROUTINES.

PSATP - PLOTS PLANET DISK, ORBITS OF PLANET SATELLITES, SPACECRAFT TRAJ., ETC.

ARCSTP - INPUTS INCOMING ASYMPTOTE UNIT VECTOR S.

PLASAT - CALCULATES POSITION VECTOR R(3) IN PLANETOCENTRIC, EQUATORIAL AND ECLIPTIC COORD.

PLPOS - CALCULATES HELIOCENTRIC POSITION OF PLANET.

PLVEL - CALCULATES HELIOCENTRIC VELOCITY OF PLANET.

VZERO - SETS VECTOR EQUAL TO ZERO.

LUNAR, MARSAT, JOVSAT, SATSAT, URSAT, NEPSAT - EPHEMERIS OF SATELLITES OF MOON AND/OR PLANETS CALLED IN BY INPUT IP.

PRINT - PRINTS TRACE WHEN DESIRED TO AID IN DEBUGGING.

MAP - SYSTEM PLOT SUBROUTINE.

VPROJ - SETS ORTHOGRAPHIC OR PERSPECTIVE VIEW.

VVIEW - DETERMINES RANGE AND DIRECTION OF VIEWS FROM SPACECRAFT TO TARGETS.

VLOAD - LOADS VECTORS

VSCALE - MULTIPLIES VECTOR BY SCALAR

VEQUAL - SETS A VECTOR EQUAL TO AN INPUT

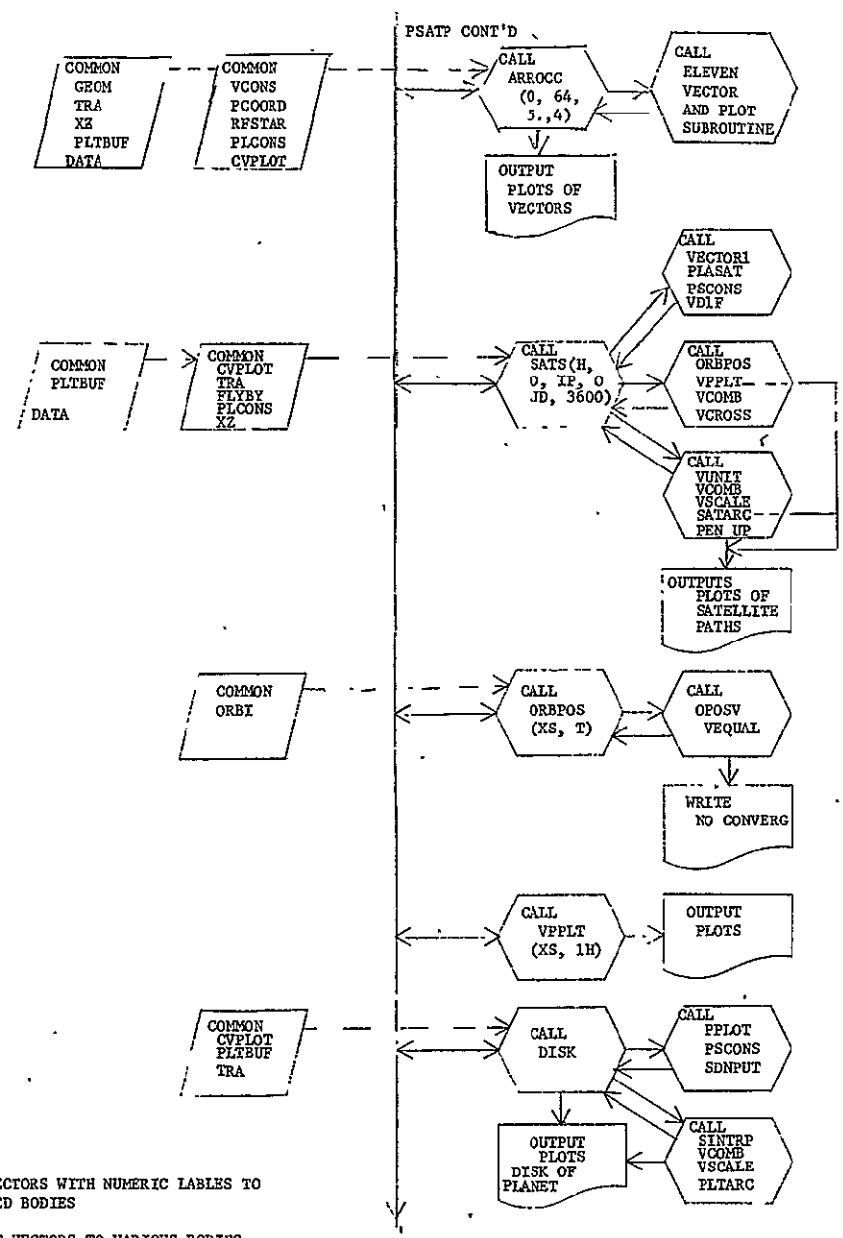
CTRANS - TRANSFORMS VECTOR TO CARTESIAN COORDINATES

VUNIT - SETS VECTOR = UNIT VECTOR

VCROSS - SETS V = A X B

PRINT - RANGE AND DIRECTIONS OF VIEW TARGET FROM SPACECRAFT.

Fig. 3-2 (cont)



ARROCC - DRAWS VECTORS WITH NUMERIC LABLES TO REQUESTED BODIES

OUTPUT - PLOTS OF VECTORS TO VARIOUS BODIES

SATS - CONTROLS PLOTTING OF SATELLITE PATHS.

VECTOR1 - PROVIDES TRACE FOR ALL SUBROUTINES CALLED.

PLASAT - CALCULATES POSITION VECTORS R(3) IN PLANETOCENTRIC, EQUATORIAL, AND ECLIPTIC COORD.

PSCONS - PROVIDES PLANET SATELLITE CONSTANTS.

VD1F - SETS V = A - B

ORBPOS - CALCULATES POSITION AND VELOCITY VECTORS FOR PLANETS AND SATELLITES WITH RESPECT TO S/C.

VPPLT - PLOTS LOCATION OF PLANET AND SATELLITES

VCOMB - SETS V = A • X + B • X

UCROSS - SETS A X B / |A X B|

VUNIT - SETS V = A / |A|

VSCALE - MULTIPLIES VECTOR BY SCALAR

SATARC - PLOTS LOCATION /POSITIONS OF PLANET AND SATELLITES.

PENUP - POSITIONS PLOT PEN

ORBPOS - CALCULATES ORBIT POSITION VECTOR

OPOSV - CALCULATES ORBIT POSITION VECTOR

VEQUAL - SETS V = A

WRITE - OUTPUTS "NO CONVERGENCE DLE" OR BASIS OF CONVERGENCE TEST.

VPPLT - PLOTS WITH INPUT SYMBOL

PLOT - OUTPUTS PLOTS WITH SYMBOLS

DISK - PLOTS PLANET DISK

PPLOT - CONTROLS PLOT OPERATIONS

PSCONS - PROVIDES PLANET SATELLITE CONSTANTS

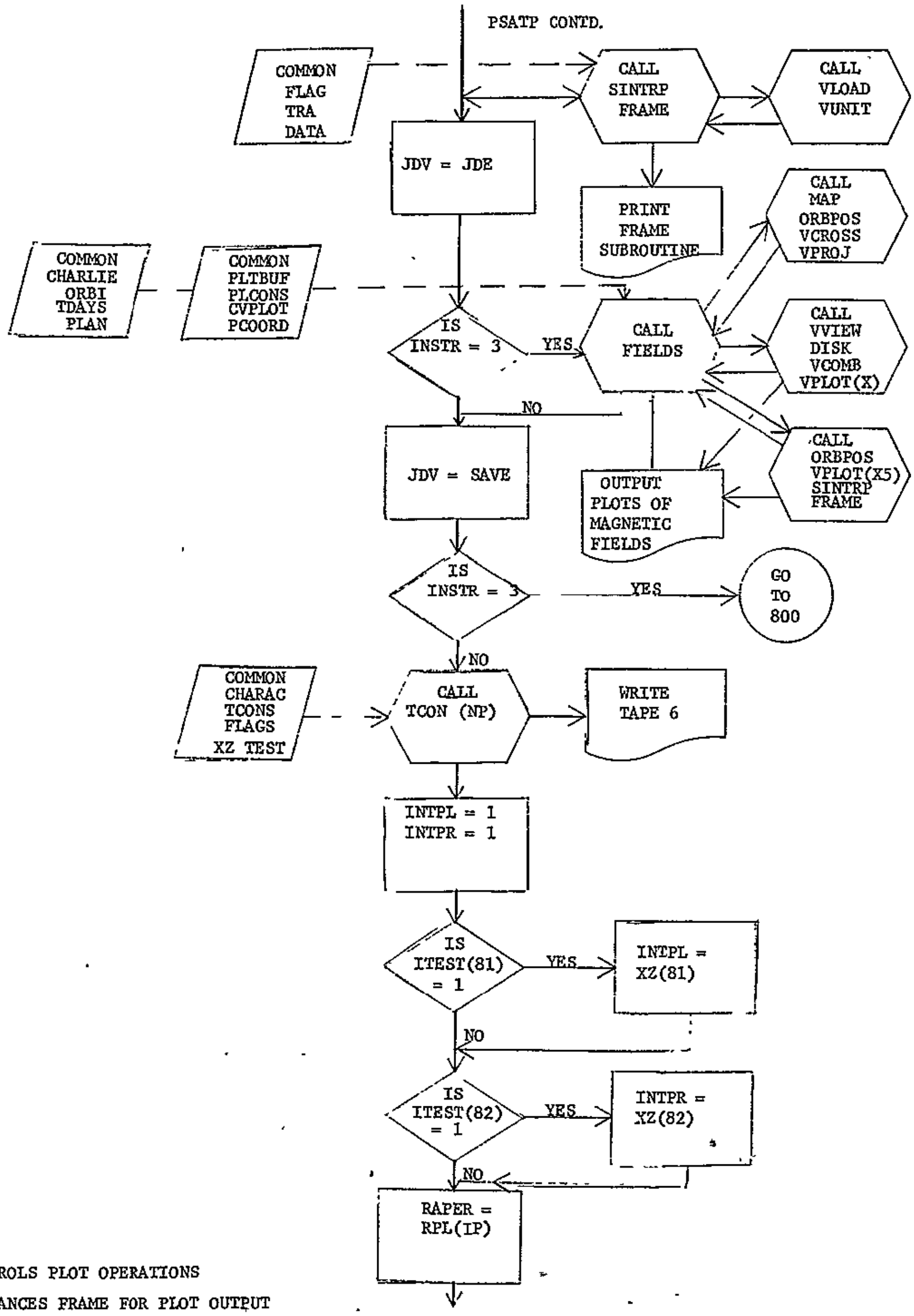
SNDPUT - CONTROLS PLOT OPERATIONS

SINTRP - CONTROLS PLOT OPERATIONS

PLTARC - PLOTS ARC OF MOVEMENT OF PLANET

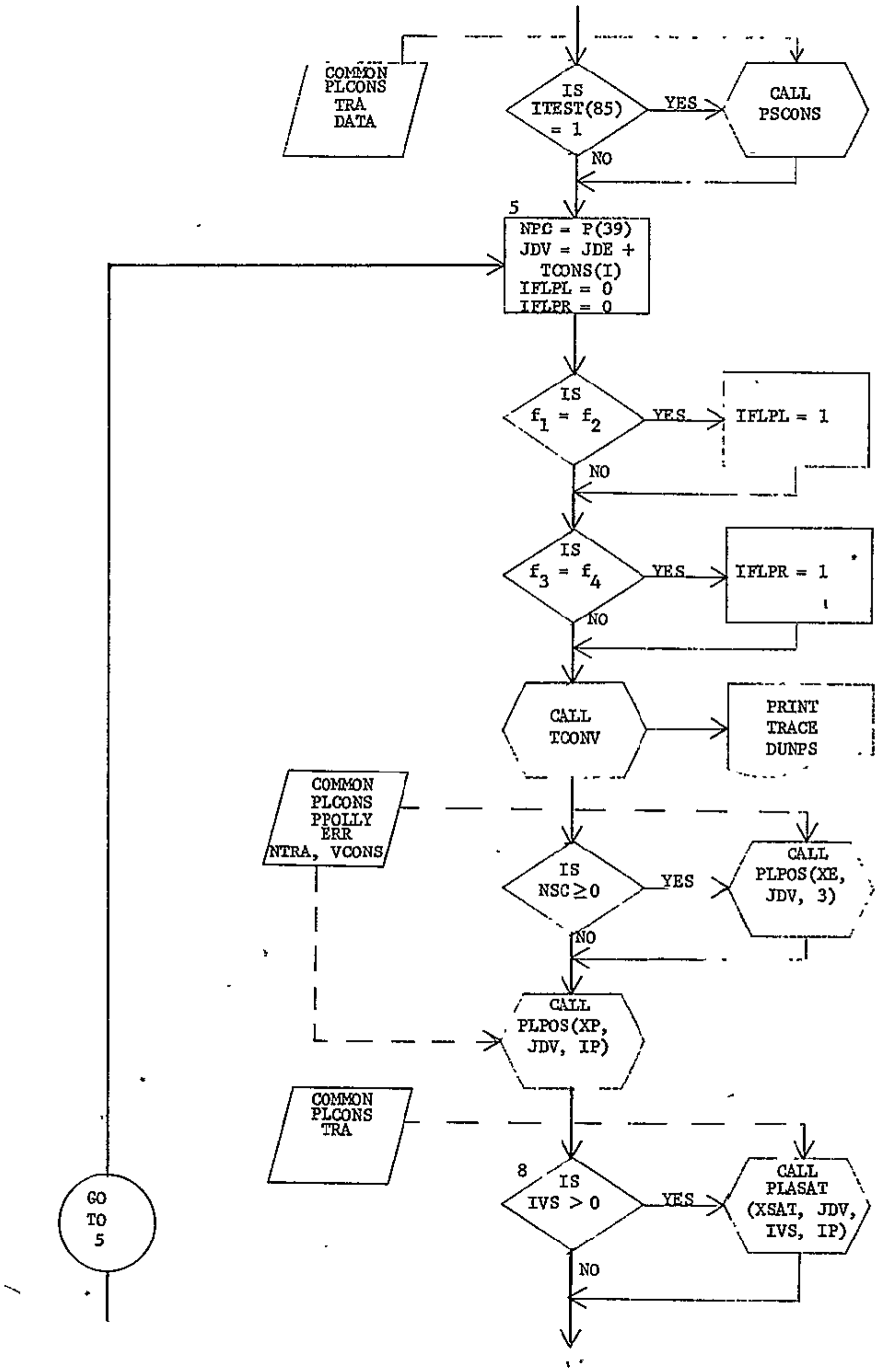
OUTPUT - PLOTS DISK OF PLANET AND MOVEMENT ARC OF SATELLITES.

Fig. 3-2 (cont)



SINTRP	-	CONTROLS PLOT OPERATIONS
FRAME	-	ADVANCES FRAME FOR PLOT OUTPUT
VLOAD	-	LOADS VECTORS
VUNIT	-	SETS V = A / |A|
PRINT	-	OUTPUTS TRACE FOR DEBUGGING
FIELDS	-	PLOTS THE MAGNETIC FIELD PATTERN
MAP	-	SYSTEM PLOT SUBROUTINE
ORBPOS	-	CALCULATES POSITION AND VELOCITY VECTORS FOR PLANETS AND SATELLITES WRT. SPACECRAFT
VCROSS	-	SETS V = A X B
VPROJ	-	SETS ORTHOGRAPHIC OR PERSPECTIVE VIEW
VVIEW	-	DETERMINES RANGE AND DIRECTION OF VIEWS FROM SPACECRAFT TO TARGETS
DISK	-	PLOTS PLANET DISK
VCOMB	-	SETS V = A · X + B · X
VPLOT(X)	-	CALLS VPLOT1 TO PLOT VECTORS AND LABELS FROM THE OBSERVER TO TARGETS.
SINTRP	-	CONTROLS PLOT OPERATIONS
FRAME	-	ADVANCES FRAME FOR PLOT OUTPUTS
OUTPUT	-	PLOTS OF VECTORS, PLANETS, LABELS AND PERSPECTIVE VIEWS OF MAGNETIC FIELDS
TCON	-	CONVERTS TIME BETWEEN DAYS, HOURS, MINUTES, SECONDS, OR MIXED
WRITE	-	WRITES TIME IN DAYS, HOURS, MINUTES, AND SECONDS ON TAPE 6.

Fig. 3-2 (cont)



PSCONS - PROVIDES PLANET AND SATELLITE CONSTANTS

f_1 = FLOAT (I-1)/INTPL

f_2 = (FLOAT (I) - 1.0)/FLOAT(INTPL)

f_3 = FLOAT (I-1)/INTRP

f_4 = (FLOAT (I) - 1.0)/FLOAT(INTPR)

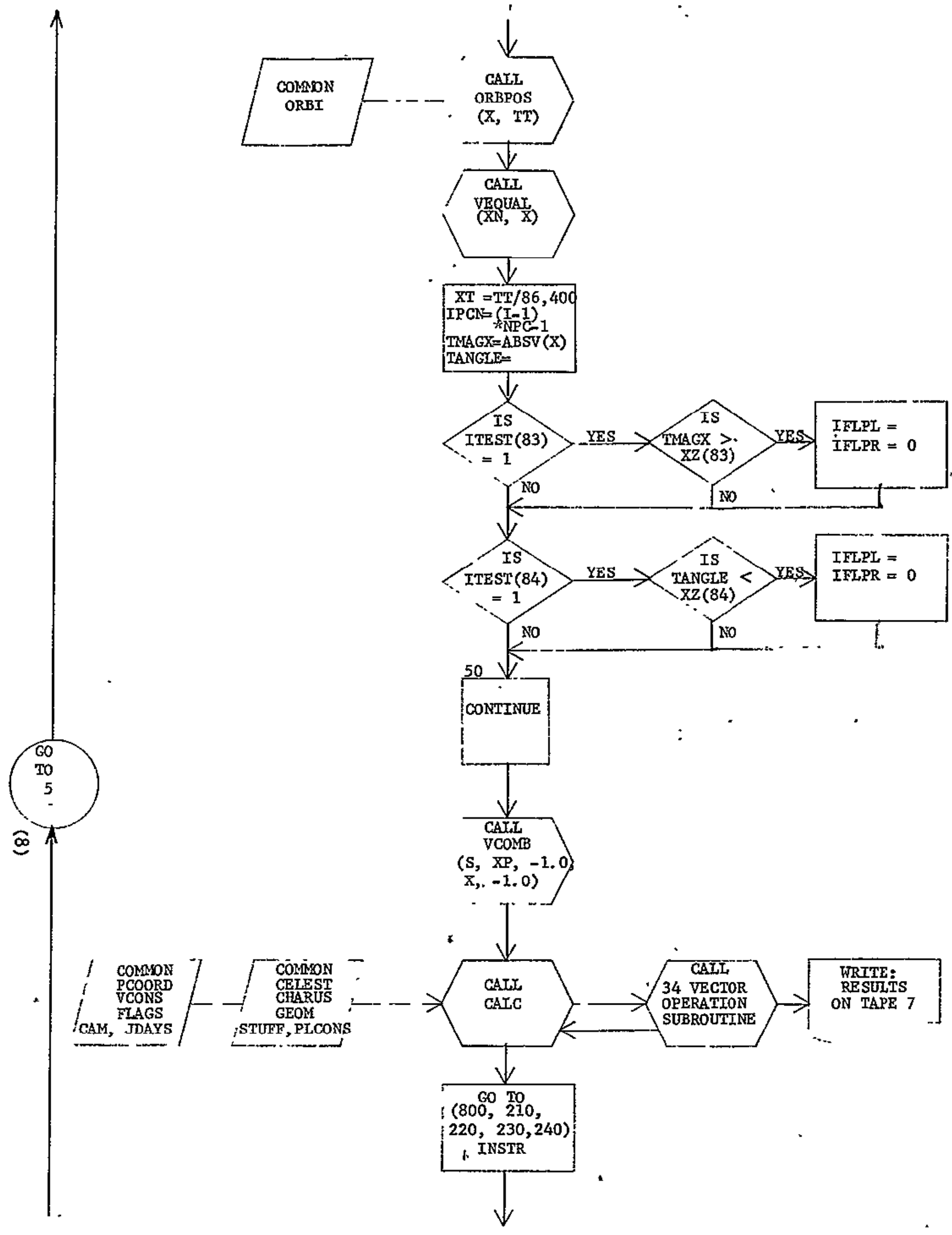
TCONV - CONVERTS TIME BETWEEN UNITS OF SECONDS, MINUTES, HOURS, AND DAYS

PRINT - DUMPS DEBUG DATA WHEN ERRORS ARE DETECTED

PLPOS - CALCULATES HELIOCENTRIC POSITION OF PLANETS

PLASAT - CALCULATES POSITION AND VELOCITY VECTORS IN PLANETOCENTRIC, EQUATORIAL, AND ECLIPTIC COORDINATES.

Fig. 3-2 (cont)



ORBPOS - CALCULATES ORBIT POSITION VECTOR

VEQUAL - SETS V = A

WRITE - OUTPUTS "NO CONVERGENCE DLE" ON BASIS OF CONVERGENCE TEST

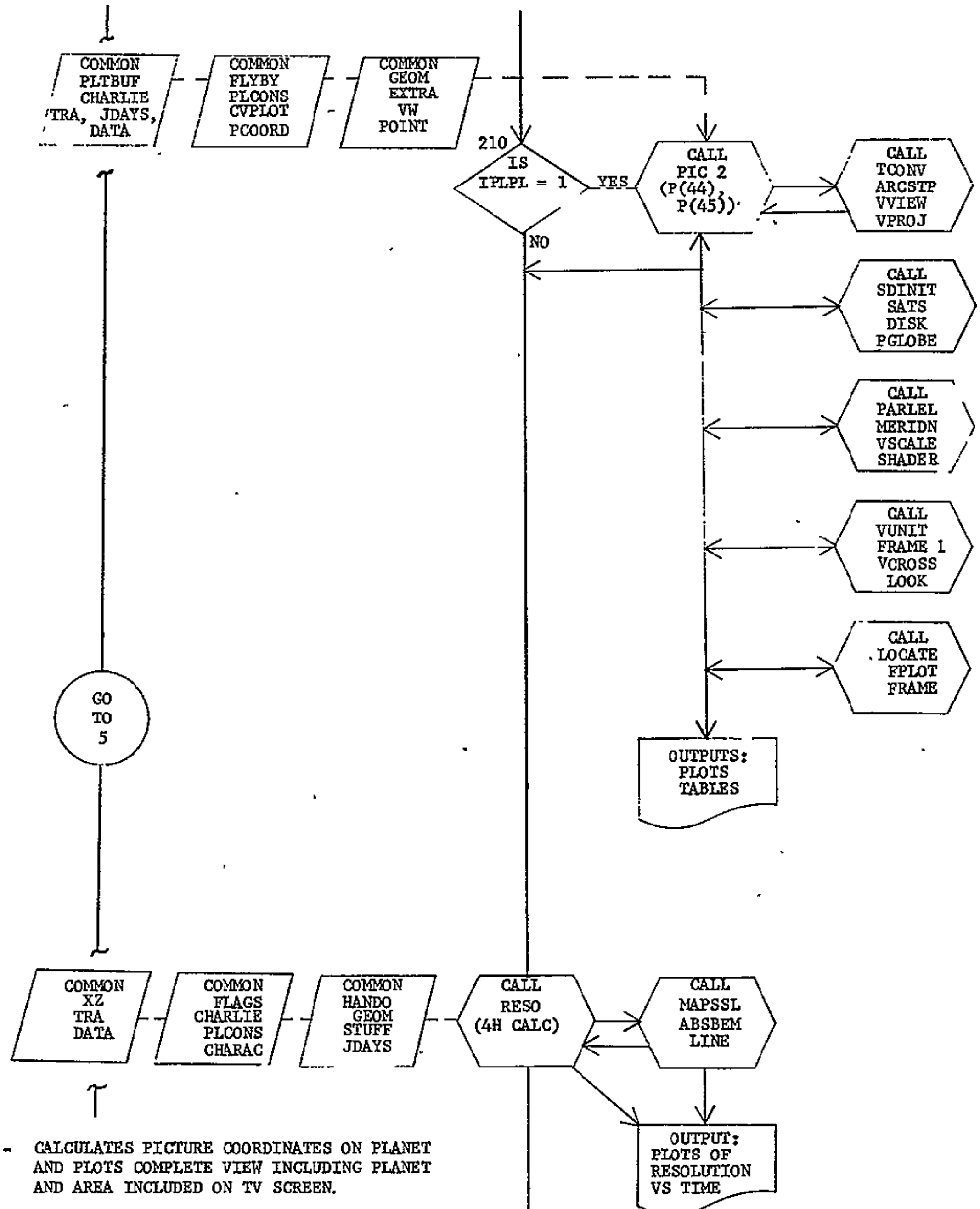
VEQUAL - SETS XN = X

VCOMB - SETS S = XP • -1.0 + X • -1.0

CALC - CALCULATES RANGE, CONE, AND CLOCK ANGLES TO PLANET AND SATELLITES; VECTORS TO PICTURE TARGETS; VELOCITIES RELATIVE TO PICTURE; LOCATION OF PICTURE ON TARGET; PHASE ANGLES; AND TO AND THRU THE LENS

WRITE - WRITES RESULTS OF CALCULATIONS ON TAPE 7.

Fig. 3-2 (cont)



PIC2 - CALCULATES PICTURE COORDINATES ON PLANET AND PLOTS COMPLETE VIEW INCLUDING PLANET AND AREA INCLUDED ON TV SCREEN.

TCONV - CONVERTS TIME BETWEEN DAYS, HOURS, MINUTES, SEC.

ARCSTP - SETS ANGLE FOR PLOTTING ARC.

VVIEW - DETERMINES RANGE AND DIRECTION OF VIEWS.

VPROJ - SETS ORTHOGRAPHIC OR PERSPECTIVE VIEW.

SDINIT - SETS UP SYSTEM PLOTTER AND CALCULATES SCALE UNITS AND COORDINATES.

SATS - CONTROLS PLOTTING OF SATELLITE PATHS.

DISK - PLOTS PLANET DISK.

PGLOBE - CALCULATES PLANETOCENTRIC TO PLANET POINTS

PARLEL - PLOTS PARALLELS ON PLANET

MERIDN - PLOTS MERIDIANS ON PLANET

VSCALE - MULTIPLIES A VECTOR BY A SCALAR

SHADER - SHADES AREA PAST TERMINATOR

VUNIT - SETS VECTOR TO UNITY.

FRAME 1 - ADVANCES FRAME FOR PLOT OUTPUTS

VCROSS - SETS V = A X B

LOOK - CALCULATES PLANET LAT. AND LONG. FROM CONE - AND CLOCK ANGLES AND PRINTS RESULTS

LOCATE - CALCULATES PLANET VIEW AREA

FPLOT - PLOTS

FRAME - ADVANCES FRAME FOR PLOT OUTPUTS

OUTPUTS - PLOTS OF PLANET AND VIEW AREAS AND DIGITAL TABLES OF CALCULATIONS.

RESO - CALCULATES RESOLUTION OF TV IMAGER AND PLOTS RESULTS VERSUS TIME.

MAPSSL - SYSTEM PLOT SUBROUTINE.

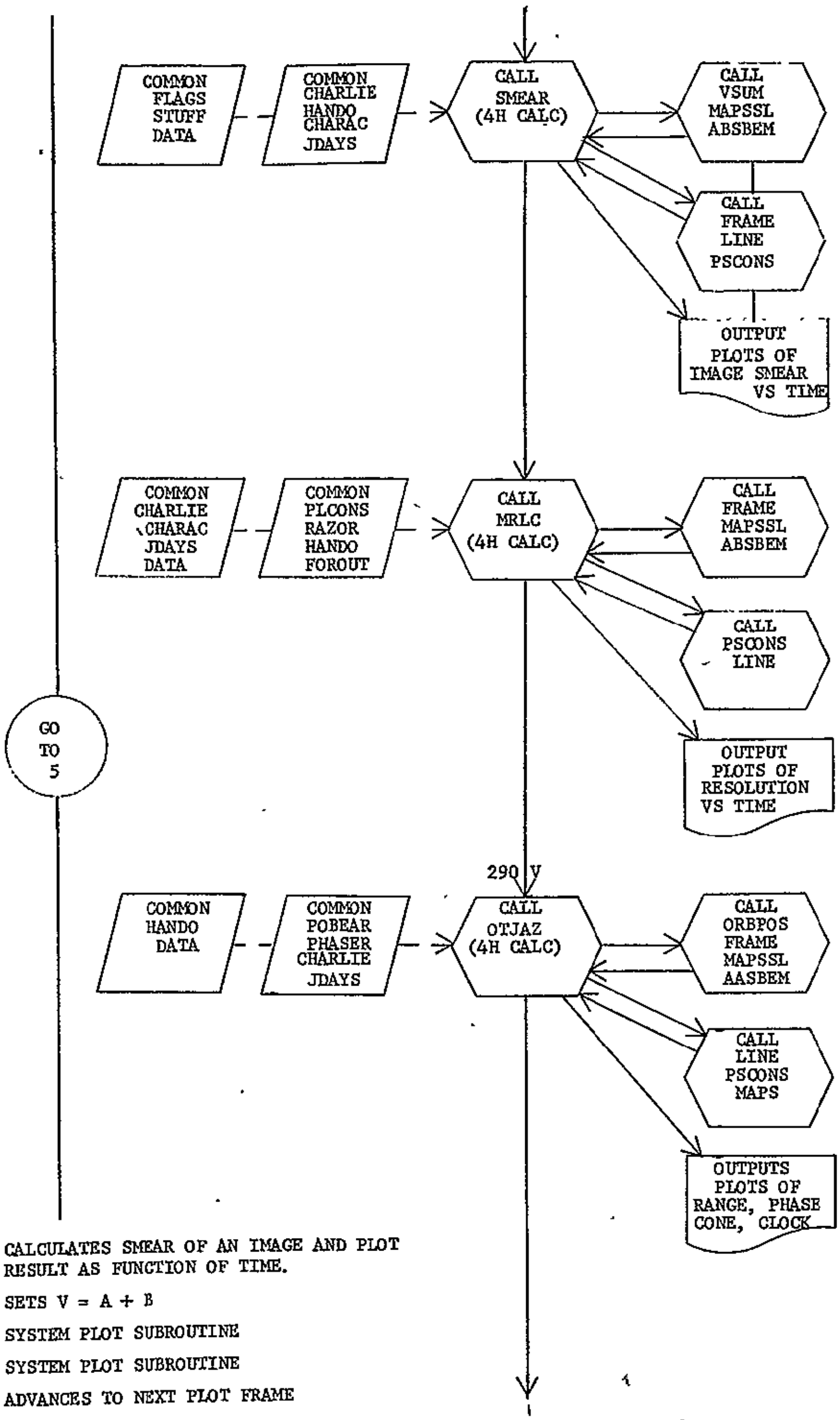
ABSBEM - SYSTEM PLOT SUBROUTINE.

LINE - SYSTEM PLOT SUBROUTINE

FRAME - ADVANCES TO NEXT PLOT FRAME.

OUTPUT - PLOTS OF RESOLUTION VERSUS TIME AND DIGITAL TABLE OF RESOLUTION VALUES.

Fig. 3-2 (cont)



SMEAR - CALCULATES SMEAR OF AN IMAGE AND PLOT RESULT AS FUNCTION OF TIME.

VSOM - SETS V = A + B

MAPSSL - SYSTEM PLOT SUBROUTINE

ABSBEM - SYSTEM PLOT SUBROUTINE

FRAME - ADVANCES TO NEXT PLOT FRAME

LINE - SYSTEM PLOT SUBROUTINE

POINT - SYSTEM PLOT SUBROUTINE

PSCONS - PROVIDES SATELLITE AND PLANET CONSTANTS

OUTPUT - PLOTS OF IMAGE SMEAR AS FUNCTION OF TIME

MRLC - MINIMUM RESOLUTION LEVEL CALCULATION OF NUMBER OF PICTURES VERSUS TIME WITHIN GIVEN RESOLUTION RANGE.

NUMBER - SYSTEM PLOT SUBROUTINE

OUTPUT - PLOTS OF RESOLUTION VERSUS TIME FOR TARGETS IN FIELD OF VIEW

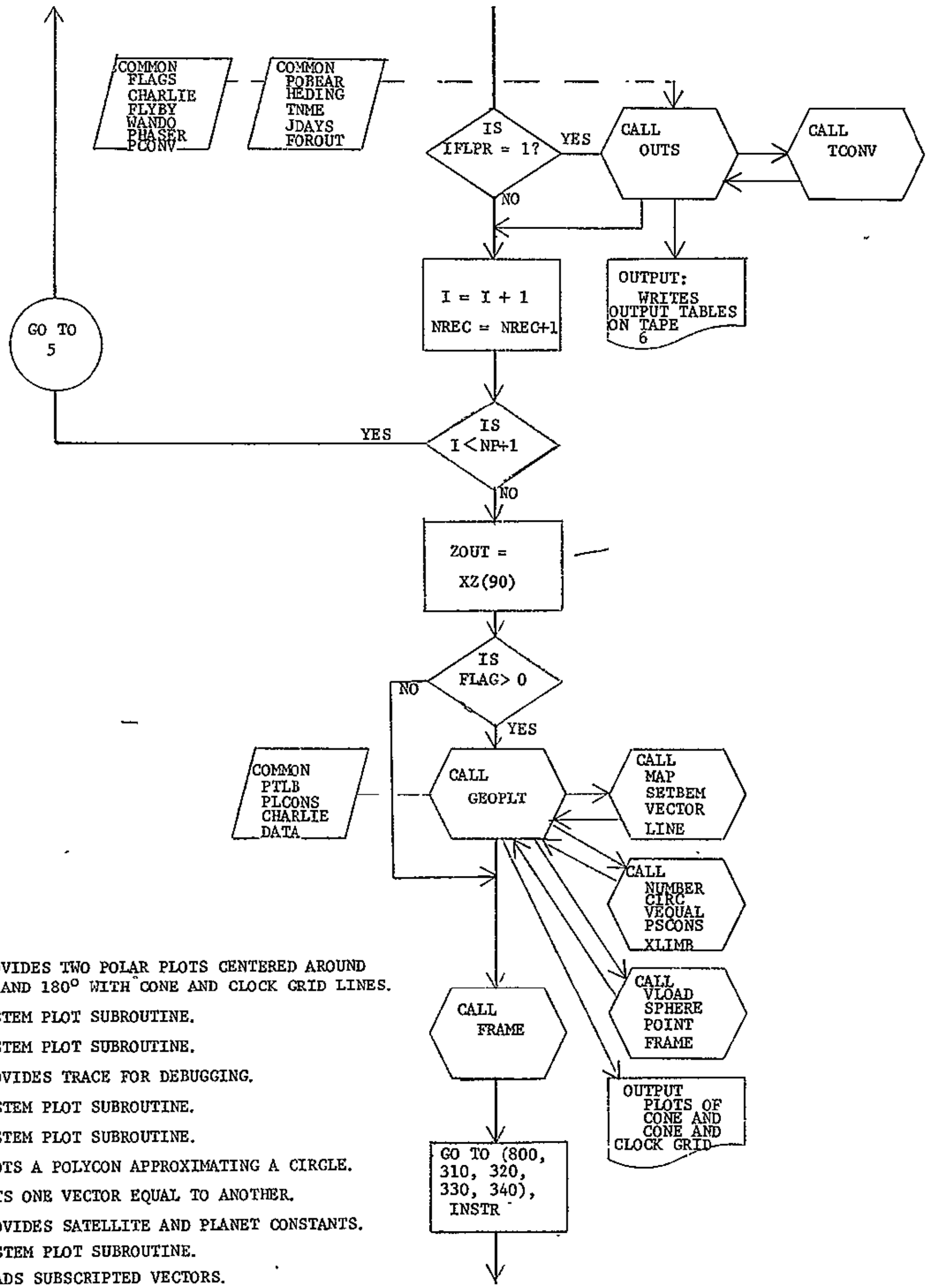
OTJAZ - PLOTS: (THIS ITERATION CALCULATES)
1. SATELLITE RANGE VERSUS TIME
2. SATELLITE PHASE ANGLE VERSUS TIME
3. SATELLITE CONE ANGLE VERSUS TIME
4. SATELLITE CLOCK ANGLE VERSUS TIME

ORBPOS - CALCULATES ORBIT POSITION AND VELOCITY VECTORS FOR PLANET AND SATELLITE WITH RESPECT TO SPACECRAFT.

MAPS - SYSTEM PLOT SUBROUTINE

OUTPUTS - SEE OTJAZ

Fig. 3-2 (cont)



GEOPLT - PROVIDES TWO POLAR PLOTS CENTERED AROUND 0° AND 180° WITH CONE AND CLOCK GRID LINES.

MAP - SYSTEM PLOT SUBROUTINE.

SETBEM - SYSTEM PLOT SUBROUTINE.

VECTOR - PROVIDES TRACE FOR DEBUGGING.

LINE - SYSTEM PLOT SUBROUTINE.

NUMBER - SYSTEM PLOT SUBROUTINE.

CIRC - PLOTS A POLYCON APPROXIMATING A CIRCLE.

VEQUAL - SETS ONE VECTOR EQUAL TO ANOTHER.

PSCONS - PROVIDES SATELLITE AND PLANET CONSTANTS.

XLIMB - SYSTEM PLOT SUBROUTINE.

VLOAD - LOADS SUBSCRIPTED VECTORS.

SPHERE - TRANSFORMS A VECTOR BETWEEN CARTESIAN AND SPHERICAL COORDINATES.

POINT - SYSTEM PLOT SUBROUTINE.

FRAME - ADVANCES FRAME FOR NEXT PLOT.

OUTPUTS - POLAR PLOTS OF CONE AND CLOCK GRIDS.

310 - TV ROUTINES

320 - FP ROUTINES

330 - IR ROUTINES

340 - UV ROUTINES

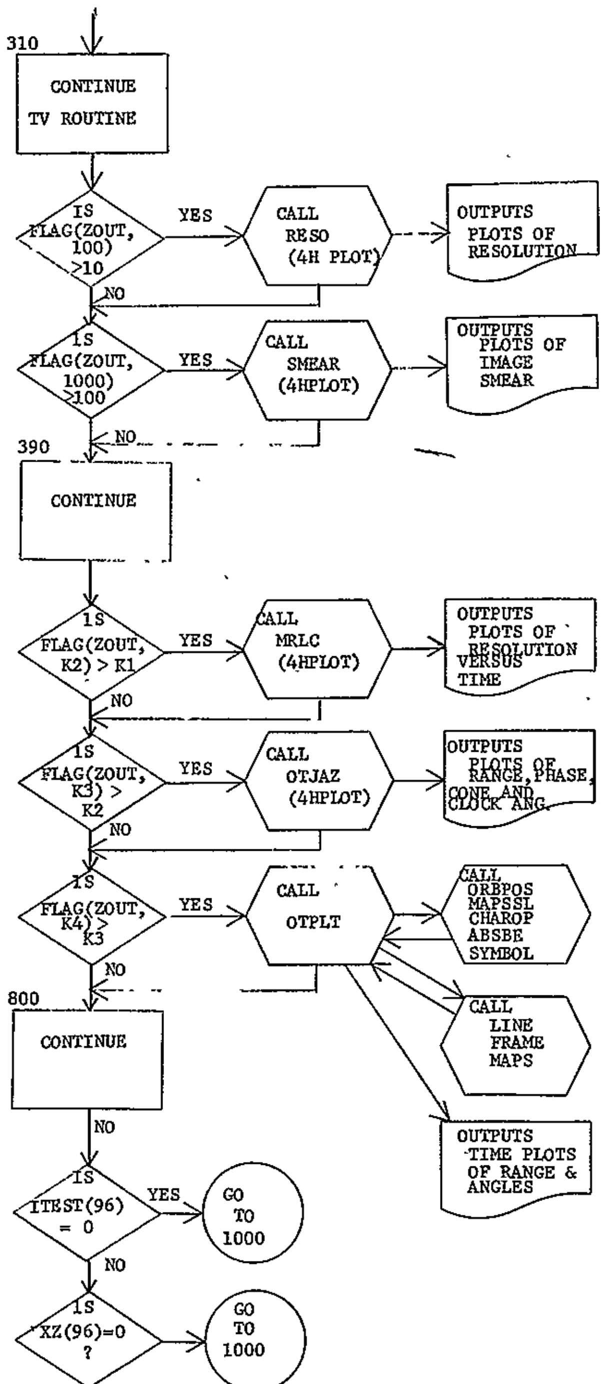
800 - NO INSTRUMENTATION

OUTS - WRITES OUTPUT TAPE 6 IN FORMAT FOR FAROUT TABLES

TCONV - CONVERTS TIME BETWEEN DAYS, HOURS, MINUTES AND SECONDS.

OUTPUT - WRITES OUTPUT TABLES ON OUTPUT TAPE 6.

Fig. 3-2 (cont)



RESO - CALCULATES RESULTS OF TV IMAGER AND OUTPUTS PLOTS OF RESOLUTION VERSUS TIME. (THIS ITERATION OUTPUTS PLOTS).

SMEAR - CALCULATES SMEAR OF AN IMAGE AND PLOTS RESULT AS A FUNCTION OF TIME (THIS ITERATION OUTPUTS PLOTS)

K1 = 1000
K2 = 10000
K3 = 100000
K4 = 1000000 } INPUT FLAG VALUES

MRLC - MINIMUM RESOLUTION LEVEL CALCULATION OF NUMBER OF PICTURES VERSUS TIME WITHIN A GIVEN RESOLUTION LEVEL. (THIS ITERATION OUTPUTS PLOTS).

OTJAZ - PLOTS: (THIS ITERATION PLOTS)
1. SATELLITE RANGE VERSUS TIME;
2. SATELLITE PHASE ANGLE VERSUS TIME;
3. SATELLITE CONE ANGLE VERSUS TIME;
4. SATELLITE CLOCK ANGLE VERSUS TIME.

OTPLT - TIME PLOTS OF:
1. PLANET RANGE;
2. PLANET PHASE ANGLE;
3. PLANET CONE ANGLE;
4. PLANET CLOCK ANGLE;
5. SATELLITE RANGE;
6. SATELLITE PHASE ANGLE;
7. SATELLITE CONE ANGLE;
8. SATELLITE CLOCK ANGLE.

ORBPOS - CALCULATES ORBIT POSITION AND VELOCITY VECTORS FOR PLANET AND SATELLITE WITH RESPECT TO SPACECRAFT.

MAPSSL; CHAROP; ABSBEM; SYMBOL; LINE; FRAME; AND MAPS - SYSTEM PLOT SUBROUTINES.

OUTPUTS - SEE OTPLT ABOVE.

Fig. 3-2 (cont)

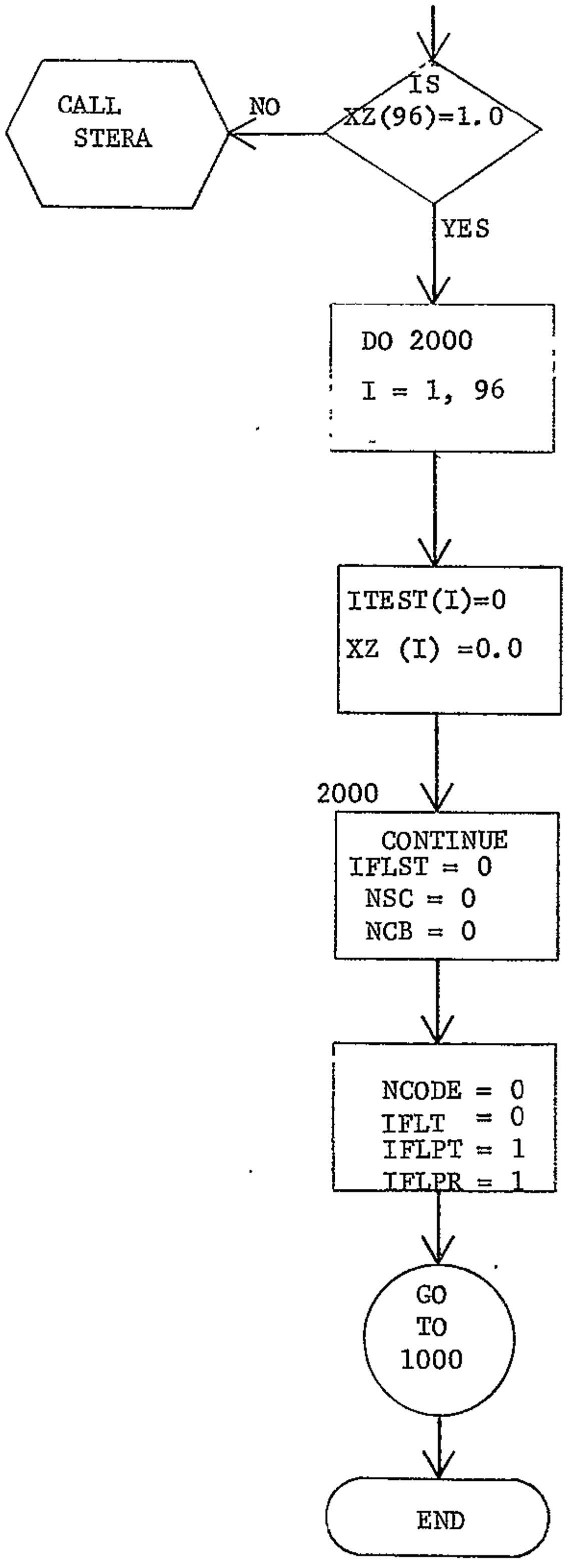

CALL
STERA
NO
IS
XZ(96)=1.0
YES
DO 2000
I = 1, 96
ITEST(I)=0
XZ (I) =0.0
2000
CONTINUE
IFLST = 0
NSC = 0
NCB = 0
NCODE = 0
IFLT = 0
IFLPT = 1
IFLPR = 1
GO
TO
1000
END

Fig. 3-2 (concl)

3.3 <u>Computer Translation Information</u> - Table 3-3 is a comparison of the Univac 1108 and CDC 6400/6500 computers. From this data, the following programming guidelines were adopted in developing the FORTRAN IV CDC 6400/6500 program that was subsequently converted to FORTRAN V for the Univac 1108:

1. The 60 bit word of the CDC 6400/6500 allows 10 Hollerith characters. Univac's 36 bit word permits only six characters per word.

   Use six or less Hollerith characters per word.

2. The CDC 6400/6500 FORTRAN IV compiler allows symbols with up to seven alphanumeric characters. Univac 1108 FORTRAN V permits only six alphanumeric characters.

   Use a maximum of six alphanumeric characters per symbol.

3. CDC FORTRAN IV allows Hollerith characters in FORMAT statements to be specified by placing asterisks around the Hollerith information. Univac has this feature with apostrophes instead of asterisks.

   Use the Hollerith FORMAT specification nH.

4. CDC FORTRAN IV allows the DO loop index to be altered by a replacement statement within the loop. Univac FORTRAN V does not permit DO indices to be altered within the loop.

   Do not reset DO loop indices within the iteration loop.

5. CDC FORTRAN IV allows the DO loop index to be referenced outside the limits of the loop. Univac does not have this feature.

   Do not reference the index of a DO loop outside the iteration limits. Store the index value in a separate word.

6. The CDC loader allocates storage for common blocks according to the length of their first appearance in the job string. The Univac loader searches the referenced elements and allocates storage for the longest appearance of the common block.

   Keep common statements consistent throughout all elements.

Table 3-3 Univac 1108/CDC 6500 Comparisons

ITEM	UNIVAC 1108	CONTROL DATA 6400/6500
MEMORY	$65K_{10}$	$150K_8 \sim 54K_{10}$
WORD LENGTH	36 Bits	60 Bits
FLOATING PT. BIAS	200_8	2000_8
DOUBLE PREC. BIAS	2000_8	2000_8
MANTISSA	FRACTION	INTEGER
REAL	$10^{-38} < R < 10^{38}$ (9 decimal digits)	$10^{-294} < R < 10^{322}$ (14 decimal digits)
DOUBLE PRECISION	$10^{-308} < D < 10^{308}$ (18 decimal digits)	$10^{-294} < D < 10^{322}$ (31 decimal digits)
INTEGER	$\|I\|<(2^{35} - 1)$	$\|I\| \leq (2^{59} - 1)$ DO index $I \leq (2^{17} - 2)$ $I \rightarrow R$ $\|R\|$ $(2^{48} - 1)$
CHARACTER	6 FIELDATA CHAR/WORD	10 Hollerith Char/Word
ARRAYS	UP TO 7 DIMENSIONS	UP to 3 DIMENSIONS
WALK BACK	AVAILABLE	NA
FUNCTIONS: (1108 has many functions that are not available on (DI 6400/6500	CBRT(X)	NA
	$X\emptyset R(X_1, X_2, \ldots, Xn$	NA
	BOOL	NA
	FLD	NA

ITEM EXECUTABLE	UNIVAC 1108	CONTROL DATA 6400/6500
DO n i = M_1, M_2, M_3	$\left(M_j\right)$ may be signed constant or variable	$\left(M_j\right)$ unsigned (variable may have negative value)
	$\left(M_j\right)$, i may not be altered during execution of DO	$\left(M_j\right)$, i may be changed during execution of DO
RETURN	RETURN k	RETURN
STOP	STOP name	STOP 00000_8
PAUSE	PAUSE name	PAUSE 00000_8
READ	READ (u,f, ERR = c, END = d) list	READ (u,f) list
WRITE	WRITE (u,f, ERR = c, END = c)	WRITE (u,f) list
ENCODE	ENCODE (c,v,f) list v = block f = format c = no. of char.	ENCODE (c,f,v) list v = block f = format c = no. char. in record
DECODE	DECODE (c,v,f) list	DECODE (c,f,v) list

Table 3-3 (concl)

ITEM NONEXECUTABLE	UNIVAC 1108	CONTROL DATA 6400/6500
BLOCK DATA	Reference in MAP	BLOCK DATA name
FORMAT	'hhh'	*hhh*
PARAMETER	PARAMETER	NA
INTEGER	INTEGER list /data/	INTEGER
REAL	REAL list /data/	REAL
DOUBLE PRECISION	DOUBLE PRECISION list /data/	DOUBLE PRECISION or DOUBLE
COMPLEX	COMPLEX list /data/	COMPLEX
LOGICAL	LOGICAL list /data/	LOGICAL
ABNORMAL	ABNORMAL	NA
DIMENSION	DIMENSION list /data/	DIMENSION
EQUIVALENCE	EQUIVALENCE (V(o), w)	EQUIVALENCE (X(n), w)
COMMON	Largest block loaded	First block loaded
DATA	Hollerith may be in 'hhh' nested implied DO's	Hollerith in nH field. Single variable subscript
END	n END n is statement number	END name name is element name
NAME LIST		
ENTRY	ENTRY name (a_1, a_2, a_3) unique arguments not necessarily like parent	ENTRY name arguments defined in parent element
IMPLICIT	IMPLICIT	NA
DEFINE	DEFINE name (1) =	Name (1) =
COMPILER	COMPILER	NA
ITEM SOURCE CONTROL	**UNIVAC 1108**	**CONTROL DATA 6400/6500**
INCLUDE	INCLUDE	NA
DELETE	DELETE	NA
EDIT	EDIT	NA

7. The CDC loader creates an absolute element of all elements in the job string including BLOCK DATA subprograms. The Univac loader only loads those elements that are referenced by the main program and subsequently referenced elements.

   Do not use BLOCK DATA subprograms. All elements to be loaded must be referenced - whether or not the CALL statement will be executed.

8. The following list indicates special CDC 6400/6500 FORTRAN IV capabilities that are incompatible with Univac 1108 FORTRAN V. An alternative statement is shown where applicable.

Special CDC 6400/6500 FORTRAN IV Statements that are Incompatible	CDC 6400/6500 FORTRAN IV Statements that are Compatible
DOUBLE list	DOUBLE PRECISION list
PAUSE 00000_8	PAUSE
STOP 00000_8	STOP
END name	END
IF (logical expression) N1,N2	IF (logical expression) GO TO
IF (UNIT, i) N1,N2,N3,N4	Not applicable on Univac 1108
IF (EOF, i) N1,N2	Not applicable on Univac 1108
IF (ENDFILE i) N1,N2	Not applicable on Univac 1108
BUFFER IN (i,m) list	Not applicable on Univac 1108
BUFFER OUT (i,m) list	Not applicable on Univac 1108
Ow FORMAT specification	Not applicable on Univac 1108
Multiple replacements	Not applicable on Univac 1108
Multiple statements per line	Not applicable on Univac 1108

4. MODEL SYMBOL DICTIONARY

This section presents a definition of the program symbols, or variables, which are common to two or more subroutines.

4.1 Common Blocks - Symbols are used to identify program variables. Those symbols used in two or more subroutines are assigned to blocks of variables and identified by a code name. The blocks are referred to as "common blocks" by the code name. Table 4-1 presents the common blocks of variables, the code letters for the variable (symbol), variable definition, and the loading subroutine.

4.2 Model Inputs - The input consists of a set of cards that define the mission, date, type of spacecraft orientation, instrumentation, and instrument characteristics. In addition, there are a number of stored values such as planet constants and planet positions with respect to time. MAIN calls in subroutine INPUT, which reads in the input data. These input values are assigned a common block, identified and dimensioned as variables XZ (97). The names of the variables are assigned by the INPUT subroutine, which also reads additional input cards for labeling and identification of the computer output run.

Table 4-2 contains the input variable identification for input data. The particular inputs for television, and for fields and particles instrumentation are identified in Tables 4-3 and 4-4, respectively.

4.3 Tape Files - Because of the large number of subroutines, tapes are used to store calculated values for many variables that are calculated in one subroutine and used by other subroutines. Table 4-5 defines the variables that are calculated by the loading subroutine and read onto a tape by tape number. Subroutines that use these variables are also identified.

Table 4-1 Definition of Variables

COMMON BLOCK	VARIABLE SYMBOL	DEFINITION OF VARIABLE	LOADING SUBROUTINE
/CAM/	CCONE	User's input cone angle for camera	INPUT
	CCLOK	User's input clock angle for camera	INPUT
/CELEST/	C(3)	Unit vector to Canopus	MAIN
	C(3)	Vector to sun from S/C	MAIN
	SN(3)	Not used	
/CHARAC/	P(60)	Instrument input characteristics	TVCHAR, ETC.
/CHARLI/	NCODE	Not used	
	NREC	Number of records	MAIN
	IP	Planet number index	INPUT
	IVS	Satellite number index	INPUT
	NSC	Spacecraft type index	MAIN
	IFLST	Type run flag = 1 satellite = 0 planet/ can switch	MAIN
	NCB	Switch index = 0, on planet; = 1 switched to satellite	CALC
/ERR/	ERFLAG	Error flag.	
/EXTRA/	RPLP/ROT	Radius of planet/planet rot. rate (rev/day)	PIC2
/FAME/	C1(3)	Camera coord axis left to right vector	LOCATE
	C3(3)	Camera coord axis S/C to object vector	LOCATE
/FLAG/	LL	Vector index if = 1 only center; ≠ all nine vectors.	
	NI	Symbol plotter at center of TV frame plot	FPLOT
	MFL	= 0	FPLOT
/FLAGS/	IFLT	Instrument mode (I mode) flag	INPUT
	IFLPL	Plot flag = 1, plot data	MAIN
	IFLPR	Print flag = 1, print data	MAIN
/FLYBY/	SI(3)	Vector of incoming asymptote	
	TP(3)	Vector of trajectory pole	PSATP
	XN(3)	Vector of S/C from planet (modified from sat)	ORBPOS
	PA(3)	Vector to periapsis from planet center	
	SO(3)	Vector of outgoing asymptote	
	SV(3)	Vector to satellite from planet	PIC2
/FOROUT/	POWER	Power thru lens (ergs/cm^2/sec)	CALC
	ENERGY	Total energy for exposure (ergs)	CALC
	PERCT	% of dynamic range above threshold	CALC
	IDN	Digital number (shades of gray)	CALC
	R(3)	Range, cone & clock to planet from S/C	CALC
	RSAT(3)	Range, cone & clock to satellite from S/C	CALC
	TARG P(3)	Radius, latitude & longitude of picture center	CALC
	IPCN	Picture number	MAIN
	FRACT	Fraction of dynamic range for exposure	CALC
	PHASED	Phase angle of planet (deg)	CALC
	PHASEP	Phase angle of specified satellite (deg)	CALC

Table 4-1 (cont)

COMMON BLOCK	VARIABLE SYMBOL	DEFINITION OF VARIABLE	LOADING SUBROUTINE
	POWLN	Power flux to lens (ergs/cm^2/sec)	CALC
	NDN	Calculated exposure in digital no. above min.	CALC
	EPERDN	Energy difference per digital number	CALC
/GEOM/	XE(3)	Vector to Earth from sun	PLPOS
	XP(3)	Vector to planet from sun/modified to planet from Earth	PLPOS
/HANDO/	TIM	Max time from periapsis for plot limit (days)	RESO
	REC(9)	Resolution (km/pix) for planet/8 sat at center	RESO
	REL(9)	Resolution (km/pix) for planet/8 sat at limb	RESO
	RES	Resolution (km/pix) for principal body center	RESO
	SMR(9)	Velocity of pic ctr normal to site due to orbit	SMEAR
	SMRR(9)	Smear in pixels	SMEAR
	SMRS(9)	Velocity of pic ctr normal to site due to S/C	SMEAR
	TSMR(9)	Exp. time to reduce smear to less than one pixel	SMEAR
	SPSMR	Prin body velocity due to orbit normal to sight	SMEAR
	SPSMRS	Prin body velocity due to S/C	SMEAR
	SPTSMR	Exposure time to reduce smear	SMEAR
	SPSMRR	Smear in pixels	SMEAR
/HEDING/	TITLE	Mission title (78 characters)	INPUT
/JDAYS/	JDV	Julian date	MAIN
	T	Time from periapsis (sec)	MAIN
/JUNK/	PB	Planet surface brightness	FOLDER
	EFBP	Effective planet brightness	FOLDER
	B(8)	Satellite brightness	FOLDER
	EFB(8)	Effective satellite brightness	FOLDER
/NTRA/	NTRACE	"No Trace"	
/ORBITQ/	DUMZ(7)	Dummy	ORBIT
	E	Eccentricty	ORBIT
	H	Angular momentum	ORBIT
	PQ	Semilatus rectum	ORBIT
	AA	Semimajor axis	ORBIT
	DUMZZ(A)	Dummy	ORBIT

Table 4-1 (cont)

COMMON BLOCK	VARIABLE SYMBOL	DEFINITION OF VARIABLE	LOADING SUBROUTINE
/PC/	PC(3,9)	Frame coordinate set in ecliptic coord.	LOCATE
/PCOORD/	PV(3)	Planet pole vector	CPLANN
	VE(3)	Planet vernal equinox vector	CPLANN
	PM(3)	Planet prime meridian vector	CALC
/PHASER/	PHAS(8)	Phase angle for 8 possible satellites	CALC
	SRAT(3,8)	Range, cone, clock to satellite from S/C	CALC
/PHOTO/	SOLAR(31)	Solar intensity (erg/cm^2/micron)	FOLDER
	ACBDO (31,3)	Albedo per wavelength band per model	FOLDER
	F(31,3)	Filter transmission per wavelength band	FOLDER
	CAMEF (31,3)	Camera sensitivity per wavelength band	FOLDER
/PLAN/	GM	Planet gravitational constant (km^3/sec^2)	CPLANN
/PLCONS/	GMS	Sun gravitational constant	PLCON
	GMPL(12)	Planet gravitational constant	PLCON
	RS	Radius of sun (km)	PLCON
	RPL(12)	Radius of planet	PLCON
	SNAME	Name of sun	PLCON
	PLNAME(12)	Name of planet	PLCON
	SPV(3)	Sun pole vector	PLCON
	PLPV(3,12)	Planet pole vector	PLCON
	SVE(3)	Sun vernal equinox vector	PLCON
	PLVE(3,12)	Planet vernal equinox vector	PLCON
	SROT	Sun's rotation rate (rev/day)	PLCON
	PLROT(12)	Planet's rotation rate (rev/day)	PLCON
	SOBL	Sun's oblateness	PLCON
	PLOBL(12)	Planet's oblateness	PLCON
/PLOTER/	PEN	PEN flag-last condition-'up,' 'down,'	VPLOT
	R(3)	Vector from viewer, X, Y projections are plotted	VPLOT
	SCALE	½ angle or field of view	PIC2
	R2	Range S/C to planet center	VVIEW
	VT(3)	View coord axis (horizontal) unit	VVIEW
	VR(3)	View coord axis (vertical) unit	VVIEW
	VW(3)	View coord axis (to or from) viewpoint unit	VVIEW
	RPP	Radius of planet (km)	PIC2
	TANR	Tan. of ½ angle or fov during viewing	PIC2
/PLTBUF/	BUF(20)	Plotting buffer array	PIC2
/POINT	VC(3,9)	Vector from planet CTR to frame points in ecliptic coord.	LOCATE
	P(3,9)	Vector to 9 points on TV frame (camera coord.)	FRAME

Table 4-1 (concl)

COMMON BLOCK	VARIABLE SYMBOL	DEFINITION OF VARIABLE	LOADING SUBROUTINE
/POBEAR/	SSV(8)	Phase angle for 8 possible satellites	CALC
	SAF(8)	½ angle of 8 possible satellites	CALC
	HANG2	Angle subtended by planet	CALC
	DCNCLK(8)	Delta cone and clock - required adjustment	CALC
/PPOLY/	AA(12,2)	Coefficients for Polynomial Planet Orbital Constants: Major axis	PLCON
	EE(12,3)	Eccentricity	PLCON
	II(12,3)	Inclination	PLCON
	OO(12,3)	Arguments of Ascending Node	PLCON
	WW(12,3)	Longitude of Periapsis	PLCON
	MM(12,3)	Mean Anomaly	PLCON
	VV(12,2)	Prime Meridian	PLCON
/PTLB/	THA	Half angle of planet/satellite (radians)	GEOPLT
	R(3)	Range, cone & clock to planet/satellite	GEOPLT
	DEI	Inches per degree - plotting scale	GEOPLT
/RAZOR/	PICS(4,9)	Number of pictures over given scale	MRCL
/REVER/	ROTW(3)	Rotation velocity vector on S/C	MAIN
/RING/	RR(5)		
/RFSTAR/	C(3)	Vector to Canopus	MAIN
/STUF1/	XSAT(3)	Vector to nearest satellite	CALC
/STUFF/	X(3)	Position vector to S/C from planet/sat	CALC
	TARGE(3)	Velocity vector of pic center	CALC
	ROTV(3)	Rotation velocity vector due to S/C rates	CALC
	VSV(3,8)	Vector to satellite from S/C camera coord.	CALC
	VVSAT(3,8)	Velocity vector of satellite toward S/C camera coord.	CALC
	SVSV(3,8)	Velocity of picture of satellite due to S/C rotation	CALC
	RCCL(3)	Unitvector to pic ctr from S/C xyz sun/Earth-Canopus	CALC
	DTARE(3)	= TARGE if planet;=VVSAT if satellite	CALC
	DROTV(3)	= ROTV if planet;= SVSV if satellite	CALC
/TCONS/	TCONS	Time after periapsis (days)	INPUT
/TEST/	ITEST(97)	Indicator or user inputs = 1 if input	INPUT
/TNME/	TVNAME(5)	Instrument name	INPUT
/TRA/	TRACE	"Trace" for debugging	
/VCONS/	UX(3)	Unit vector along x-axis	VECTOR
	UY(3)	Unit vector along y-axis	VECTOR
	UZ(3)	Unit vector along z-axis	VECTOR
/VAR/	VAR(97)	Name list of input variables	INPUT
/VW/	VW(3)	View vector toward/from selected point	PIC2
	RIP	Range S/C to planet center	PIC2
	ASDP	½ angle or planet satellite or switches	PIC2
/XZ/	XZ(97)	Input variable array (see Table 4-2)	INPUT
	IXZ6	Integer specification of input state	MAIN.
/BRTC/	IV(3)/ V(3)	Planetocentric vector to picture on planet surface	FPLOT

Table 4-2 Input Variable Identification - Common XZ (97)

CARD NUMBER	DESCRIPTION
Header Card	Required, prints run title on each page.
XZ (1)	Not used
XZ (2)	Planet Number*
XZ (3)	Satellite number† if var. EQ. 0 and minimum range (86) has been specified, data will be generated for "closest" satellite within minimum range. If var. EQ. 0 and (86) has been specified, data will be generated for specific satellite number within minimum range. If 86 has not been specified only geometry data for specified satellite will be generated.
XZ (4)	Date of encounter (periapsis)
XZ (5)	Not used
XZ (6)	Input form of state vector - uses ORBIT convention.
XZ (7)	State vector see var. (6)
	State vector see var. (6)
	State vector see var. (6)
	State vector see var. (6)
	State vector see var. (6)
XZ (12)	State vector see var. (6)
XZ (13 - 17)	Not used
XZ (18)	Spacecraft model var = -1 sun oriented (TOPS)
	var = 0. Earth oriented (TOPS)
	var = 1. Earth oriented (PIONEER)
	var = 2. input spin rates (EARTH ORIENT)
XZ (19)	Pitch rate (deg/sec)
XZ (20)	Yaw rate (deg/sec)
XZ (21)	Roll rate (deg/sec)
XZ (22)	Instrument ID 110. - Narrow angle (SIT) TOPS (Can. Val)
	120. - Wide angle (SIT) TOPS
	130. - Wide angle Silicon Vidicon
	140. - Wide angle Silicon Vidicon
	150. - Wide angle SEC
	160. - Wide angle SEC

* Sun = 0, Mercury = 1, . . .

† Satellites are numbered from 1 to N starting with innermost satellite

Table 4-2 (cont)

CARD NUMBER	DESCRIPTION
XZ (22) - Cont	Instrument ID 210 - Particle and fields <u>plots</u> only 310 - IR (not available) 410 - UV (not available)
XZ (23)	Variable specifies mode description (aligns picture center to radius vector or to center of lit limb if radius vector is in shadow). 0 - uses built-in time steps 1 - input discrete time of picture, cone, clock of camera center line
XZ (24)	Cone angle } Input only if Var 23 = 1
XZ (25)	Clock angle } Input only if Var 23 = 1
XZ (26)	Time from encounter } Input only if Var 23 = 1
XZ (27)	Not used
XZ (28)	Not used
XZ (29)	Not used
XZ (30)	Not used
XZ (31 - 70)	Instrument characteristics (see Tables 4-3 and 4-4
XZ (81)	Plot interval [number of calculation steps between plots (TV)]
XZ (82)	Print interval, number of calculation steps between print
XZ (83)	Not used
XZ (84)	Not used
XZ (85)	Flag to specify a satellite run satellite specified by var (3) If 85 = 1 it is a satellite run 85 = 0 it is a planet run - can switch to Satellite if XZ (86) min range is loaded
XZ (86)	Minimum range required before switching to satellite
XZ (87)	Start of time interval to inhibit satellite switch if minimum range specified.
XZ (88)	End of time interval to inhibit satellite switch if minimum range is specified. Units = <u>Hours</u>.
XZ (89)	Not used

Table 4-2 (concl)

CARD NUMBER	DESCRIPTION
XZ (90)	Flag to specify plot output. 1. Cone clock of planet limb and specified satellite 10. Resolutions 100. Smears 1000. Numbers of pictures with resolution greater than a specified value; var (41-50) 10000. Satellite phase angles, ranges, cone and clock angles 100000. Sun-satellite-S/C angles, planet cone and clock, and planet range
XZ (91 - 97)	Not used.

Table 4-3 Input Variable Identification - TV Characteristics

CARD NUMBER	DESCRIPTION
XZ (31 thru 35)	Instrument name
XZ (36)	Focal length, cm
XZ (37)	Clear aperture diameter, cm
XZ (38)	Obscuration diameter, cm
XZ (39)	Optics transmission, Refl + absorp, fraction Sensor dimensions, cm
XZ (40)	Vertical, across scan lines
XZ (41)	Horizontal, along scan line
XZ (42)	Number of scan lines
XZ (43)	Number of pixels per scan line
XZ (44)	Number of bits per pixel
XZ (45)	Minimum detectable exposure, ergs/cm^2
XZ (46)	Maximum or highlight exposure, ergs/cm^2
XZ (47)	Thru 50, minimum resolution levels
XZ (51)	Sensor response curve identification
XZ (52)	Blue filter transmission curve ID
XZ (53)	Green filter
XZ (54)	Red filter
XZ (55)	Polarizing filter
XZ (56)	Exposure time
XZ (57)	Thru 58 open
XZ (59)	Frame read time, seconds
XZ (60)	Frame erase time, seconds
XZ (61)	Picture taking start time (days)
XZ (62)	Picture taking stop time (days)
XZ (63)	Delta T per picture (min) - can be changed by program (see section 5.2)
XZ (64)	Bulk storage capacity (bits)
XZ (65)	Bulk zero state (bits)
XZ (66)	Transmission bit rate, BPS
XZ (67)	Other instrument bit rates
XZ (68)	Buffer capacity (bits)
XZ (69)	Pictures per calculation step
XZ (70)	Delta T per picture (min) fixed

Table 4-4 Input Variable Identification - Fields and Particles

CARD NUMBER	DESCRIPTION
36	Magnetopause range (radii) (km)
37	Bow shock range (radii) (km)
38	Latitude mag pole (deg)
39	Longitude mag pole (deg)
40	Sensitivity of magnetometer (gauss)

Table 4-5 Variable to Tape Identification

TAPE NO.	VARIABLE	DEFINITION	LOADED BY	READ BY
13	TT	Time from periapsis (sec)	CACL	
	X(3)	Vector to S/C from planet	↓	OTJAZ
	XP(3)	Vector to planet from ref body	↓	OTJAZ
	XSAT(3)	Vector to satellite of interest	↓	
	XE(3)	Vector to Earth from sun	↓	
	R(3)	Range, cone and clock to planet	↓	OTPLT
	PHASE	Phase angle of planet (radians)	↓	OTPLT
	RADP(3)	Vector to picture ctr from S/C		
	TARGE(3)	Velocity vector of pic center (camera coord)		
	TARGP(3)	Latitude & longitude of pic center		
	RSAT(3)	Range cone & clock of satellite		OTPLT
	PHASES	Phase angle of satellite (rad)		OTPLT
	XSS(3)	Vector to S/C from satellite		
	JDV	Julian date of interest		
	EM	Emission angle		
8	TX	Time from periapsis (days)	FIELDS	FIELDS
	BMINS	Minimum detectable B in gauss	↓	↓
	DIRB(3)	Mag, cone & clock magnetic flux vector		
9	T	Time from periapsis (days)	OTJAZ	OTJAZ
	SSV(8)	Phase angle, 8 possible satellites	↓	↓
	SAF(8)	Half angle, 8 possible satellites	↓	↓
	HANG2	Angles subtended by planet	↓	↓
	DCNCLK	Sum of angular change to switch to sat.	↓	↓
10	TX	Time from periapsis (days)	SMEAR	RESO/SMEAR
	REC(9)	Resolution at center of planet/sat..	↓	RESO
	REL(9)	Resolution at limb of planet/sat.	↓	RESO
	RES	Resolution at center of primary body	↓	RESO
	SMR(9)	Vel of pic ctr due to orbit		SMEAR
	SMRR(9)	Smear in pixels planet/sat.		↓
	SMRS(9)	Vel of pic ctr due to S/C rotation		↓
	SPSMR	Vel due to orbit (principal body)		↓
	SPSMRS	Vel due to S/C rotation (principal body)		↓
	SPSMRR	Smear in pixels (principal body)		SMEAR
11	T	Time from periapsis (days)	OTJAZ	OTJAZ
	PHASE (8)	Phase angle at 8 possible satellites	↓	↓
	XRCC(3,8)	Range, cone & clock to 8 satellites		
12	TP	Time from periapsis (days)	MRLC	MRLC
	PICS	Acceptable pictures (reso better than)	MRLC	MRLC

5. MODEL OPERATING PROCEDURES

5.1 Comment Cards Contained in Program - The operating procedures are contained on comment cards as "User Instructions for Program Input Requirements" at the front of the program. Table 5-1 is a listing of the comment cards.

The user should give special attention to the mandatory inputs and those input variables that shall be used if not overridden by input (canned or default values). The mandatory inputs are marked on the comment card by an asterisk immediately preceding the input code number. All "canned" values are listed in the user instruction comment cards. When a run is made, a listing of all input used by the program is made, and the user input values are defined by comment (Appendix A). All other input used by the program was by "default," since none was specified by the user.

5.2 Special Imaging Options - Because of the emphasis on imaging, the following discussion is included to illustrate the optional analysis modes available. The input described below does not eliminate the mandatory input described above. The three general options, with suboptions indicated, are as follows:

1) Purpose: Mission Simulation

Input Number	Input Description
61	Image start time (days)
62	Image stop time (days)
63	Delta time per calculation step* (min)
69	Number of pictures/calculation step**
22	Instrument Identification Code
Optional Inputs	
18	Spacecraft Identification Code
19	Spacecraft Rates ($\dot{x}$)
20	($\dot{y}$)
21	($\dot{z}$)

*Number of calculation steps must not exceed 200

**Allows calculations every Nth picture, i.e., N = 10, allows 2000 pictures

Table 5-1 Comment Cards

```
TOPS, (J-S-P) JUPITER ENCOUNTER, 10. DEG FOV SIT CAMERA.

 2        5.0000000     PLANET JUPITER (J-S-P)
 3        2.0000000     SATEL  EUROPA (J-II)
 4        790114.69     DATE       AT ENCOUNTER (SEE APPENDIX C FOR
 6        102.00000                              DEFINITION OF DATE FORMAT)
 7        127064.46     STATE1
 8        213218.85     STATE2       PERIAPSIS VECTOR FROM LEADBETTERS
 9       -13387.052     STATE3       79 JSP RUN TO MAXIMIZE SATELLITE
10       -29.431552     STATE4       ENCOUNTER
11        17.610963     STATE5
12        1.1418359     STATE6
22        120.00000     INSTIDINSTRUMENT ID (TV,  WIDE ANGLE SIT)
23        1.0000000
26       -.22200000
24        151.65000
25        254.70000
26       -.16700000
24        141.34000
25        254.62000
26       -.11100000
24        125.66000
25        254.54000
26      -5.49999997-02
24        100.92000
25        254.46000
26       0.00000000
24        65.679999
25        254.37000
26        5.49999997-02
24        30.440000
25        247.18000
26        .11100000
24        5.7100000
25        224.92000
26        .16700000
24        9.9599999
25        38.300000
26        .22200000
24        20.270000
25        70.049999
45        7.99999998-06CHAR10SIT MIN DETECTABLE EXP
46        1.99999998-03CHAR11SIT HIGHLIGHT EXP
51        1.0000000            NO
61       -1.5000000     TSTAR TIME FROM PERIAPSIS
62        1.5000000     TSTOP TIME FROM PERIAPSIS
63        5.0000000     DTINSTPICTURE TAKING INTERVAL (MIN)
64        1.00000000+09        TORAGE CAPACITY
65       0.00000000            BULK STORAGE ZERO STATE
66        47400.000     BITRATTRANSMIT BIT RATE  HALF FOR EACH CAMERA
67        640.00000            DATA RATE FROM OTHER INSTRUMENTS
68        2000000.0            BUFFER STORAGE CAPACITY
69        6.0000000     CALCSTPICTURES PER CALC STEP
81        72.000000     DPLOT CALC STEPS PER PLOT  1.5 DAYS PER PLOT
82        1.0000000     DPRNT CALC STEPS PER PRINT
90        1111111.0
```

Optional
Camera
Inputs

31 - 70.

When the Δt between pictures (i.e., Δt per calculation step/# pictures per calculation step) is less than the picture rate allowable, as determined by the bulk storage and data bit rate, the frame read time, or (in the case of a spinning spacecraft) the spin rate (one picture/revolution) input (21), the Δt between pictures is reset to be consistent with spacecraft capabilities.

2) Purpose: Provide Summary and Graphic Data of TV Pictures

Input Number	Input Description	
22	Instrument identification	
23	Flag to identify discrete input	
24	Cone angle	Up to 200 *SETS*,
25	Clock angle	cone, and clocks to
26	Time from periapsis	*PLANET ONLY*
Optional Inputs		
18	Spacecraft ID	
19	Spacecraft rates ($\dot{x}$)	
20	($\dot{y}$)	
21	($\dot{z}$)	
31 - 70	Camera inputs	

3) Purpose: To provide first cut information. Data compressed about periapsis (generates 40 data points)

Input Number	Input Description
22	Instrument identification
61	Start and stop time from periapsis (days)
Optional Inputs	
18	Spacecraft ID
19	Spacecraft rates ($\dot{x}$)
20	($\dot{y}$)
21	($\dot{z}$)
31 - 70	Camera input

The inputs described below are associated with the available target viewing options. The option matrix describes the combination of inputs that yields the desired targets.

Input Number	Description
03	Satellite number
85	Flag to specify satellite as principal body for entire run
86	Minimum range at which camera switches to satellite
87	Start inhibit camera switch to satellite*
88	Stop inhibit camera switch to satellite*

*The program will not switch to the satellite during the time span indicated between inputs 87 and 88.

Option Matrix

Input Number	Planet Data Only	Switch to Specified Satellite	Switch to Closest Satellite	Satellite as Viewed Body
03	Not used	Satellite number	Not used	Satellite number
85	Not used	Not used	Not used	1
86	Not used	Minimum range	Minimum range	Not used
87	Not used	Not used	Start inhibit time	Not used
88	Not used	Not used	Stop inhibit time	Not used

5.3 <u>Example of Computer Input Cards</u> - Table 5-2 is an example of an actual data presentation model run (verification run 2-2 as shown in Appendix A). In the input listing, cards 2, 3, and 4 provide the planet index number (5 for Jupiter), the satellite index number (2 for Europa) and the calendar date (79011469 meaning 69/100 days past the start or January 14, 1979).

Input 6 (with a value of 102) describes the six following numbers to be the planetocentric position vector components (km) and the velocity vector components (km/sec) referenced to the ecliptic and the vernal equinox of 1950. (A detailed description of variable 6 is given in JPL Technical Memorandum 293-10, October 13, 1970, "Orbit, A general Purpose Conic Orbit Program" by P. H. Roberts, including applicable revisions).*

Card 22 states that the instrument identification number is 120, providing canned input TV characteristics for a SIT, wide-angle camera. Card 23 having a value of 1 indicates that the user will input cone and clock angles for TV pointing at a specific number of days from periapsis.

Nine sets of cards numbered 24, 25 and 26 follow, with cards 24 listing cone angle in degrees, cards 25 listing clock angle in degrees, and cards 26 listing time from periapsis in days. The first time input is -.222, which means 0.222 days (5 hours and 20 minutes) before periapsis.

*This reference is included in Appendix C Vol II

Table 5-2 User Instructions

```
C***************************************************************************
C***************************************************************************
C
C
C        USER INSTRUCTIONS FOR PROGRAM INPUT REQUIREMENTS
C
C
C***************************************************************************
C***************************************************************************
C
C
C     01            NOT USED
C
C *   02   IP       IS THE NUMBER OF THE PLANET TO BE CONSIDERED
C
C     03   IVS      IS THE NUMBER OF THE SATELLITE TO BE CONSIDERED  (SEE 86)
C
C *   04   DATE     IS THE TIME OF PERIAPSIS OR ENCOUNTER. DATE.GE.0 MEANS
C                   DATE=10000*YR + 100*MO + DAY. WHERE YR IS THE LAST TWO
C                   DIGITS OF THE YEAR (1900 TO 1999). MO IS THE MONTH NUMBER,
C                   JAN = 1. AND DAY IS THE DAY OF THE MONTH INCLUDING FRACTIONAL
C                   DAYS.
C
C     05            NOT USED
C
C *   06   A(2)     INPUT FORM OF ORBIT (THE ORBIT CONVENTION IS USED. REF
C                   JPL TECHNICAL MEMORANDUM 393-10 OCTOBER 13, 1970
C                   AUTHOR = P. H. ROBERTS )
C
C *   07   A(3)     ORBIT INPUT X(1)
C
C *   08   A(4)     ORBIT INPUT X(2)
C
C *   09   A(5)     ORBIT INPUT X(3)
C
C *   10   A(6)     ORBIT INPUT X(4)
C
C *   11   A(7)     ORBIT INPUT X(5)
C
C *   12   A(8)     ORBIT INPUT X(6)
C
C     13            NOT USED
C
C     14            NOT USED
C
C     15            NOT USED
C
C     16            NOT USED
C
C     17            NOT USED
C
C     18   NSC      SPACE CRAFT MODEL
C                   = -1.  SUN ORIENTED (TOPS)
C                   =  0.  EARTH ORIENTED (TOPS)
C                   =  1.  EARTH ORIENTED (PIONEER)
C                   =  2.  EARTH ORIENTED - SPIN RATES INPUTS 19 THRU 21
C
C     19   ROTW(1)=  PITCH RATE (DEG/SEC)
C
C     20   ROTW(2)=  YAW RATE (DEG/SEC)
C
C     21   ROTW(3)=  ROLL RATE (DEG/SEC)
```

Table 5-2 (cont)

```
C
C *   22  INST    INSTRUMENT ID
C                 = 110. NARROW ANGLE SIT (TOPS)
C                 = 120. WIDE ANGLE SIT (TOPS)
C                 = 130. NARROW ANGLE SILICON VIDICON
C                 = 140. WIDE ANGLE SILICON VIDICON
C                 = 150. NARROW ANGLE SEC VIDICON
C                 = 160. WIDE ANGLE SEC VIDICON
C                 = 210. FIELDS AND PARTICLES PLOTS ONLY
C                 = 310. IR (NOT AVAILABLE)
C                 = 410. UV (NOT AVAILABLE)
C
C     23  IMODE   MODE DESCRIPTION
C                 =0.USES BUILT IN TIME STEPS (UNLESS OVERRIDDEN BY TV
C                    CHARACTERISTICS 63 AND 69). PICTURE CENTER WILL BE
C                    ALIGNED WITH RADIUS VECTOR OR TO CENTER OF LIT LIMB
C                    IF RADIUS VECTOR IS IN THE SHADOW.
C                 =1.USER WILL INPUT DISCRETE TIME FROM ENCOUNTER (26)
C                    CONE ANGLE (24) AND CLOCK ANGLE (25)
C
C     24  CCONE   CONE ANGLE (DEG)
C
C     25  CCLOCK  CLOCK ANGLE (DEG)
C
C     26  TT      TIME FROM ENCOUNTER (DAYS)
C                 USER MAY INPUT UP TO 200 CARD SETS 24,25 AND 26
C
C     27          NOT USED
C
C     28          NOT USED
C
C     29          NOT USED
C
C     30          NOT USED
C
C     31-70 INSTRUMENT CHARACTERISTICS
C
C
C
C          TV ELEMENTS
C
C
C     31  TVNAME(1) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     32  TVNAME(2) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     33  TVNAME(3) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     34  TVNAME(4) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     35  TVNAME(5) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     36  P(6)    FOCAL LENGTH (CM)
C
C     37  P(7)    CLEAR APERATURE DIA. (CM)
C
C     38  P(8)    OBSCURATION DIA. (CM)
C
C     39  P(9)    OPTICS TRANSMISSION FRACTION
C
C     40  P(10)   VERTICAL DIMENSION (CM), ACROSS SCAN LINES
C
```

Table 5-2 (cont)

```
C     41  P(11)  HORIZONTAL DIMENSION (CM), ALONG SCAN LINES
C
C     42  P(12)  NUMBER OF SCAN LINES
C
C     43  P(13)  NUMBER OF PIXELS PER SCAN LINE
C
C     44  P(14)  NUMBER OF BITS PER PIXEL
C
C     45  P(15)  MINIMUM DETECTABLE EXPOSURE (ERGS/CM**2)
C
C     46  P(16)  MAXIMUM OR HIGHLIGHT EXPOSURE (ERGS/CM**2)
C
C     47  P(17)  MINIMUM RESOLUTION LEVEL 1 (KM/PIXEL)
C
C     48  P(18)  MINIMUM RESOLUTION LEVEL 2 (KM/PIXEL)
C
C     49  P(19)  MINIMUM RESOLUTION LEVEL 3 (KM/PIXEL)
C
C     50  P(20)  MINIMUM RESOLUTION LEVEL 4 (KM/PIXEL)
C
C     51  P(21)  SENSOR RESPONSE CURVE IDENTIFICATION
C
C     52  P(22)  BLUE FILTER TRANSMISSION CURVE IDENTIFICATION
C
C     53  P(23)  GREEN FILTER TRANSMISSION CURVE IDENTIFICATION
C
C     54  P(24)  RED FILTER TRANSMISSION CURVE IDENTIFICATION
C
C     55  P(25)  POLARIZING FILTER TRANSMISSION CURVE IDENTIFICATION
C
C     56  P(26)  EXPOSURE TIME (SEC)
C
C     57  P(27)  NOT USED
C
C     58  P(28)  NOT USED
C
C     59  P(29)  FRAME READ TIME (SEC)
C
C     60  P(30)  FRAME ERASE TIME (SEC)
C
C     61  P(31)  PICTURE TAKING START TIME (DAYS
C
C     62  P(32)  PICTURE TAKING STOP TIME (DAYS)
C
C     63  P(33)  DELTA T PER PICTURE (MIN) -WILL OVERRIDE CANNED TIME STEP
C
C     64  P(34)  BULK STORAGE CAPACITY (BITS)
C
C     65  P(35)  BULK ZERO STATE (BITS)
C
C     66  P(36)  TRANSMISSION BIT RATE (BPS)
C
C     67  P(37)  OTHER INSTRUMENT BRT RATES (BPS)
C
C     68  P(38)  BUFFER CAPACITY (BITS)
C
C     69  P(39)  PICTURES PER CALCULATION STEP (SEE 63 FOR SIZE OF PROGRAM STEP)
C
C     70  P(40)  NOT USED
C
C
C
```

Table 5-2 (cont)

```
C              FP ELEMENTS
C
C
C     31  TVNAME(1) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     32  TVNAME(2) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     33  TVNAME(3) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     34  TVNAME(4) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     35  TVNAME(5) INSTRUMENT NAME 6 ALPHANUMERIC CHARACTERS
C
C     36  P(6)    MAGNETOPAUSE RANGE (PLANET RADII)
C
C     37  P(7)    BOW SHOCK RANGE (PLANET RADII)
C
C     38  P(8)    LATITUDE OF MAGNETIC POLE (DEG)
C
C     39  P(9)    LONGITUDE OF MAGNETIC POLE (DEG)
C
C     40  P(10)   INSTRUMENT SENSITIVITY (GAUSS)
C
C     41 THRU 70 NOT USED
C
C
C     71 THRU 80 NOT USED
C
C     81   IFLPL  PLOT INTERVAL NUMBER OF CALCULATION STEPS BETWEEN PLOTS
C
C     82   IFLPR  PRINT INTERVAL NUMBER OF CALCULATION STEPS BETWEEN PRINT OUTS
C
C     83          NOT USED
C
C     84          NOT USED
C
C     85   IFLT   SATELLITE FLAG. IF VARIABLE IS INPUT GE 1. AND VARIABLE 3
C                 IS INPUT GE 1. THAN TV DATA WILL BE GENERATED ONLY FOR THE
C                 SATELLITE SPECIFIED BY VARIABLE 3.
C
C     86   XZ(86) MINIMUM RANGE TO START SATELLITE SWITCH
C                 NOTE 1. IF VARIABLE 3 IS GE 1. AND VARIABLE 86 IS GT 0.
C                         THAN TV DATA WILL BE GENERATED FOR THIS SATELLITE (3)
C                         WHEN ITS RANGE IS LESS THAN VARIABLE 86.
C                 NOTE 2. IF VARIABLE 3 IS EQ 0. AND VARIABLE 86 IS GT 0.
C                         THAN TV DATA WILL BE GENERATED FOR THE CLOSET
C                         SATELLITE WITH RANGE LESS THAN VARIABE 86.
C
C     87          START TIME FROM ENCOUNTER (HRS) DURING WHICH SATELLITE
C                 SWITCHING WILL BE INHIBITED
C
C     88          STOP TIME FROM ENCOUNTER (HRS) DURING WHICH SATELLITE
C                 SWITCHING WAS INHIBITED.
C
C     89          NOT USED
C
C     90   ZOUT   PLOT FLAG. SIX DIGIT NUMBER
C                 DIGIT 1 GE 1.  FOR PLOT OF CONE-CLOCK OF PLANET LIMB
C                                AND SPECIFIED SATELLITE (3)
C                 DIGIT 2 GE 1.  FOR TV RESOLUTION PLOTS
C                 DIGIT 3 GE 1.  FOR TV SMEAR PLOTS
C                 DIGIT 4 GE 1.  FOR "PICTURES WITHIN SPECIFIED RESOLUTION LEVELS"
```

Table 5-2 (cont)

```
C                                PLOTS
C                  DIGIT 5 GE 1.  FOR PLOTS OF SATELLITE RANGE, EARTH-SAT-S/C
C                                 ANGLE, SATELLITE CONE ANGLE, SATELLITE CLOCK
C                                 ANGLE, SUN-SAT-S/C ANGLE, AND SATELLITE HALF
C                                 ANGLE VS TIME.
C                  DIGIT 6 GE 1.  FOR PLOTS OF PLANET RANGE, EARTH-PLANET-S/C
C                                 ANGLE, PLANET CONE ANGLE, AND PLANET CLOCK
C                                 ANGLE VS TIME.
C
C *   96   XZ(96) RESTART FLAG. VARIABLE EQ 0 FOR FOLLOWING CARDS TO BE CHANGES
C                 OR ADDITIONS ONLY. VARIABE EQ 1 AND ALL PREVIOUS DATA WILL BE
C                 DELETED BEFORE NEXT CASE IS RUN. VARIABLE GT 1 FOR PROGRAM
C                 TERMINATION.
C
C     97          DO NOT USE
C
C *   98   XZ(98) FLAG TO SPECIFY END OF DATA CASE
C
C *   99   XZ(99) FLAG TO TERMINATE ENTIRE PROGRAM
C
C
C * NOTE -------- ASTERISK MEANS THAT THIS VARIABLE MUST BE PRESENT FOR
C                 THE CORRECT RUNNING OF THE PROGRAM
C
C
C
C***********************************************************************
C***********************************************************************
C
C
C         USER INSTRUCTIONS FOR PROGRAM OPERATION
C
C
C***********************************************************************
C***********************************************************************
C
C
C 1. NOTE -------- A HEADER CARD WHICH MAY BE WRITTEN IN ANY FORMAT MUST
C                 PROCEED EACH DATA EVEN IF THAT DATA CASE IS ONLY A 99
C                 CARD. THE INFORMATION PUNCHED ON THE HEADER CARD WILL BE
C                 PRINTED OUT AT THE TOP OF EACH PAGE OF OUTPUT.
C
C 2. NOTE ------- THE FORMAT FOR ALL DATA CARDS WITH THE EXCEPTION OF THE
C                 HEADER CARD IS AS FOLLOWS
C
C              FORTRAN
C     COL      FORMAT         DESCRIPTION
C
C     1- 2      I2            TWO DIGIT VARIABLE NUMBER
C     3-17      G15.8         VARIABLE (FLOATING NUMERIC DATA ONLY)
C    18-23      A6            VARIABLE (6 ALPHANUMERIC CHARACTERS)
C    24-77      9A6           COMMENT  (ALPHANUMERIC DATA)
C    78-79      A2            COMMENT  (ALPHANUMERIC DATA)
C       80                    NOT USED
C
C
C 3. NOTE ------- WHEN A VARIABLE IS NOT SPECIFIED BY A DATA CARD THE PROGRAM
C                 WILL USE THE FOLLOWING CANNED VALUES
C
C     03      0.0      SEE SATELLITE OPTION INPUT 86
C     18      0.0      EARTH ORIENTED TOPS S/C
C     19      0.01     PITCH RATE (DEG/SEC)
```

Table 5-2 (cont)

```
C     20      0.01     YAW RATE (DEG/SEC)
C     21      0.01     ROLL PATE (DEG/SEC) IF 18=-1. OR 0.
C     21     90.00     ROLL RATE (DEG/SEC) IF 18= 1.
C     23      0.0      RADIUS POINTING MODE
C
C     IF INST (XZ(22)) = 110.
C
C     31      SIT NA
C     32      RROW A
C     33      NGLE C
C     34      AMERA
C     35      (CAND)
C     36      200.0
C     37       23.0
C     45        8.E-6
C     46       20.E-4
C     51        1.0
C     56        0.001
C     57        0.0
C     58        0.0
C     59       40.0
C     60        0.0
C
C     IF INST (XZ(22)) = 120.
C
C     31      SIT WI
C     32      DE ANG
C     33      LE CAM
C     34      ERA
C     35      (CAND)
C     36       20.0
C     37        5.0
C     45        8.E-6
C     46       20.E-4
C     51        1.0
C     56        0.0001
C     57        0.001
C     58        0.010
C     59       40.0
C     60        0.1
C
C     IF INST (XZ(22)) = 130.
C
C     31      SI NAR
C     32      ROW AN
C     33      GLE CA
C     34      MERA
C     35
C     36      200.0
C     37       23.0
C     45        2.E-3
C     46        0.5
C     51        1.0
C     56        0.05
C     57        0.1
C     58         .5
C     59       40.0
C     60        0.1
C
C     IF INST (XZ(22)) = 140.
C
C     31      SI WID
```

Table 5-2 (cont)

```
C     32        E ANGL
C     33        E CAME
C     34        RA
C     35
C     36        20.0
C     37         5.0
C     45         2.E-3
C     46         0.5
C     51         1.0
C     56         0.05
C     57         0.1
C     58         0.5
C     59        40.0
C     60         0.1
C
C     IF INST (XZ(22)) = 150.
C
C     31        OPE 2
C     32        DEG RO
C     33        V CAME
C     34        RA SEC
C     35          TUBE
C     36        50.8
C     37         7.62
C     45          .0001
C     46          .01
C     51         1.0
C     56         0.4
C     57         0.1
C     58         0.01
C     59        40.0
C     60         0.1
C
C     IF INST (XZ(22)) = 160.
C
C     31       SEC WI
C     32       DE ANG
C     33       LE CAM
C     34       ERA
C     35
C     36         1.524
C     37        10.16
C     45         0.0001
C     46         0.01
C     51         1.0
C     56         0.1
C     57         0.01
C     58         0.001
C     59        40.0
C     60         0.1
C
C     FOR ALL TV INSTRUMENTS
C
C     38         0.0
C     39         0.85
C     40         1.60
C     41         1.60
C     42       800.0
C     43       800.0
C     44         8.0
C     47       300.0
C     48       100.0
```

Table 5-2 (concl)

```
C    49       50.0
C    50       10.0
C    52        1.0
C    53        1.0
C    54        1.0
C    55       -1.0
C    61       -1.5
C    62        1.5
C    63       15.0
C    64        1.E+9
C    65        0.0
C    66      474.E+2
C    67      100.
C    68      P(14)*P(12)*P(13)
C    69        4.
C    70      P(33)
C
C
C     IF INST = 210.
C
C    31      FIELDS
C    32       AND P
C    33      ARTICL
C    34      E INST
C    35      RUMENT
C    36       51.6  (JUPITER),  63.0  (SATURN)
C    37       65.8  (JUPITER),  80.3  (SATURN)
C    38       80.0  (JUPITER),  90.0  (SATURN)
C    39      190.0  (JUPITER),   0.0  (SATURN)
C    40        2.E-6
C
C
C    81        1.0
C    82        1.0
C    85        0.0
C    86        0.0
C    87        0.0
C    88        0.0
C    90        0.0
C
C
C*******************************************************************************
C*******************************************************************************
C
C
C         END OF USER INSTRUCTIONS
C
C
C*******************************************************************************
C*******************************************************************************
```

The next input variables are found on card 45, which gives the minimum detectable exposure (8×10^{-6} ergs/cm^2), and on card 46, which gives maximum or high light exposure (2×10^{-3} ergs/cm^2). Card 51 provides for the use of sensor curve number 1, while cards 61 and 62 establish a start and stop time of ±1.5 days. Card 63 provides for an interval of 5 minutes between pictures. Cards 64 through 68 describe the spacecraft memory storage capacity as a bulk storage of 10^8 bits with no bits stored initially, a bit transmission rate allocated to this camera of 47,400 bits per second requiring transmission from other instruments, and a buffer storage capacity of 2×10^6 bits.

Card 69 provides for six pictures per calculation step, and cards 81 and 82 provide for 72 calculation steps from -1.5 to +1.5 days, and a data printout for each calculation.

Card 90 calls for all possible summary plots to be plotted.

6. MODEL SUBROUTINES

One hundred six subroutines are employed by the main program to constitute the model itself. Since there is clear disparity between the subroutines in regard to their source and complexity, the following four classifications were established.

6.1 Subroutine Classification - The first classification includes routines supplied at the start of the contract by JPL. For this first classification, a simple descriptive sentence is given.

A second classification contains the major subroutines developed during the contract, the complexity of which requires a flow diagram and listing to aid in their understanding.

A third classification contains a few minor subroutines that are simple enough to be described by a short description. A listing is also included.

The last classification is a set of translation routines that accept the name of various CDC 6000 system plotting subroutines and perform the same functions on the Univac 1108 computer using the Univac system plotting routines.

6.2 Subroutines Supplied by JPL

Function ABSV

Purpose: To obtain the absolute value of a vector.

Calling Sequence: ABSV (A)

Function ANGV

Purpose: To determine the angle between two vectors.

Calling Sequence: ANGV(A,B,I)

Function ANOMLY

Purpose: To convert between true, mean, and eccentric anomalies.

Calling Sequence: ANOMLY (EE, ANGLE)

Subroutine ARC

Purpose: Used for orthographic projections, when the picture frame falls off of the planet.

Calling Sequence: CALL ARC

Subroutine ARCSTP

Purpose: To establish the number of degrees in the arc to be plotted, by either subroutine PLTARC or subroutine SATARC, as a straight line.

Calling Sequence: CALL ARCSTP (S)

Subroutine CALNDR

Purpose: To convert between Julian, modified Julian, and calendar dates.

Calling Sequence: CALL CALNDR (DOUT, IO, DIN, II)

Subroutine CPLANN

Purpose: To obtain the equinox and pole vectors and the gravitational constant of the planet of interest.

Call Sequence: CALL CPLANN (IP)

Subroutine CTRANS

Purpose: To transform a vector in latitude and longitude coordinates to one of three alternate Cartesian systems.

Calling Sequence: CALL CTRANS (V, IV)

Subroutine DISK

Purpose: To plot the disk of an indicated planet (and rings of Saturn if applicable).

Calling Sequence: CALL DISK (IP, ISV, GRID, ADVANC, MOVIE)

Function DOT

Purpose: To determine the DOT product of two vectors.

Calling Sequence: DOT (A,B,V)

Subroutine ERR

Purpose: To enable trace routine, by setting the variable trace equal to 5HTRACE.

Calling Sequence: CALL ERR

Subroutine EUREKA

Purpose: To determine a vector from a planet center to the pierce point of a vector directed from the spacecraft. The resulting vector is given in two coordinate systems; ecliptic - equinox, and planetocentric - prime meridian.

Calling Sequence: CALL EUREKA (VQ, VC, PC, P)

Subroutine FEB

Purpose: To supply subroutine CALNDR with number of days in the month February.

Calling Sequence: CALL FEB.

Subroutine FIND

Purpose: To find the planetocentric puncture point of the camera frame on the planet's surface in prime meridian - pole vector system.

Calling Sequence: CALL FIND (VQI, LD, I)

Subroutine FPLOT

Purpose: To plot the angular limits of the TV frame field of view as projected on the planet surface and as seen from the spacecraft.

Calling Sequence: CALL FPLOT (VQ)

Subroutine FRAME

Purpose: To provide unit vectors to the corners, center of each edge, and frame center of a camera field of view.

Calling Sequence: CALL FRAME (P,FOV1,FOV2)

Subroutine JOVSAT

Purpose: To determine the position vector of a specific satellite of the planet Jupiter for a given input Julian date.

Calling Sequence: CALL JOVSAT (X,JP,IP)

Subroutine LOCATE

Purpose: To establish a camera coordinate system around the input cone and clock angles and to determine a vector from the planet center that intersects the cone and clock vector at the planet surface.

Calling Sequence: CALL LOCATE (VQ, CONE, CLOCK, TWIST)

Subroutine LOOK

Purpose: To determine the cone and clock angles to a given latitude and longitude on a planet surface giving a particular viewing direction and distance.

Calling Sequence: CALL LOOK (DUM, CONE, DLOCK, LAT, LONG)

Subroutine MERIDN

Purpose: To plot a given number of meridian circles on the planet disk.

Calling Sequence: CALL MERIDN (PV, PM, QL)

Subroutine NEPSAT

Purpose: To determine the position vector of a specific satellite of the planet Neptune for a given input Julian Date.

Calling Sequence: CALL NEPSAT (X,JD,IS)

Subroutine ORBIN

Purpose: To accept descriptions of the spacecraft orbit in various forms and provide a standard set of orbit elements for use of other subroutines.

Calling Sequence: CALL ORBIN (X,I)

Subroutine ORBPOS

Purpose: To determine the planetocentric position of spacecraft at a given time from periapsis.

Calling Sequence: CALL ORBPOS (X,TT)

Subroutine ORBVEL

Purpose: To determine the planetocentric velocity of spacecraft at a given time from periapsis.

Calling Sequence: CALL ORBVEL (X,TT)

Subroutine PARLEL

Purpose: To plot a given number of latitude parallels on the plot of the planet's disk.

Calling Sequence: CALL PARLEL (PV, PM, QL)

Subroutine PENUP

Purpose: To set pen control flag equal to "up" and to effect the blanking of the beam during the use of subroutine VPLOT.

Calling Sequence: CALL PENUP

Subroutine PGLOBE

Purpose: To determine a planetocentric ecliptic unit vector to zero latitude and zero longitude of a planet.

Calling Sequence: CALL PGLOBE (PM, JD, IP)

Subroutine PLANET

Purpose: To load the planet constants into the appropriate "COMMON" statements.

Calling Sequence: CALL PLANET (T)

Subroutine PLASAT

Purpose: To calculate position vector R(3) in planetocentric, equatorial, and ecliptic coordinates of 1950. Distance unit kilometers.

Calling Sequence: CALL PLASAT (R, JD, IS, IP)

Subroutine PLCON

Purpose: To provide constants relating the planet of interest or the sun and constants that allow determination of the planet's orbital elements.

Calling Sequence: CALL PLCON

Subroutine PLPOS

Purpose: To determine position of a planet with respect to the helicocentric eliptic and equinox of date.

Calling Sequence: CALL PLPOS (X,JD,IP)

Subroutine PLTARC

Purpose: To plot the visible points of the locus of tip of planetocentric vector as it rotates around an input vector axis and through an input angle.

Calling Sequence: CALL PLTARC (R,PV,ARC)

Subroutine PLVEL

Purpose: To determine heliocentric ecliptic velocity of a planet.

Calling Sequence: CALL PLVEL (V,JD,IP)

Subroutine PPLOT

Purpose: To plot an alphanumeric word (up to six characters) starting at a specified point.

Calling Sequence: CALL PPLOT (Y, X, BUF, LABLE)

Subroutine PRECS

Purpose: To adjust the pole vector of Jupiter to account for precession.

Calling Sequence: CALL PRECS (TC,X,Y)

Subroutine PSCONS

Purpose: To obtain constants of the various satellites of the planet.

Calling Sequence: CALL PSCONS (QUAN, QNAM, IS, IP)

Subroutine RCNCLK

Purpose: To transform a vector in ecliptic equinox coordinates to range and cone and clock angles referenced to the sun (or Earth)

Calling Sequence: CALL RCNCLK (R,S)

Subroutine SATARC

Purpose: To plot the visible points of the locus of the tip of a satellite-centered vector as it rotates through an input angle.

Calling Sequence: CALL SATARC (R,PV,RS,ARC)

Subroutine SATS

Purpose: To plot a range of points at regular time intervals along the orbits of the satellites of a planet as seen from a given viewpoint. A second option used by the program puts the position at a given time.

Calling Sequence: CALL SATS (SAT1, SATSL1, IP,ISU,JDV,DT)

Subroutine SATSAT

Purpose: To determine the position vector of a specific satellite of the planet Saturn for a given input Julian date.

Calling Sequence: CALL SATSAT (X,JD,IS)

Subroutine SCPRNA

Purpose: To annotate a plot with 48 possible characters, at a location and orientation that are flexible and controlled by various inputs.

Calling Sequence: CALL SCPRNA (BUS, IY, IX, J, AN LABLE, L)

Subroutine SDINIT

Purpose: To load the scaling factors, both absolute and relative, into the variable "BUF."

Calling Sequence: CALL SDINIT (BUF, I, RELU, ABSU)

Subroutine SDNPUT

Purpose: To provide a plotting facility for subroutine VPLOT.

Calling Sequence: CALL SDNPUT (Y,X,BUF,DUM)

Subroutine SHADER

Purpose: To shade the dark side of the plot of a planet with a pattern of dots in order to define the sun's terminator.

Calling Sequence: CALL SHADER (SHADE, SN)

Subroutine SINTRP

Purpose: To blank plotting beam by setting first value of the buffer array equal to zero.

Calling Sequence: CALL SINTRP (BUF)

Subroutine SOLPER

Purpose: To correct the position of Saturn's rings to account for solar pressure.

Calling Sequence: CALL SOLPER

Subroutine SPHERE

Purpose: To transform a vector between cartesian and spherical coordinates.

Calling Sequence: CALL SPHERE (V,A,I,J,K)

Subroutine SRINGS

Purpose: To plot five circles about Saturn to represent rings.

Calling Sequence: None

Subroutine STERM

Purpose: To close the plotting routines and terminate the program.

Calling Sequence: CALL STERM (BUF)

Subroutine TCONV

Purpose: To convert between time in days, hours, minutes, seconds, or mixed.

Calling Sequence: CALL TCONV (TT, Units, time, UNIN)

Subroutine UCROSS

Purpose: To perform a cross product between two vectors and obtain unit vector parallel to product.

Calling Sequence: CALL UCROSS (V,A,B)

Subroutine URASAT

Purpose: To determine the position vector of a specific satellite of the planet Uranus for a given input Julian date.

Calling Sequence: CALL URASAT (RR,DEJ,IS)

Subroutine VAXIAL

Purpose: To set a vector of any magnitude in the direction of one of the coordinate axes.

Calling Sequence: CALL VAXIAL (V,S,K)

Subroutine VCOMB

Purpose: To combine (add) two vectors multiplied by two scalars.

Calling Sequence: CALL VCOMB (V,A,SA,B,SB)

Subroutine VCROSS

Purpose: To perform a cross product multiplication between two input vectors.

Calling Sequence: CALL VCROSS (V,A,B)

Subroutine VDIF

Purpose: To determine the value of a vector, V, which is the difference between two other vectors, A and B.

Calling Sequence: CALL VDIF (V,A,B)

Subroutine VECTOR

Purpose: To load unit vectors into the appropriate COMMON statement and load the variable LABEL into COMMON TRA.

Calling Sequence: CALL VECTOR (LABEL)

Subroutine VEQUAL

Purpose: To set one vector equal to another.

Calling Sequence: CALL VEQUAL (V,A)

Subroutine VLOAD

Purpose: To load a subscripted vector with three Cartesion values X, Y, and Z.

Calling Sequence: VLOAD (V,X,Y,Z)

Subroutine VPLOT

Purpose: To plot the locus of a radial vector orginating at the center of a planet, providing that the tip of the vector is visible from the viewpoint of the plot. The plot is in the form of a continuous line.

Calling Sequence: CALL VPLOT (RX, DUM1, DUM2, DUM3, DUM4, DUM5, DUM6, DUM7)

Subroutine VPPLT

Purpose: To plot an input symbol at the tip of a planetocentric vector, providing that the tip is visible from the viewpoint of the plot.

Calling Sequence: CALL VPPLT (RX, SYMBOL, DUM1, DUM2, DUM3, DUM4, DUM5, DUM6)

Subroutine VPROJ

Purpose: To establish the type of projection for Subroutines VPLOT and VPPLT

Calling Sequence: CALL VPROJ

Subroutine VPRINT

Purpose: To print a six character input word and a floating point vector number.

Calling Sequence: CALL VPRINT (V, LABEL)

Subroutine VROTAT

Purpose: To rotate a vector around a specified axis by a given angle.

Calling Sequence: Call VROTAT (V,A,B,S)

Subroutine VROTAX

Purpose: To rotate a vector around one of the Cartesian axes by a given angle.

Calling Sequence: Call VROTAX (V,A,K,S)

Subroutine VSCALE

Purpose: To multiply an input vector by a scalar.

Calling Sequence: CALL VSCALE (V,A,S)

Subroutine VSUM

Purpose: To determine the value of a vector, V, which is the sum of two other vectors, A and B.

Calling Sequence: CALL VSUM (V,A,B)

Subroutine VTRANS

Purpose: To provide a method of transferring a vector between two Cartesian systems.

Calling Sequence: CALL VTRANS (V,A,J,B,C)

Subroutine VUNIT

Purpose: To determine a unit output vector parallel to input vector.

Calling Sequence: CALL VUNIT (V,A)

Subroutine VVIEW

Purpose: To determine a vector from or to a planet or from a specified viewpoint.

Calling Sequence: CALL VVIEW (Q, NVW, VIEW, IVW, IP2)

Subroutine VZERO

Purpose: To set a vector equal to zero.

Calling Sequence: CALL VZERO (V)

Subroutine ARROCC

Purpose: To draw vectors with numeric labels to the requested bodies.

Calling Sequence: CALL ARROCC (ARROW, OCC, FR, NVW)

6.3 Major Subroutines Developed under Contract

6.3.1 CACL

Purpose:
1) Find vectors to planet from sun or Earth and spacecraft from planet.
2) Find range, cone and clock angles from spacecraft to satellite or planet.
3) Find longitude and latitude of picture center.
4) Find phase angle (sun to planet/satellite to spacecraft).
5) Find vector to picture center from spacecraft.
6) Find velocity vectors of spacecraft-planet motion and spacecraft rotational motion.
7) Find power *density* to TV lens and to TV sensor, energy of exposure and adjust exposure time if required.
8) Determine digital characteristics of exposure level.

Calling Sequence: CALL CACL

Input/Output:

I/O	FORTRAN Name	Definition
I	B	Brightness of planet surface
I	CCONE(200)	Cone angle user's input
I	CCLOK(200)	Clock angle, user's input
O	DCNCLK(8)	Total angle change to switch to satellite
O	DROTV(3)	Picture center velocity due to S/C rotation rates
O	DTARE(3)	Picture center velocity due to orbital velocities
I	EFB	Effective brightness of planet surface
O	EM	Emission angle
O	ENERGY	Energy of exposure (ergs/cm^2)
O	EPERDN	Digital number of calculated exposure
O	FRACT	Fraction of dynamic range for exposure
O	HANG2	Angle subtended by planet
O	IDN	Digital number (available shades of gray)
I	IMODE	Instrument mode = 0 instrument sighted down radius vector = 1 user's cone and clock angles
I	IP	Planet index number
O	IPCN	Picture number
I	IVS	Satellite index number, if specified
I/O	JDV	Julian date of interest
O	NDN	Exposure, expressed in a digital number
I	NREC	Index of user's cone and clock angle
I	NSC	Spacecraft type number, = -1, Sun oriented = 0 or 1 Earth oriented
I/O	P(26)	Instrument characteristics - exposure time (sec)
O	P(51)	Instrument characteristic - T/number for camera
O	PERCT	Percent of dynamic range equivalent of exposure
O	PHAS(8)	Phase angle for eight possible satellites
O	PHASE	Phase angle of planet (radians)
O	PHASED	Phase angle of planet (deg)
O	PHASEP	Phase angle of specified satellite (deg)
O	PHASES	Phase angle of specified satellite (radians)
I	PM(3)	Prime meridan vector at date of interest
O	POWER	Power density on sensor (ergs/cm^2/sec)
O	POWLN	Power density at lens (ergs/cm^2/sec)
I	PV(3)	Planet pole vector
O	R(3)	Range, cone & clock angles to planet from S/C
O	RADP(3)	Vector to picture center on planet or satellite
O	RCCL(3)	Unit vector to picture center from S/C
O	ROTV(3)	Velocity of image due to S/C rotation rates
I	ROTW(3)	Spacecraft rotation rates

I	RPL(12)	Radius of planet of interest (km)
O	RSAT(3)	Range, cone & clock angles to specified or closest satellite
O	SAF(8)	Half angle of eight possible satellites
I	SB(8)	Surface brightness of eight possible satellites
I	SEFB(8)	Effective surface brightness of eight possible satellites
O	SRAT(3,8)	Range, cone and clock to eight possible satellites from S/C
O	SSV(8)	Phase angle to eight possible satellites
O	SVSV(3,8)	Velocity of satellite picture center due to S/C
O	TARGE(3)	Velocity vector of picture center (km/sec)
O	TARGP(3)	Radius, latitude & longitude of picture center
I/O	TT	Time from periapsis (sec)
O	VSV(3,8)	Vector to satellite from S/C (camera coord)
O	VVSAT(3,8)	Velocity vector of eight satellites toward S/C (camera coord)
O	X(3)	Position vector Ø to S/C from planet/satellite
I/O	XE(3)	Heliocentric position vector of Earth
O	XN(3)	Vector to S/C from planet/satellite
I	XP(3)	Heliocentric position vector of planet from sun
O	XP(3)	Vector to planet from reference body (sun or Earth)
O	XSAT(3)	Vector to closest satellite
O	XSS(3)	Vector to S/C from satellite
I	XZ(97)	Input variable array
	XS(3)	Vector to reference body from S/C

Subprograms required: VDIF, VSCALE, RCNCLK, VUNIT, PGLOBE, VCROSS, UCROSS, VCOMB, VEQUAL, VLOAD, SPHERE, VTRANS, ORBVEL, ANGV, PLASAT, VROTAT, PSCONS

Approximate Storage Required (octal): 1760

Discussion: None

Table 6-1 presents the CACL listing; Fig. 6-1 is the CACL flow diagram.

Table 6-1 CACL Listing

```
      SUBROUTINE CACL
C
C                                                                   CACL-2
C
      COMMON/JUNK/B,EFB,SB(8),SEFB(8)
      COMMON/POBEAR/SSV(8),SAF(8),HANG2,JCNCLK(8)
      COMMON/HEDING/TITLE(13)
      COMMON/TNME/TVNAME(5)
      DIMENSION ZSAT(3),VORB(3)
      DATA AU/1.49599E08/
      DATA WTERG/1.E+7/
      DATA NCHGT/0/,NCHGF/0/
      COMMON/TEST/ITEST(97)
      COMMON/XZ/XZ(97)
      COMMON/CHARAC/P(60)
      COMMON/FLYBY/DUM(6),XN(3),DUMM(9)
      COMMON/CELEST/C(3),S(3),SN(3)
      COMMON/GEOM/XE(3),XP(3)
      COMMON/PHASER/PHAS(8),SRAT(3,8)
      COMMON/STUFF/X(3),TARGE(3),ROTV(3),VSV(3,8),VVSAT(3,8),SVSV(3,8)
     X,RCCL(3),DTARE(3),DROTV(3)
      COMMON/STUF1/XSAT(3)
      COMMON /PLCONS/GMS,GMPL(12),RS,RPL(12),SNAME(2),PLNAME(2,12),
     X SPV(3),PLPV(3,12),SVE(3),PLVE(3,12),SROT,PLROT(12),SOBL,PLOBL(12)
      COMMON/PCOORD/PV(3),VE(3),PM(3)
      COMMON/VCONS/UX(3),UY(3),UZ(3),PI
      COMMON/REVER/ROTW(3)
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      DIMENSION       U(3),XSN(3),UXSS(3),TARG(3),XTER(3)
     X,                         XSAT1(3),RADP(3),VV(3)
     X,RAD1(3),C2(3),C1(3),XS(3),XSS(3),VSAT(3)
     X,TARGV(3)            ,UXP(3),ROTC(3)
      COMMON/FOROUT/POWER,ENERGY,PERCT,IJN,R(3),RSAT(3),TARGP(3),IPCN
     X,FRACT,PHASED,PHASEP  ,POWLN,NDN,EPERDN
      COMMON/FLAGS/IMODE,IFLPL,IFLRP
      COMMON/CAM/CCONE(200),CCLOCK(200)
      REAL JDV
      COMMON/JDAYS/JDV,TT
       DIMENSION NS(9)
       DATA NSO/9/,NS/0,0,1,2,4,7,5,2,0/
      T=TT/3600.
      TTT=TT/86400.
      XPM=(ABSV(XP)/AU)**2
      TS1=XZ(87)
      TS2=XZ(88)
      ISAVE=0
      NCB=0
      IDN=2.**P(14)+0.01
      PI=3.1415926536
C
C
C
C
```

Table 6-1 (cont)

```
      CALL VUNIT(UXP,XP)
      IF(NSC.LT.0) GO TO 6
      CALL VDIF (XP,XP,XE)
    6 CALL VSCALE(XS,XP,-1.)
      CALL VDIF (XS,XS,X)
      CALL VSCALE(R,X,-1.)
      CALL RCNCLK(R,XS)
      CALL VUNIT(U ,X)
      CALL VSCALE(UXP,UXP,-1.)
      PHASE=ACOS(DOT(UXP,U ))
      CALL PGLOBE(PM,JDV,IP)
      CALL VSCALE(TARG,U , RPL(IP))
      CALL VDIF(RADP,TARG,X)
      HANG=ASIN(RPL(IP)/R(1))*180./PI
      HANG2=2.*HANG
C
C     IMODE NE 0 MEANS CONE/CLOCK IS INPJT
C
      IF(IMODE.NE.0) GO TO 10
      IF(PHASE.LT.PI/2.) GO TO 20
C     SUBVEHICLE POINT IN SHADOW
      CALL VCROSS(XSN,UXP,U)
      CALL UCROSS(XTER,XSN,UXP)
      ANGL=90.-HANG
      ANGL=PHASE*180./PI-ANGL
      DEL=(90.-ANGL)/2.
      ANGL=ANGL+DEL
C     PICTURE CENTER BISECTS ANGLE BETWEEN LIT LIMB + TERMINATOR
      CALL VCOMB(TARG,UXP,COS(ANGL*PI/180.),XTER,SIN(ANGL*PI/180.))
      CALL VSCALE(TARG,TARG,RPL(IP))
C     RADP IS VECTOR FROM S/C TO PIC CENTER
      CALL VDIF(RADP,TARG,X)
      CALL VEQUAL(RAD1,RADP)
      CALL RCNCLK(RAD1,XS)
      GO TO 20
C
C
C
   10 CALL VLOAD(RCCL,1.,CCONE(NREC),CCLOCK(NREC))
      CALL SPHERE(RCCL,RCCL,4HFROM,5HPOLAR,6HDEGREE)
      CALL VTRANS(RADP,RCCL,4HFROM,C,XS)
      CALL VSCALE(U,U,-1.)
      CA=DOT(U,RADP)
      SROOT=RPL(IP)**2-R(1)**2*(1.-CA*CA)
C     IF SROOT LE 0 CENTER OF FRAME IS OJF PLANET
      IF(SROOT.LT.0.) SROOT=0.
      TARGM=R(1)*CA-SQRT(SROOT)
      CALL VCOMB(TARG,RADP,TARGM,U,-R(1))
      CALL VSCALE(RADP,RADP,TARGM)
```

Table 6-1 (cont)

```
   20   CONTINUE
        CALL VTRANS(TARGP,TARG,2HTO,PM,PV)
        CALL VCROSS(TARGV,UZ,TARGP)
        CALL SPHERE(TARGP,TARGP,2HTO,6HLATLON,6HDEGREE)
        CALL VSCALE(TARGV,TARGV,PLROT(IP)/86400.*2.*PI)
        CALL VTRANS(TARGE,TARGV,4HFROM,PM,PV)
        CALL ORBVEL(VV,TT)
        CALL VDIF(VV,TARGE,VV)
        EM=180.-ANGV(RADP,TARG,6HDEGREE)
C
C
C
        IF(IVS.EQ.0) GO TO 8
C       XSS IS VECTOR FROM SATELLITE TO S/C
        CALL VDIF(XSS,X,XSAT)
        CALL VUNIT(UXSS,XSS)
C       ASSUNES SUN VECTOR IS SMME AS PLANETS
        PHASES=ACOS(DOT(UXP,UXSS))
        CALL VSCALE(RSAT,XSS,-1.)
        CALL RCNCLK(RSAT,XS)
C       CHECK TO SEE IF PLANET OBSCURES SATELLITE
        IF((ANGV(X,XSS,6HDEGREE)).LT.HANG.AND.RSAT(1).GT.R(1))
       XPRINT 999,TTT
    8   CONTINUE
C
C
C
        CALL ORBVEL(VORB,TT)
        NSP=NS(IP)
        IF(NSP.EQ.0) GO TO 106
        ZAP=R(3)
        IF(ZAP.GT.180.) ZAP=ZAP-360.
        DO 500 I=1,NSP
        IS=I
        IF(IP.EQ.6.AND.IS.EQ.1) GO TO 500
        CALL PLASAT(ZSAT,JDV,IS,IP)
        CALL VDIF(VSV(1,IS),X,ZSAT)
        SSV(I)=ANGV(VSV(1,I),UXP,6HDEGREE)
        CALL PSCONS(RPSAT,6HRADIUS,I,IP)
        RPSC=ABSV(VSV(1,I))
  600   SAF(I)=ASIN(RPSAT/RPSC)*(360./PI)
        CALL VUNIT(UXSS,VSV(1,IS))
        PHAS(I)=ACOS(DOT(UXP,UXSS))
        CALL VSCALE(SRAT(1,IS),VSV(1,IS),-1.)
        CALL RCNCLK(SRAT(1,IS),XS)
        SAP=SRAT(3,IS)
        IF(SAP.GT.180.) SAP=SAP-360.
        DCNCLK(I)=ABS(SRAT(2,IS)-R(2)) + ABS(SAP-ZAP)
        CALL PLASAT(XSAT1,JDV+1./8640.,IS,IP)
        CALL VDIF(XSAT1,XSAT1,ZSAT)
        CALL VSCALE(XSAT1,XSAT1,.1)
        CALL VDIF(VVSAT(1,IS),VORB,XSAT1)
        CALL VSCALE(VSV(1,IS),VSV(1,IS),-1.)
        C'   UCROSS(C2,XS,VSV(1,IS))
```

Table 6-1 (cont)

```
      CALL UCROSS(C1,C2,VSV(1,IS))
      CALL VTRANS(VVSAT(1,IS),VVSAT(1,IS),2HTO,C1,VSV(1,IS))
      CALL VTRANS(ROTC,ROTW,4HFROM,C,XS)
      CALL VTRANS(ROTC,ROTC,2HTO,C1,VSV(1,IS))
      CALL VTRANS(VSV(1,IS),VSV(1,IS),2HTO,C1,VSV(1,IS))
      CALL VCROSS(SVSV(1,IS),ROTC,VSV(1,IS))
      IF(IP.EQ.6.AND.I.EQ.1) GO TO 700
      IF(ISAVE.NE.0) GO TO 650
      RMIN=SRAT(1,I)
      ISAVE=I
650   IF(SRAT(1,I).GT.RMIN) GO TO 700
      RMIN=SRAT(1,I)
      ISAVE=I
700   CONTINUE
500   CONTINUE
      IF(XZ(3).NE.0) GO TO 106
      CALL VEQUAL(RSAT,SRAT(1,ISAVE))
      IF(RSAT(1).GT.XZ(86).OR.T.GT.TS1.OR.T.LT.TS2) GO TO 105
      CALL VUNIT(UXSS,VSV(1,ISAVE))
      CALL VEQUAL(XSAT,VSV(1,ISAVE))
C
C
C
106   IF(IFLST.EQ.1) GO TO 99
C     RADP IS VECTOR FROM S/C TO PIC CENTER
      IF(ITEST(86).EQ.1.AND.RSAT(1).LT.XZ(86)) 99,105
99    CALL PSCONS(RADS,6HRADIUS,IVS,IP)
      NCB=1
      CALL VCOMB(RADP,XSS,-1.,UXSS,RADS )
      CALL VSUM(TARG,XSS,RADP)
      CALL VTRANS(TARGP,TARG,2HTO,PM,PV)
      CALL SPHERE(TARGP,TARGP,2HTO,6HLAT.ON,6HDEGREE)
      CALL PLASAT(XSAT1,JDV+1./8640.,IVS,IP)
      CALL VDIF(VSAT,XSAT1,XSAT)
      CALL VSCALE(VSAT,VSAT,.1)
      CALL ORBVEL(VV,TT)
      CALL VDIF(VV,VV,VSAT)
      CALL VEQUAL(XN,XSS)
      IF(IVS.NE.0) ISAVE=IVS
      SSB=SB(ISAVE)/XPM
      SBBEF=SEFB(ISAVE)/XPM
      POWER=SBBEF/(4.*P(51)**2)*PZ*WTERG
      POWLN=SBB*PZ*WTERG
      PZ=(SIN(PHASES)+(PI-PHASES*COS(PHASES))/PI)
105   CONTINUE
C
C
C
C
      IF(NCB.EQ.1) GO TO 101
      BB=B/XPM
      BBEF=EFB/XPM
```

Table 6-1 (cont)

```
C       LAMBERT PLANET PHASE FUNCTION
      PZ=(SIN(PHASE)+(PI-PHASE*COS(PHASE))/PI*ABS(COS(EM*PI/180.)))
      POWLN= BB*PZ*WTERG
      POWER= BBEF/(4.*P(51)**2)*PZ*WTERG
101   CONTINUE
      ENERGY=POWER*P(26)
      IF(ENERGY-0.95*P(16))2,2,1
    1 DTMIN=(.85*(P(16)-P(15))+P(15))/POWER
      P(26)=DTMIN
      WRITE(6,1005)TITLE,TVNAME
      WRITE(6,1000)ENERGY,DTMIN
      ENERGY=0.85*(P(16)-P(15))+P(15)
      WRITE(6,1001)ENERGY
      NCHGT=NCHGT+1
      WRITE(6,1004)NCHGT
      IF(DTMIN-1.E-4)3,2,2
    3 DTMIN=1.E-4
      P(26)=DTMIN
      POWXX=POWER*(P(51)**2)/(P(52)**2)
      FILTF=DTMIN*POWXX/ENERGY
      P(51)=P(52)*SQRT(FILTF)
      WRITE(6,1002)FILTF
      NCHGF=NCHGF+1
      WRITE(6,1004)NCHGF
    2 IF(ENERGY-P(15))4,5,5
    4 IF(POWER.LT.1.E-8)GOTO11
      DTMAX=P(15)/POWER
      ENERGY=P(15)
      P(26)=DTMAX
      WRITE(6,1003)DTMAX
      GO TO 5
   11 WRITE(6,1006)POWER
    5 CONTINUE
      FRACT=0.0
      IF(ENERGY.GT.P(15).AND.ENERGY.LT.P(16))FRACT=(ENERGY-P(15))/
     $ (P(16)-P(15))
      XDN=IDN
      NDN=FRACT*XDN + 0.001
      EPERDN=(P(16)-P(15))/XDN
      PERCT=FRACT*100.
C
C
C
      CALL UCROSS(C2, XS,RADP)
      CALL UCROSS(C1,C2,RADP)
      CALL VROTAT(C1,C1,RADP,TWIST*PI/180.)
      CALL VTRANS(TARGE,VV,2HTO,C1,RADP)
      CALL VTRANS(RCCL,RADP,2HTO,C1,RADP)
      CALL VTRANS(ROTC,ROTW,4HFROM,C,XS)
      CALL VTRANS(ROTC,ROTC,2HTO,C1,RADP)
```

Table 6-1 (concl)

```
      CALL VCROSS(ROTV,ROTC,RCCL)
      SMEAR=ABSV(TARGE)
      SCMER=ABSV(ROTV)
      PHASEP=PHASES*180./PI
      PHASED=PHASE*180./PI
      WRITE(7) TT,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT,PHASES,XSS,
     XJDV,EM
      IF(NCB.EQ.1) CALL VEQUAL(X,XSS)
C
C
C
      CALL VEQUAL(DTARE,TARGE)
      IF(NCB.EQ.1) CALL VEQUAL(DTARE,VVSAT(1,IVS))
      CALL VEQUAL(DROTV,ROTV)
      IF(NCB.EQ.1) CALL VEQUAL(DROTV,SVSV(1,IVS))
      RETURN
901   FORMAT(5X,11HPLANET DATA,/)
904   FORMAT(5X,14HPICTURE LAT = ,F6.2,5X,6HLON = ,F6.2,8HSMEAR = ,
     XE16.8,5X,12HSMEAR S/C = ,F16.8)
902   FORMAT(5X,14HSATELLITE DATA,/)
906   FORMAT(5X,8HRANGE = ,E16.8,5X,7HCONE = ,F6.2,5X,8HCLOCK = ,F6.2,5X
     X,8HPHASE = ,F6.2,/)
 999  FORMAT(28HSATELLITE OBSCURED BY PLANET,F6.2,19HDAYS FROM ENCOUNTE
     XR)
 1000 FORMAT(1X ,19HEXPOSURE TOO HIGH  ,E10.3,13H (ERGS/CM**2),/
     $ 26H EXPOSURE TIME REDUCED TO ,E10.3,4H SEC)
 1003 FORMAT(47H EXPOSURE LESS THAN MINIMUM  , EXP TIME SET TO ,E10.3,/)
 1001 FORMAT(13H EXPOSURE IS ,E10.3,12H (ERG/CM**2),)
 1002 FORMAT(34H EXPOSURE TIME SET TO 0.1 MILLISEC,/
     $27H REQUIRED FILTER FACTOR IS ,E10.3)
 1004 FORMAT(5X,3HCHG,I5)
 1005 FORMAT(1H1,13A6,/,5A6/)
 1006 FORMAT(/,35H POWER DENSITY THRU LENS TOO LOW = ,E15.8,14H ERGS/SQC
     *M/SEC    ,/)
      END
```

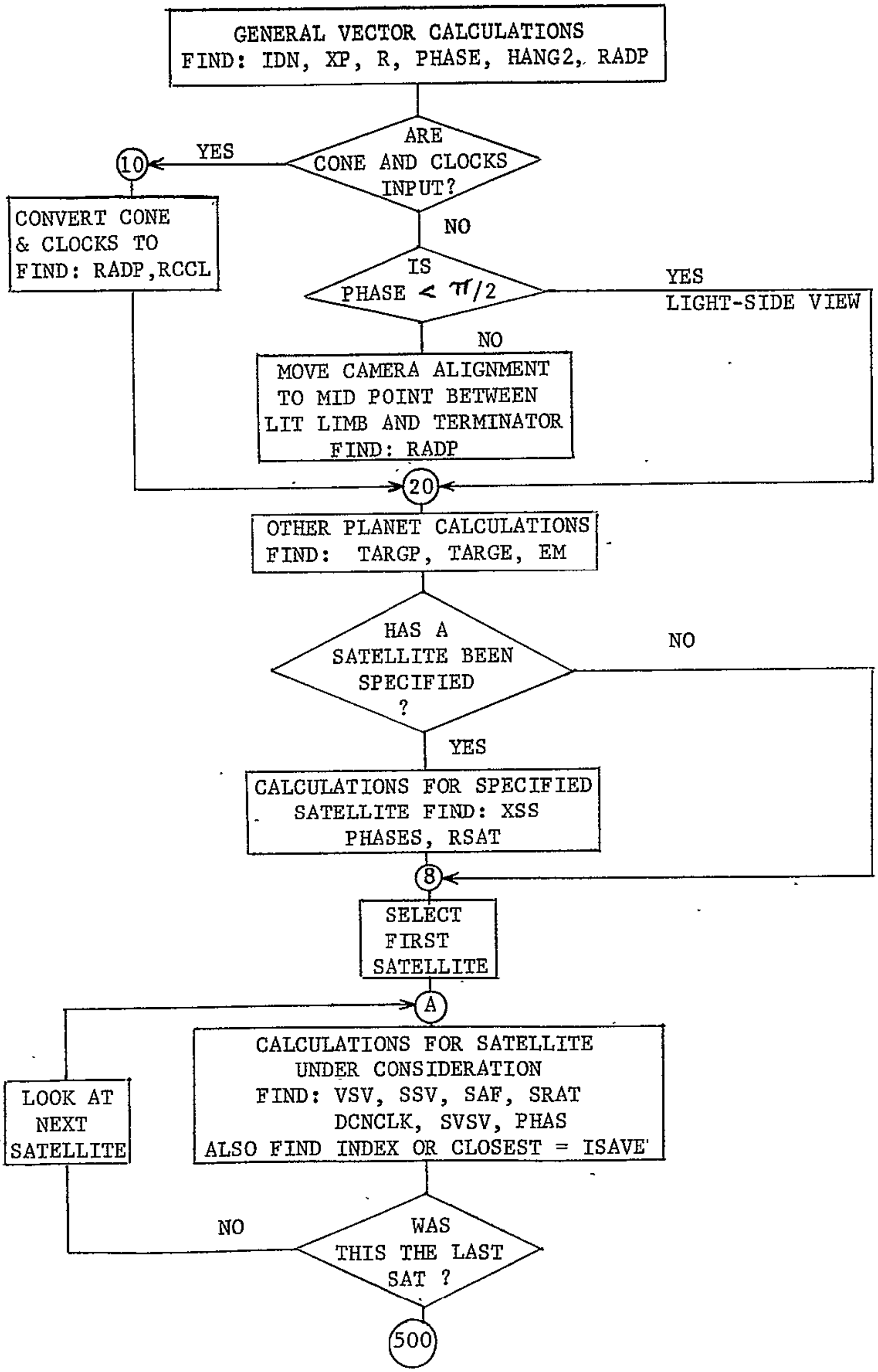


Fig. 6-1 CACL Flow Diagram

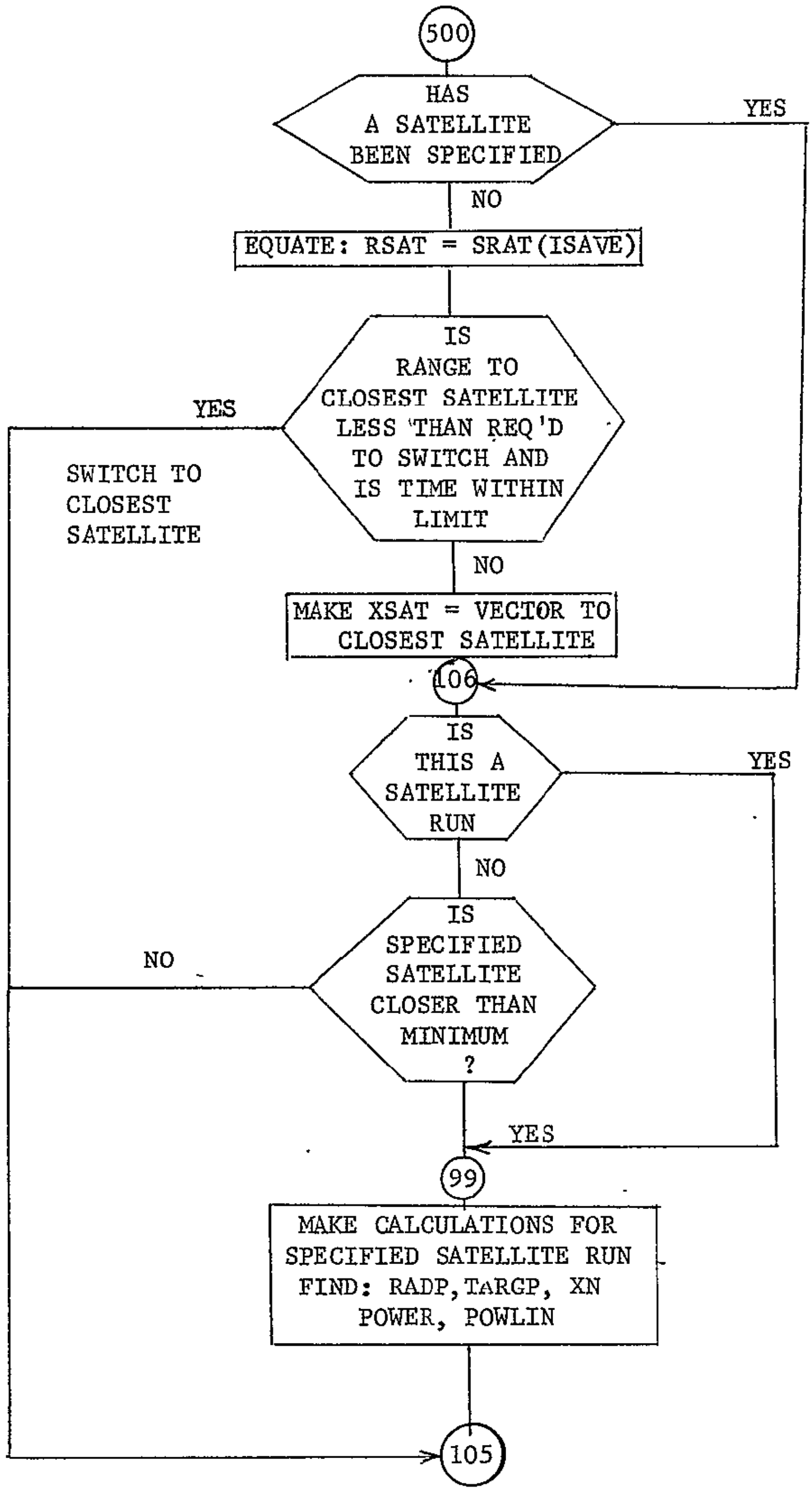

500
HAS A SATELLITE BEEN SPECIFIED
YES
NO
EQUATE: RSAT = SRAT(ISAVE)
IS RANGE TO CLOSEST SATELLITE LESS THAN REQ'D TO SWITCH AND IS TIME WITHIN LIMIT
YES
SWITCH TO CLOSEST SATELLITE
NO
MAKE XSAT = VECTOR TO CLOSEST SATELLITE
106
IS THIS A SATELLITE RUN
YES
NO
IS SPECIFIED SATELLITE CLOSER THAN MINIMUM ?
NO
YES
99
MAKE CALCULATIONS FOR SPECIFIED SATELLITE RUN FIND: RADP, TARGP, XN POWER, POWLIN
105

Fig. 6-1 (cont)

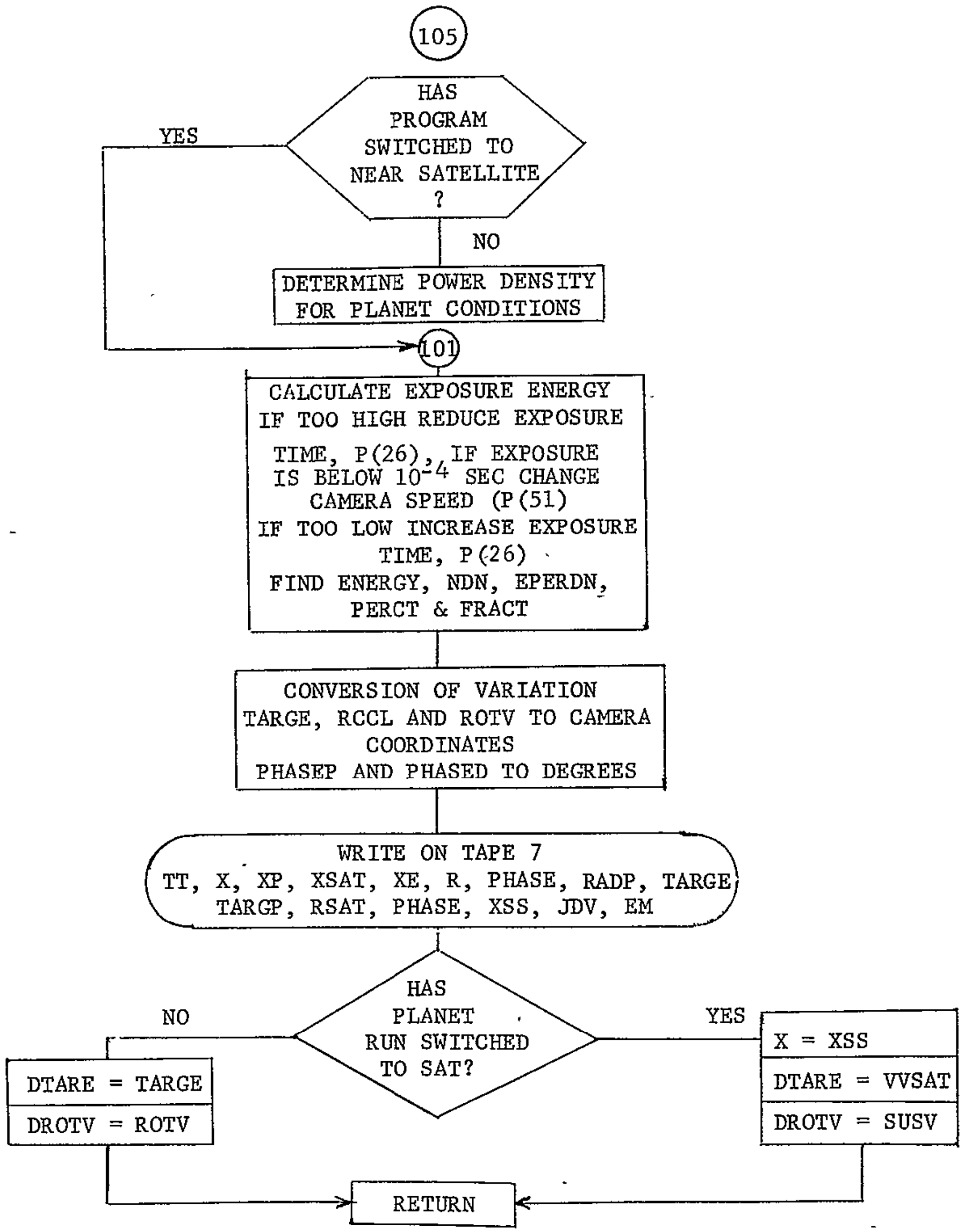

105
HAS PROGRAM SWITCHED TO NEAR SATELLITE ?
YES
NO
DETERMINE POWER DENSITY FOR PLANET CONDITIONS
101
CALCULATE EXPOSURE ENERGY IF TOO HIGH REDUCE EXPOSURE TIME, P(26), IF EXPOSURE IS BELOW 10-4 SEC CHANGE CAMERA SPEED (P(51) IF TOO LOW INCREASE EXPOSURE TIME, P(26)
FIND ENERGY, NDN, EPERDN, PERCT & FRACT
CONVERSION OF VARIATION TARGE, RCCL AND ROTV TO CAMERA COORDINATES
PHASEP AND PHASED TO DEGREES
WRITE ON TAPE 7
TT, X, XP, XSAT, XE, R, PHASE, RADP, TARGE
TARGP, RSAT, PHASE, XSS, JDV, EM
HAS PLANET RUN SWITCHED TO SAT?
NO
YES
DTARE = TARGE
DROTV = ROTV
X = XSS
DTARE = VVSAT
DROTV = SUSV
RETURN

Fig. 6-1 (concl)

6.3.2 FIELDS

Purpose: To construct plots that illustrate the magnetic field of the planet, and the interaction of this field with the solar wind.

Calling Sequence: CALL FIELDS

Input/Output:

I/O	FORTRAN Name	Definition
I	XLAT	Latitude of magnetic pole of planet
I	XLON	Longitude of magnetic pole of planet
I	RPL(12)	Radius of planet (km)
I	PV(3)	Pole vector of planet
I	IP	Planet index number
I	AA	Semimajor axis of spacecraft trajectory
I	E	Eccentricity of S/C orbit
I	P	Semi-latus rectum of S/C orbit
I	JDV	Julian date of encounter
I	DPAUSE	Distance to magentopause
O	TX	Time from encounter (days)
O	BMINS	Minimum detectable planet magnetic dipole at S/C
O	DIRB(3)	Direction vector of line tangent to magnetic line of force

Subprograms Required: MAP, ORBPOS, UCROSS, VPROJ, VVIEW, DISK, VCOMB, VPLOT, ANOMLY, SINTRP, VPPLT, PLPOS, VSCALE, PGLOBE, RNCLK, LINE, SPHERE, VTRANS, VSCALE,

Approximate Storage Required (octal): 2026

Discussion: None

Table 6-2 presents the FIELDS listing; Fig. 6-2 is the FIELDS flow diagram.

Table 6-2 .FIELDS Listing

```
      SUBROUTINE FIELDS
      DIMENSION DIRB(3),REF(3),XMA(3),YMA(3),ZMA(3),XE(3),XP(3)
      COMMON/CHARAC/PF(60)
      COMMON /PLTBUF/BUF(20)
      COMMON /PLCONS/GMS,GMPL(12),RS,RPL(12),SNAME(2),PLNAME(2,12),
     X SPV(3),PLPV(3,12),SVE(3),PLVE(3,12),SROT,PLROT(12),SOBL,PLOBL(12)
      COMMON /CVPLOT/VT(3),VR(3),VW(3),RPP,TANR,R2,SCALE,R(3),PEN
       COMMON/PCOORD/PV(3),VE(3),PM(3)
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      COMMON/ORBI/AA,E,P,ORB,C1,TA,TEJ,VV(3),QV(3)
      REAL JD,JDV
       COMMON/JDAYS/JDV,TT
      COMMON/PLAN/GM
       DIMENSION XM(3),X(3),XF(3),V(3),XS(3) ,SN(3)
      PI=3.14159
       RTD=57.295780
      CALL ARCSTP(5.)
      SENS=PF(10)
      XLAT=PF(8)
      XLON=P(9)
      XLON=PF(9)
      DPAUSE=PF(6)*RPL(IP)
      DINC=PF(7)*RPL(IP)
      A=AA
      ECC=E
      AE=ECC
        DPAUSE=64.*RPL(IP)
        DINC=0.275*DPAUSE
        DELANG=0.
        DELTR=DELANG/RTD
        R2=0.
        CALL ORBPOS(X,0.)
       SCALE=ABSV(X)*5.
       IF(SCALE.LT.11.*RPL(IP)) SCALE=11.*RPL(IP)
       SCALE=SCALE+RPL(IP)*1.1
       BUF(4)=-(SCALE-RPL(IP)*1.1)
       BUF(5)=RPL(IP)*1.1
       BUF(6)=-SCALE/2.
       BUF(7)=SCALE/2.
       BUF(8)=0.0
       BUF(9)=1.0
       BUF(10)=0.0
       BUF(11)=1.0
       CALLMAP(BUF(4),BUF(5),BUF(6),BUF(7),BUF(8),BUF(9),BUF(10),BUF(11))
        RPP=RPL(IP)
        CALL UCROSS(VW,PV,X)
        CALL VPROJ
        CALL VVIEW(1,11,1,15,0)
        DMDT=SQRT(GM/ABS(A)**3)
C       PLOT PLANET DISK
        CALL DISK(IP,0,0,0,0)
        IJK=0
        X(3)=0.
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Table 6-2 (cont)

```
        PLOT THE MAGNETIC FIELD
      NMAX=SCALE/RPL(IP)
      DO 101 I=2,NMAX
       RO= I*RPL(IP)
       DO 100 J=90,270,2
       ALAT=(J-5)/RTD
       ANG=ALAT-DELTR
       RADIUS=RO*COS(ANG)**2
       IF(RPL(IP) .GT. RADIUS) GO TO 50
C      X IS THE VECTOR TO THE SPACECRAFT
       CALL VCOMB(X,VR,RADIUS*SIN(ALAT),VT,RADIUS*COS(ALAT))
       IF(IJK .EQ. 1)  GO TO 75
       IF(J .EQ. 5)  GO TO 75
C      PLOT MAGNETIC LINES OF FLUX
       CALL VPLOT(X)
       GO TO 100
 50    IJK=1
C      LIFT THE PEN
       CALL SINTRP(BUF)
       GO TO 100
 75    IJK=0
 100   CONTINUE
       CALL SINTRP(BUF)
C      LIFT THE PEN
 101   CONTINUE
 250   R=SCALE
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      COMPUTE TRAJECTORY PARAMETERS
       CAT=(P/R-1.)/ECC
       AT=ACOS(CAT)
       T=ANOMLY(-AE,AT)/DMDT
      N=ABS(T)/3600.
      NN=-N
    * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
       COMPUTE MAGNETIC POLE VECTOR (V)
       CALL VLOAD(V,1.,XLAT,XLON)
       CALL SPHERE(V,V,4HFROM,6HLATLON,6HDEGREE)
C      COMPUTE AND PLOT S/C TRAJECTORY
      DO 300 I=NN,N
      T=I*3600.
      CALL ORBPOS(XS,T)
       JD=JDV+ I*3600./86400.
       CALL PGLOBE(PM,JD,IP)
      CALL VTRANS(ZMA,V,4HFROM,PM,PV)
      DEC= ANGV(ZMA,XS,6HRADIAN)
       RADIUS=ABSV(XS)
      TMP=90./RTD+DEC
       CALL VCOMB(X,VR,RADIUS*SIN(TMP),VT,RADIUS*COS(TMP))
       CALL VPPLT(X,1H*)
  300 CONTINUE
      CALL SINTRP(BUF)
 310   CALL FRAME
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
```

Table 6-2 (cont)

```
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      SCALE=DINC*1.1
       BUF(4)=-SCALE
       BUF(5)=+SCALE
       BUF(6)=-SCALE
      BUF(7)=+SCALE
      BUF(8)=0.0
      BUF(9)=1.0
      BUF(10)=0.0
      BUF(11)=1.0
       CALL PLPOS(XS,JDV,IP)
       CALL VSCALE(XS,XS,-1.)
C      XS IS THE VECTOR POINTING TO THE SUN
      CALL MAP(-SCALE,SCALE,-SCALE,SCALE,0.0,1.0,0.0,1.0)
C      PLOT PLANET DISK
      CALL LINE(-SCALE,0.,.9*SCALE,0.)
      CALL SYMBOL(3HS$.)
       CALL DISK(IP,0,0,0,0)
      N=30
       NN=-N
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C      PLOT MAGNETOPAUSE CONTOUR
       DO 400 J=NN,N
       ALAT=J/RTD
       RPAUSE=DPAUSE*COS(ALAT)**2
       CALL VCOMB(X,VR,RPAUSE*SIN(ALAT),VT,RPAUSE*COS(ALAT))
       CALL VPLOT(X)
  400  CONTINUE
C      LIFT THE PEN
       CALL SINTRP(BUF)
      RPAUSE=DPAUSE*COS(PI/6.)**2
      Y=RPAUSE*SIN(PI/6.)
      XX=RPAUSE*COS(PI/6.)
      CALL VCOMB(X,VR,Y,VT,XX)
      CALL VPLOT(X)
      CALL VCOMB(X,VR,Y,VT,-SCALE)
      CALL VPLOT(X)
      CALL SINTRP(BUF)
      CALL VCOMB(X,VR,-Y,VT,XX)
      CALL VPLOT(X)
      CALL VCOMB(X,VR,-Y,VT,-SCALE)
      CALL VPLOT(X)
      CALL SINTRP(BUF)
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
      N=90
      NN=-N
C      PLOT BOW SHOCK CONTOUR
      PP=2.*DINC
       DO 500 J=NN,N
       ALAT=J/RTD
       RBOWS=PP/(1.+COS(ALAT))
       CALL VCOMB(X,VR,RBOWS *SIN(ALAT),VT,RBOWS *COS(ALAT))
       CALL VPLOT(X)
 500   CONTINUF
```

Table 6-2 (cont)

```
       CALL SINTRP(BUF)
C      LIFT THE PEN
       R=SCALE
       CAT=(P/R-1.)/ECC
       AT=ACOS(CAT)
       T=ANOMLY(-AE,AT)/DMDT
      N=ABS(T)/3600.
      NN=-N
C      PLOT TRAJECTORY
C   * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
       DO 600 I=NN,N
      T=I*3600.
C      X IS THE VECTOR TO THE SPACECRAFT
      CALL ORBPOS(X ,T)
       JD=JDV+ I*3600./86400.
       CALL PLPOS(XS,JD,IP)
       CALL VSCALE(XS,XS,-1.)
       CALL ORBPOS(X,T)
       PHASE=ANGV(XS,X,6HRADIAN)
       XM=ABSV(X)
       CALL VCOMB(XF,VR,XM*SIN(PHASE),VT,XM*COS(PHASE))
       CALL VPPLT(XF,1H*)
 600   CONTINUE
C      LIFT THE PEN
       CALL SINTRP(BUF)
       CALL FRAME
      N=3.*24.
      NN=-N
      DO 612 I=NN,N
      T=I*3600.
      CALL ORBPOS(XS,T)
      TX=T/86400.
      IF(I.EQ.NN)TSTART=TX
      IF(I.EQ.N)TSTOP=TX
      JD=JDV+TX
      CALL PGLOBE(PM,JD,IP)
      CALL VTRANS(ZMA,V,4HFROM,PM,PV)
      CALL PLPOS(XE,JD,3)
      CALL PLPOS(XP,JD,IP)
      CALL VDIF(REF,XE,XP)
      RM=ABSV(XS)
      ANGLE=ANGV(ZMA,XS,6HRADIAN)
      ANGLE=PI/2. -ANGLE
      BMINS=(SENS/((RPL(IP)/RM)**3*SQRT(1.+3.*SIN(ANGLE)**2)/COS(ANGLE)*
     X*6))
      IF(I.EQ.NN)XMIN=BMINS
      XMIN=AMIN1(XMIN,BMINS)
      XMAX=AMAX1(XMAX,BMINS)
      CALL UCROSS(YMA,ZMA,XS)
      CALL  UCROSS(XMA,YMA,ZMA)
      CALL VTRANS(XS,XS,2HTO,XMA,ZMA)
```

Table 6-2 (cont)

```
      XX=XS(1)
      YY=XS(3)
      CALL SPHERE(XS,XS,2HTO,LATLON,6HRADIAN)
      RIB=XS(1)/COS(XS(2))**2
      IF(ABS(YY).LT.1.E-6) GO TO 610
      SLOPE=(2.*XX*RIB-3.*XX*SQRT(XX**2 + YY**2))
     X/(3.*YY*SQRT(XX**2 + YY**2))
      Y1=YY-SLOPE*XX
      IF(YY.LT.0.) CALL VLOAD(DIRB,XX,0.,(YY-Y1))
      IF(YY.GT..0) CALL VLOAD(DIRB,-XX,0.,(Y1-YY))
      GO  TO 611
610   CALL VSCALE(DIRB,ZMA,-1.)
611   CALL VUNIT(DIRB,DIRB)
      CALL VSCALE(DIRB,DIRB,-1.)
      CALL VTRANS(DIRB,DIRB,4HFRON,XMA,ZMA)
      CALL RCNCLK(DIRB,REF)
 612  WRITE (8)TX,BMINS,DIRB
      CALL MAPSSL(TSTART,TSTOP,XMIN,XMAX,.1,1.,.1,1.)
      CALL CHAROPT(0,0,0,1,0)
      CALL ABSBEAM(.05,.4)
      CALL SYMBOL(22HMINIMUM DETECTABLE B$.)
      CALL SYMBOL(10H (GAUSS)$.)
      CALL ABSBEAM(.4,.05)
      CALL CHAROPT(0,0,0,0,0)
      CALL SYMBOL(29H TIME FROM ENCOUNTER (DAYS)$.)
      REWIND 8
      READ(8)TX,BMINS
502   READ(8)TX2,BMINS2
      IF(EOF,8) 501,503
503   CALL LINE(TX,BMINS,TX2,BMINS2)
      TX=TX2
      BMINS=BMINS2
      GO TO 502
 501  CONTINUE
      CALL FRAME
      REWIND 8
      CALL MAPS(TSTART,TSTOP,-180.,180.,.1,1.,.1,1.)
      CALL CHAROPT(0,0,0,1,0)
      CALL ABSBEAM(.05,.2)
      CALL SYMBOL(37HCLOCK ANGLE OF MAGNETIC FLUX VECTOR$.)
      CALL CHAROPT(0,0,0,0,0)
      CALL ABSBEAM(.2,.05)
      CALL SYMBOL(28HTIME FROM ENCOUNTER (DAYS)$.)
      READ (8) TX,BMIN,DIRB
      IF(EOF,8)622,621
621   CLOCK=DIRB(3)
      IF(CLOCK.GT.180.0) CLOCK=CLOCK-360.0
      T=TX
624   READ (8) TX,BMIN,DIRB
      IF(EOF,8) 622,623
623   ZAP=DIRB(3)
      IF(ZAP.GT.180.) ZAP=ZAP-360.
      CALL LINE(T,CLOCK,TX,ZAP)
```

Table 6-2 (concl)

```
      T=TX
      CLOCK=ZAP
      GO TO 624
622   REWIND 8
      CALL FRAME
      CALL MAPS(TSTART,TSTOP,0.,180.,.1,1.,.1,1.)
      CALL CHAROPT(0,0,0,1,0)
      CALL ABSBEAM(.05,.2)
      CALL SYMBOL(36HCONE ANGLE OF MAGNETIC FLUX VECTOR$.)
      CALL CHAROPT(0,0,0,0,0)
      CALL ABSBEAM(.2,.05)
      CALL SYMBOL(28HTIME FROM ENCOUNTER (DAYS)$.)
      READ (8) TX,BMIN,DIRB
      IF(EOF,8) 626,627
627   CONE=DIRB(2)
      T=TX
628   READ (8) TX,BMIN,DIRB
      IF(EOF,8) 626,629
629   CALL LINE(T,CONE,TX,DIRB(2))
      CONE=DIRB(2)
      T=TX
      GO TO 628
626    CONTINUE
      CALL FRAME
       RETURN
       END
```

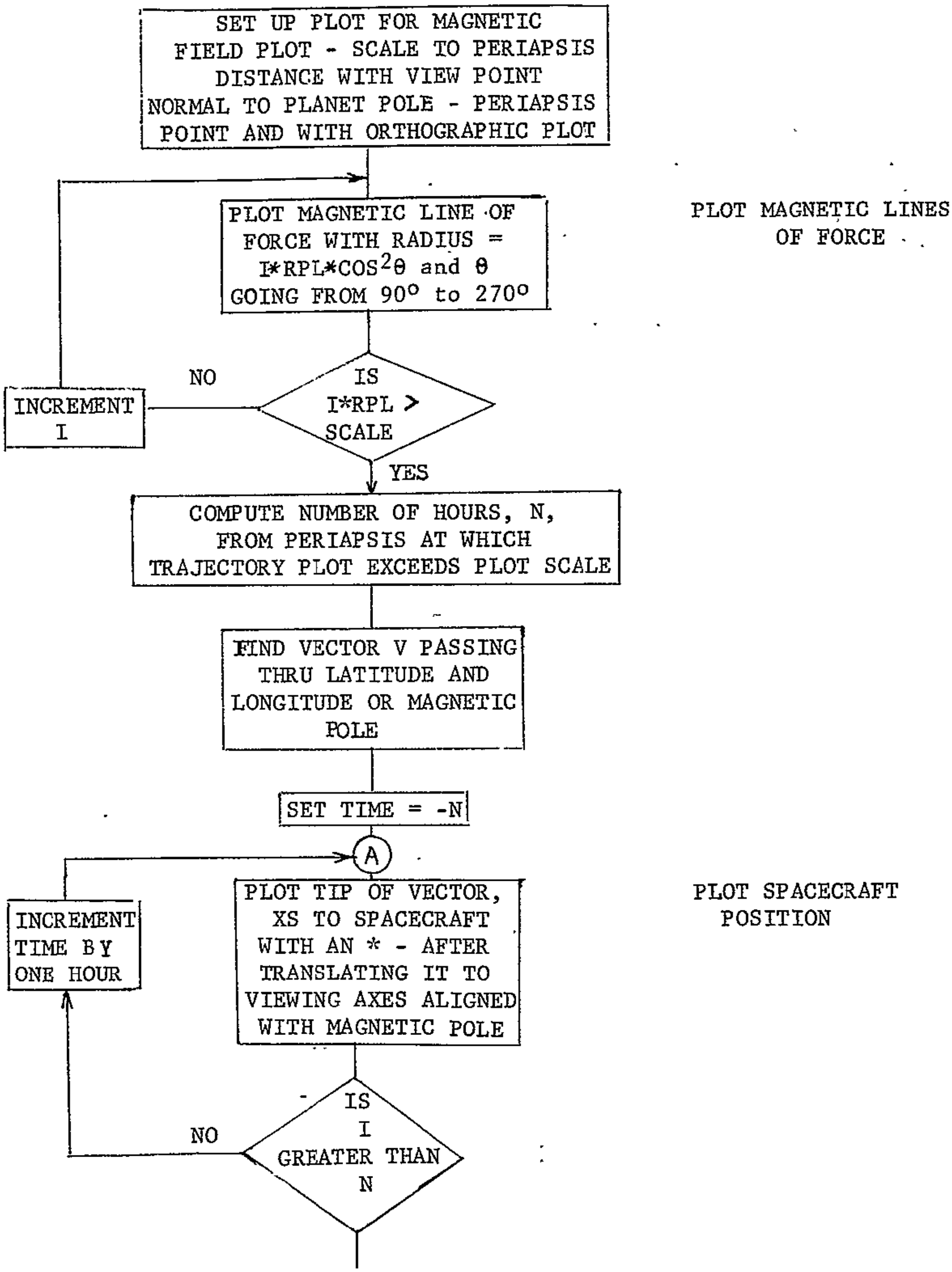


Fig. 6-2 FIELDS Flow Diagram

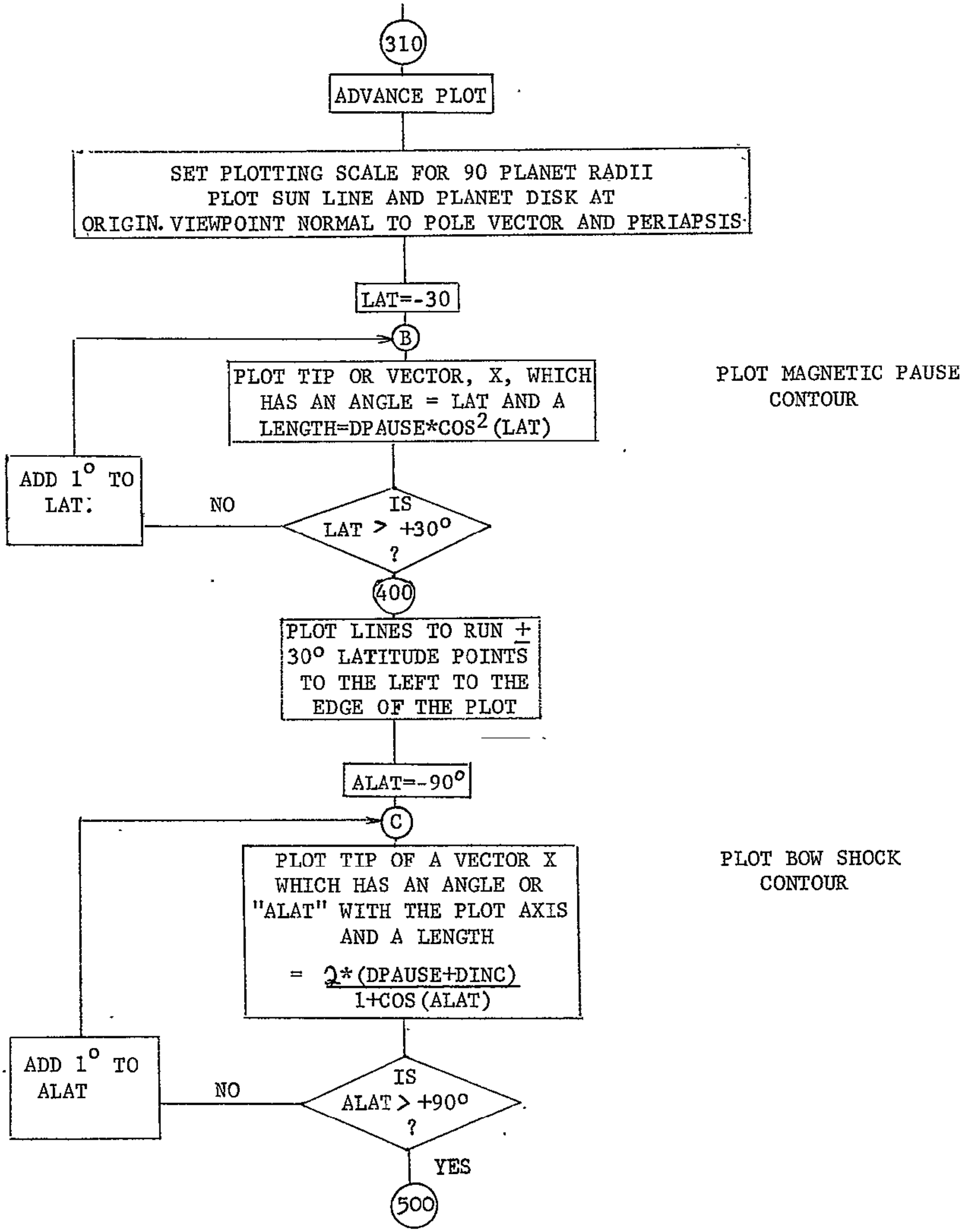

310
ADVANCE PLOT
SET PLOTTING SCALE FOR 90 PLANET RADII
PLOT SUN LINE AND PLANET DISK AT
ORIGIN. VIEWPOINT NORMAL TO POLE VECTOR AND PERIAPSIS
LAT=-30
B
PLOT TIP OR VECTOR, X, WHICH
HAS AN ANGLE = LAT AND A
LENGTH=DPAUSE*COS2 (LAT)
PLOT MAGNETIC PAUSE
CONTOUR
ADD 1° TO
LAT.
NO
IS
LAT > +30°
?
400
PLOT LINES TO RUN ±
30° LATITUDE POINTS
TO THE LEFT TO THE
EDGE OF THE PLOT
ALAT=-90°
C
PLOT TIP OF A VECTOR X
WHICH HAS AN ANGLE OR
"ALAT" WITH THE PLOT AXIS
AND A LENGTH
= 2*(DPAUSE+DINC) / 1+COS(ALAT)
PLOT BOW SHOCK
CONTOUR
ADD 1° TO
ALAT
NO
IS
ALAT > +90°
?
YES
500

Fig. 6-2 (cont)

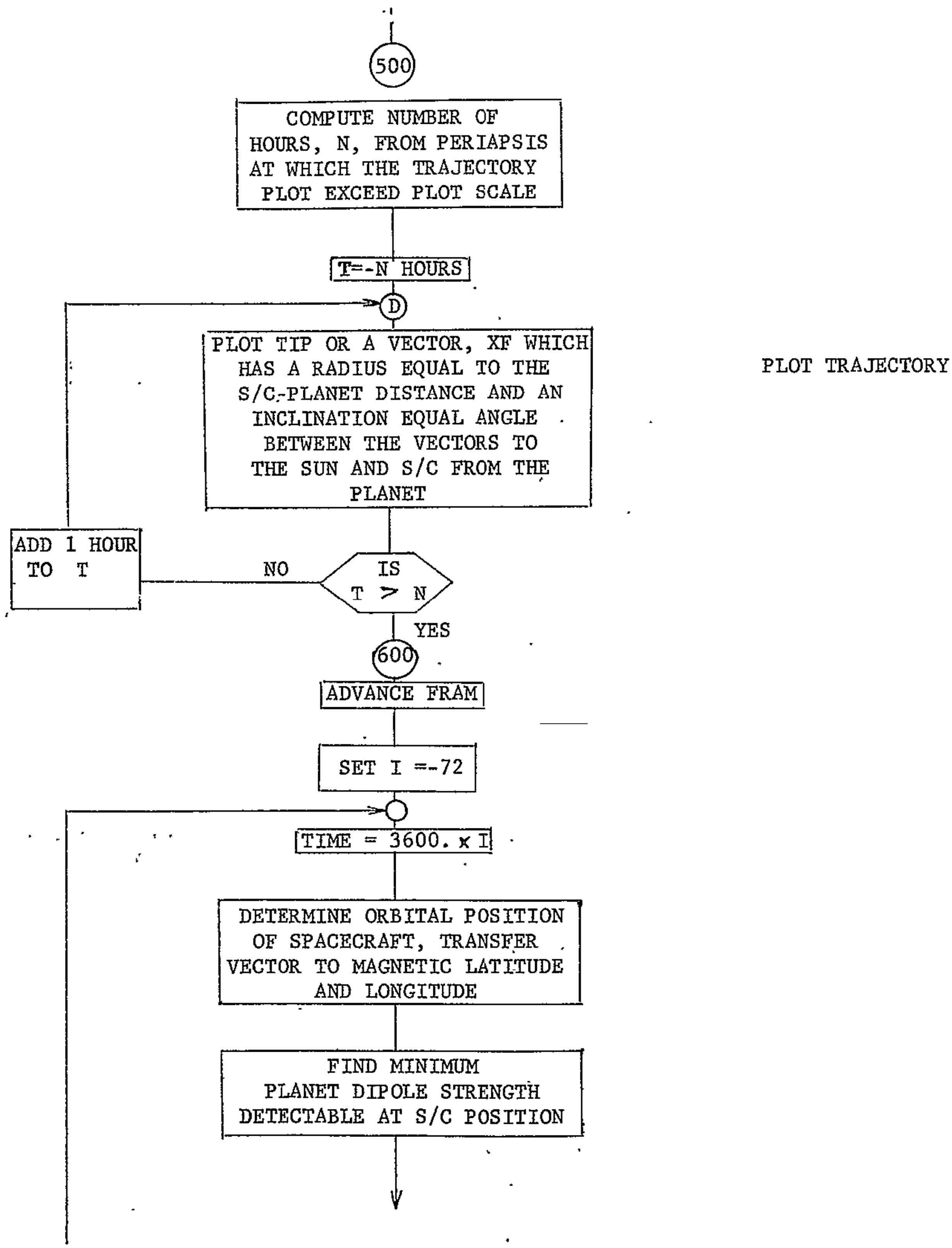

500
COMPUTE NUMBER OF HOURS, N, FROM PERIAPSIS AT WHICH THE TRAJECTORY PLOT EXCEED PLOT SCALE
T=-N HOURS
D
PLOT TIP OR A VECTOR, XF WHICH HAS A RADIUS EQUAL TO THE S/C-PLANET DISTANCE AND AN INCLINATION EQUAL ANGLE BETWEEN THE VECTORS TO THE SUN AND S/C FROM THE PLANET
PLOT TRAJECTORY
ADD 1 HOUR TO T
NO
IS T > N
YES
600
ADVANCE FRAM
SET I =-72
TIME = 3600. x I
DETERMINE ORBITAL POSITION OF SPACECRAFT, TRANSFER VECTOR TO MAGNETIC LATITUDE AND LONGITUDE
FIND MINIMUM PLANET DIPOLE STRENGTH DETECTABLE AT S/C POSITION

Fig. 6-2 (cont)

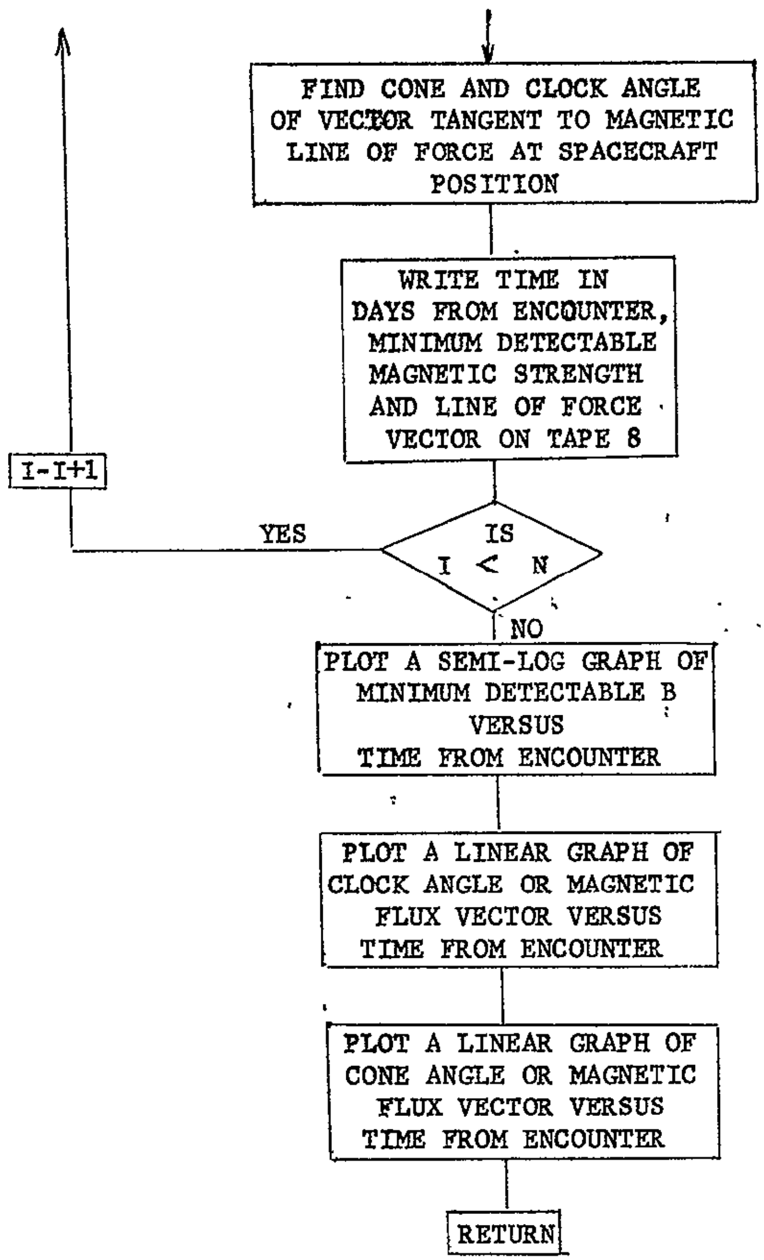

FIND CONE AND CLOCK ANGLE OF VECTOR TANGENT TO MAGNETIC LINE OF FORCE AT SPACECRAFT POSITION
WRITE TIME IN DAYS FROM ENCOUNTER, MINIMUM DETECTABLE MAGNETIC STRENGTH AND LINE OF FORCE VECTOR ON TAPE 8
I-I+1
YES
IS I < N
NO
PLOT A SEMI-LOG GRAPH OF MINIMUM DETECTABLE B VERSUS TIME FROM ENCOUNTER
PLOT A LINEAR GRAPH OF CLOCK ANGLE OR MAGNETIC FLUX VECTOR VERSUS TIME FROM ENCOUNTER
PLOT A LINEAR GRAPH OF CONE ANGLE OR MAGNETIC FLUX VECTOR VERSUS TIME FROM ENCOUNTER
RETURN

Fig. 6-2 (concl)

6.3.3 FOLDER

Purpose: To integrate by wavelength bands, surface brightness (and effective brightness as determined by camera characteristics) for the subject planet and satellite.

Calling Sequence: CALL FOLDER (INST)

Input/Output:

I/O	FORTRAN Name	Definition
I	INST	Instrument index
I	SOLAR(31)	Solar flux at 1 A.U. ($watts/cm^2/micron$)
I	ALBDO(31,3)	Surface albedo three modals
I	CAMFF(31,3)	Camera sensitivity (three curves)
I	F(31,3)	Filter transmissibility three colors
I	MULT	Albedo model multiplier for planet or satellite
O	BO	Brightness for wavelength band ($watts/cm^2/micron$)
O	BEF	Effective brightness for wavelength band
O	BP	Total planet surface brightness ($watts/cm^2$)
O	EFB	Total effective planet brightness ($watts/cm^2$)
O	IBS(8)	Total surface brightness (satellite) ($watts/cm^2$)
O	EFBS(8)	Total effective brightness (satellite) ($watts/cm^2$)

Subprograms Required: None

Approximate Storage Required (octal): 371

Discussion: The following integration is performed

$$B = \sum_{i=1}^{31} \left(SOLAR_i \quad ALBDO_i \cdot .025 \cdot MULT_i \cdot F_i\right)$$

$$BEF = \sum_{i=1}^{31} B_i \ CAMEF_i$$

Table 6-3 presents the FOLDER listing; Fig. 6-3 is the FOLDER flow diagram.

Table 6-3 FOLDER Listing

```
 SUBROUTINE FOLDER(INST)
 DIMENSION NS(9)
 DATA NS/0,0,1,2,4,7,5,1,0/
 DIMENSION PLNMUL(9),MOD(9),SATMUL(8,4)
 REAL MULT
 INTEGER CAMK
 COMMON/HEDING/TITLE(13)
 COMMON/TNME/TVNAME(5)
 COMMON/TEST/ITEST(97)
 COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
 COMMON /JUNK/BP,EFBP,BS(8),EFBS(8)
 COMMON/PHOTO/SOLAR(31),ALBDO(31,3),F(31,4),CAMEF(31,3)
 COMMON/CHARAC/P(60)
 DATA  F   / 31*1.0,               5*0.,.035,.500,.860,.990,.960,.500,
X .080,.030,.010,.015, 21*.0,
X             .150,.010,.150,.020,.080,.500,.950,.930,.895,.500,.100,
X .027,.015,.013,18*0.,
X             .005,.010,.020,.005,.005,.010,.025,.080,.500,.850,.930,
X .750,.800,.160,.050,.015,.005,.010,.010,.025,.005,.005,.010,.040,
X .030/
 DATA SOLAR /.0070,.0204,.0514,.0971,.1093,.1157,.1429,.1693,.2006,
X.2044       ,.1942,.1852,.1725,.1719,.1666,.1586,.1511,.1442,.1369,
X.1302,.1235,.1171,.1107,.1048,.0988,.0939,.0889,.0862,.0835,.0791,
X.0746/
 DATA ALBDO/.300,.282,.270,.265,.269,.272,.285,.315,.364,.385,.406,
X.430,.450,.460,.465,.469,.470,.469,.458,.440,.410,.380,.355,.330,
X.315,.300,.282,.264,.246,.232,.222,
X.045,.050,.055,.060,.065,.070,.075,.080,.085,.100,.105,.113,.120,
X.125,.130,.135,.145,.155,.165,.170,.175,.180,.185,.290,.295,.200,
X.200,.200,.205,.208,.210,
X.075,.060,.047,.045,.040,.045,.050,.060,.075,.100,.120,.140,.155,
X.180,.210,.225,.250,.260,.275,.280,.290,.295,.300,.300,.302,.300,
X.300,.300,.298,.295,.290/
 DATA CAMEF/.0   ,.002,.095,.476,.762,.953,.984,1.00,.984,.953,.794,
X .762,.682,.555,.476,.445,.360,.254,.206,.159,.095,.040,.024,.010,
X .003,.022, 10*.0
X                                   ,.238,.799,.905,.954,.976,1.00,
X 1.00,.994,.960,.916,.870,.834,.810,.799,.785,.740,.685,.620,.542,
X .477,.417,.357,.310,.250,.202,.167,
X                        .025,.050,.100,.200,.500,.750,1.00,.910,.700,
X .565,.450,.415,.365,.340,.245,.092,.025,.009,13*.0/
 DATA PLNMUL/1.0,1.0,1.0,1.0,1.0,1.04,1.28,1.07,0.655/
 DATA MOD/1,1,1,3,1,1,1,1,4/
 DATA SATMUL/
X4.00,4.00,4.00,1.99,2.26,2.32,4.00,0.50,
X3.72,4.00,4.00,4.00,4.00,1.79,1.31,3.75,
X4.00,4.00,4.00,4.00,4.00,0.00,0.00,0.00,
X2.67,4.00,0.00,0.00,0.00,0.00,0.00,0.00/
```

Table 6-3 (concl)

```
      NSP=1+NS(IP)
      CAMK=P(21) + 0.01
      NF=P(23) + 0.01
C
      DO 200 KIL=1,NSP
      IMOD=2
      IF(KIL.EQ.1) IMOD=MOD(IP)
      MULT=PLNMUL(IP)
      IF(KIL.NE.1) MULT=SATMUL(KIL,IP-4)
      B=0.
      EFB=0.
      WL=.25
      WRITE(6,101)TITLE
      WRITE(6,100)TVNAME
      IF(KIL.NE.1) GO TO 70
      WRITE(6,160) IP
      GO TO 80
   70 CONTINUE
      KIK=KIL-1
      WRITE(6,170) KIK
   80 CONTINUE
      WRITE(6,110)
      DO 10 I=1,31
      BO=SOLAR(I)*ALBDO(I,IMOD)*0.025*MULT
      IF(NF.NE.0) BO=BO*F(I,NF)
      BEF=BO*CAMEF(I,CAMK)
      WRITE(6,120) WL,SOLAR(I),ALBDO(I,IMOD),CAMEF(I,CAMK),F(I,NF),BO,
     * BEF
      WL=WL+.025
      B=B+BO
10    EFB=EFB+BEF
      IF(KIL.NE.1) GO TO 150
      PB=B
      EFBP=EFB
      GO TO 200
  150 CONTINUE
      BS(KIL-1)=B
      EFBS(KIL-1)=EFB
  200 CONTINUE
      RETURN
  100 FORMAT(5X,19HTV ENERGY DATA FOR ,5A6,//)
  101 FORMAT(1H1,4X,13A6,//)
  110 FORMAT(4X,7HMICRONS,4X,5HSOLAR,5X,6HALBEDO,4X,6HSENSOR,4X,6HFILTER
     *,4X,10HBO(W/SQCM),4X,13HBEFF(W/SQCM) )
  120 FORMAT(5(4X,F6.4),2X,2E14.6)
  160 FORMAT(5X,7HPLANET ,I2,5H DATA/)
  170 FORMAT(5X,10HSATELLITE ,I2,5H DATA/)
      END
```

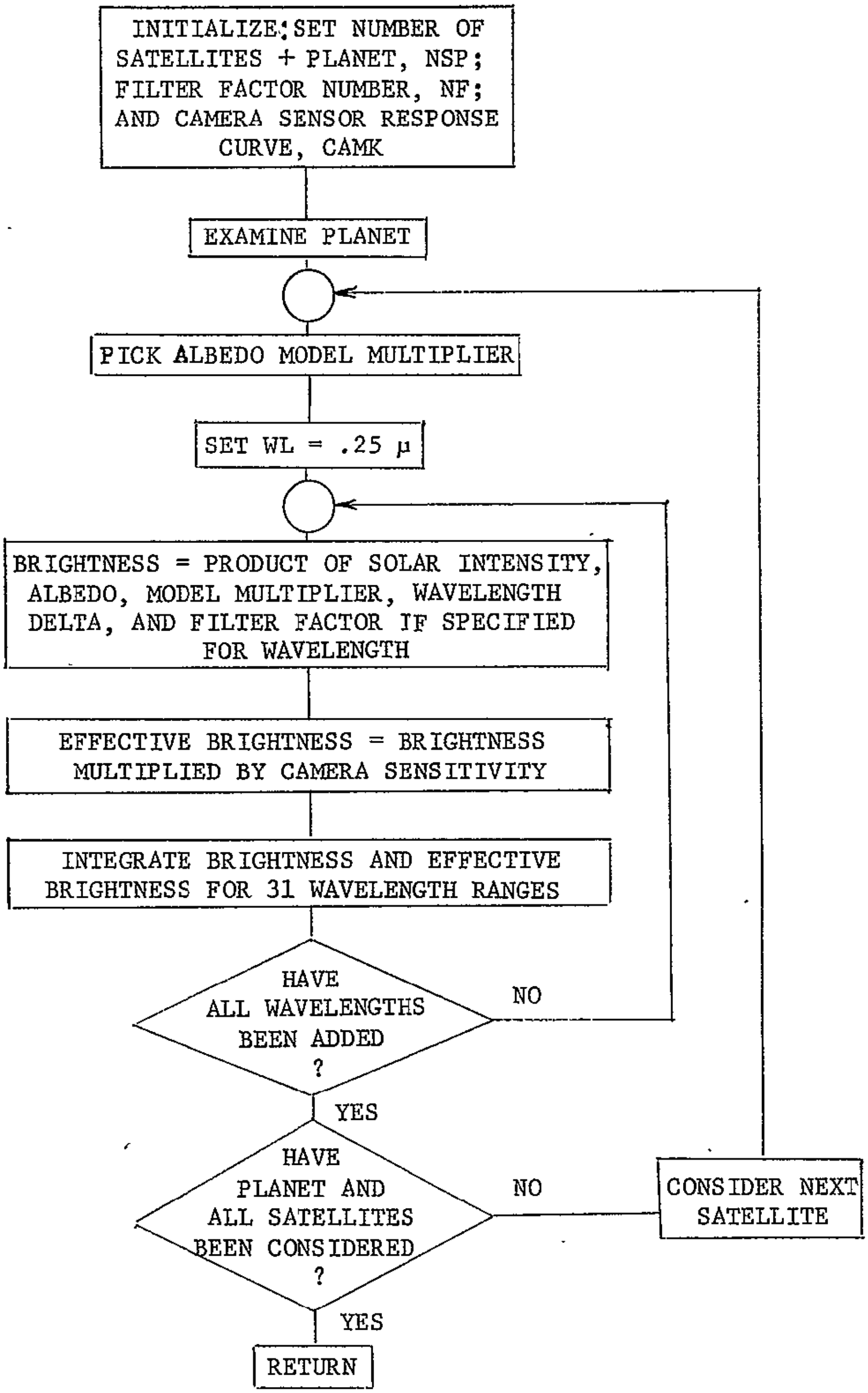


Fig. 6-3 FOLDER Flow Diagram

6.3.4 GEOPLT

Purpose: To provide two polar plots centered around cone angles of 0 and 180 degrees with cone and clock grid lines. Each time the subroutine is called, the cone and clock positions of the planet and indicated satellite are determined and plotted.

Calling Sequence: CALL GEOPLT

Input/Output:

I/O	FORTRAN Name	Definition
I	RPL	Radius of planet (km)
I	RPP	Radius of satellite (km)
I	R(3)	Range, cone, & clock from S/C to planet
I	RSAT(3)	Range, cone & clock from S/C to satellite
I	NREC	Number of record
I	IP	Planet number
I	IVS	Satellite number
I	NCB	Planet/satellite flag: =0 planet, =1 satellite
O	THA	Half angle of planet/satellite (radians)
O	R(3)	Range, cone, & clock from S/C to planet/ spacecraft.
O	DEI	Scaling factor for plot (inches/degree)
O	K	Plot index = 1 departure; = 2 arrival

Subprograms Required: CHAROP, MAP, SETBEM, VECTR1, LINE, NUMBER CIRC, VEQUAL, PSCONS, XLIMB, VLOAD, SPHERE, POINT, SYMBOL, DOT ADV.

Approximate Storage Required (octal): 1102

Discussion: None.

Table 6-4 presents the GEOPLT listing; Fig. 6-4 is the GEOPLT flow diagram.

Table 6-4 GEOPLT Listing

```
      SUBROUTINE  GEOPLT
      DIMENSION X(3),XP(3),XSAT(3),XE(3),RADP(3),TARGE(3),TARGP(3),
     XRSAT(3),XSS(3)
      DIMENSION A(3),V(3),V1(3)
      DIMENSION NUM(8)
      COMMON/PTLB/THA,R(3),DEI,K
      COMMON /PLCONS/GMS,GMPL(12),RS,RPL(12),SNAME(2),PLNAME(2,12),
     X SPV(3),PLPV(3,12),SVE(3),PLVE(3,12),SROT,PLROT(12),SOBL,PLOBL(12)
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      DIMENSION SCUNIT(4)
      DATA NUM/30,60,90,120,150,180,270,0/
      DATA SCUNIT/0.,1.,0.,1./
      SCAL2=2.625
      CALL CHAROPT(0,0,1,0,0)
      K=1
       KREC=0
      PI=3.1415926536
       RAD=57.2957795
        STEP=2./RAD
      SCALE=3.5
   50 CALL MAP(-SCALE,SCALE,-SCALE,SCALE,SCUNIT(1),SCUNIT(2),SCUNIT(3),
     XSCUNIT(4))
      CALL SET BEAM(-SCALE,-SCALE)
      CALL VECTOR (SCALE,-SCALE)
      CALL VECTOR (SCALE, SCALE)
      CALL VECTOR (-SCALE,SCALE)
      CALL VECTOR (-SCALE,-SCALE)
      CALL LINE(-SCALE,-SCALE/2.,SCALE,-SCALE/2.)
      CALL MAP(-3.5,3.5,-2.625,2.625,0.,1.,.25,1.)
      CALL LINE(-3.5,0.,3.5,.0)
      CALL LINE(0.,-2.625,0.,2.625)
      DEI=3.5/90.
      R0=15.*DEI
      DO 12 I=10,360,10
      TH=I/RAD
      SN=SIN(TH-PI/2.)
      CS=COS(TH-PI/2.)
      X1=R0*CS
      Y1=R0*SN
      RMAX=3.5
      X2=RMAX*CS
      Y2=RMAX*SN
      IF(ABS(Y2).LT.2.625) GO TO 12
      Y2=SIGN(2.625,Y2)
      SL=SN/CS
      X2=Y2/SL
   12 CALL LINE (X1,Y1,X2,Y2)
      DO 15 I=1,6
      R=R0*I
   15 CALL CIRC(R ,STEP)
       CALL SETBEAM(.08,-SCAL2)
      CALL NUMBER(NUM(8),2HI1)
```

Table 6-4 (cont)

```
      CALL SET BEAM(.08,SCAL2- .17)
      CALL NUMBER(NUM(6),2HI3)
      CALL SET BEAM (-SCALE+.01,.0)
      CALL NUMBER(NUM(7),2HI3)
      IF(K.EQ.2) GO TO 21
      DO 20 I=1,3
      RO=I*R0*2.-.25
      CALL SET BEAM(RO,0)
20    CALL NUMBER(NUM(I),2HI3)
      GO TO 23
   21 DO 22 I=1,3
      M=7-I
      RO=2.*(I-1)*R0-.25
      CALL SET BEAM(RO,0)
 22   CALL NUMBER(NUM(M),2HI3)
23    REWIND 7
10     READ(7) TT,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT,PHASES,XSS,
     XJDV,EM
      IF(EOF,7)31,11
   11 KREC=KREC+1
      TMAX=TT
      TMIN=TT
      IF(NCB  .EQ.1) CALL VEQUAL(R,RSAT)
       IF(K.EQ.1.AND.R(2).GT.90.) GO TO 10
       IF(K.EQ.2.AND.R(2).LT.90.) GO TO 10
      RPP=RPL(IP)
      IF(IFLST.EQ.1.OR.NCB.EQ.1)CALL PSCONS(RPP,6HRADIUS,IVS,IP)
      THA=ASIN(RPP/R(1))
      CALL XLIMB
      CALL VLOAD(A,1.,R(2),R(3))
      CALL SPHERE(V,A,4HFROM ,5HPOLAR,6HDEGREE)
6      READ(7) TT,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT,PHASES,XSS,
     XJDV,EM
      IF(EOF,7) 30,7
    7 KREC=KREC+1
      IF(NCB  .EQ.1) CALL VEQUAL(R,RSAT)
      IF(K.EQ.1.AND.RSAT(2).GT.90.) GO TO 40
      IF(K.EQ.2.AND.RSAT(2).LT.90.) GO TO 40
      CON=DEI*ABS(180.*(K-1)-RSAT(2))
      ZON=RSAT(3)*PI/180.-PI/2.
      X=CON*COS(ZON)
      Y=CON*SIN(ZON)
      CALL POINT(X,Y)
      IF(RSAT(1).LT.R(1)) CALL SYMBOL(3H*$.)
      IF(RSAT(1).GT.R(1)) CALL SYMBOL(3H+$.)
   40 CONTINUE
      IF(K.EQ.1.AND.R(2).GT.90.) GO TO 6
      IF(K.EQ.2.AND.R(2).LT.90.) GO TO 6
      TMAX=AMAX1(TMAX,TT)
      TMIN=AMIN1(TMIN,TT)
```

Table 6-4 (concl)

```
      CALL VLOAD(A,1.,R(2),R(3))
      CALL SPHERE(V1,A,4HFROM,5HPOLAR,6HDEGREE)
      CSA=DOT(V,V1)
      SNA=SQRT(1.-CSA**2)
      THA=RPP/R(1)
      IF(SNA.LT.THA) GO TO 6
      THA=ASIN(THA)
      CALL XLIMB
      CALL VEQUAL(V,V1)
      IF(KREC.EQ.NREC) GO TO 30
      GO TO 6
   31 CONTINUE
  30  K=K+1
      CALL ABSBEAM(.1,.05)
      CALL SYMBOL(14HSTART TIME  $.)
      CALL NUMBER(TMIN/86400.,4HF4.1)
      CALL SYMBOL(13HSTOP TIME  $.)
      CALL NUMBER(TMAX/86400.,4HF4.1)
      IF(K.GT.2) RETURN
      CALL FRAME
      GO TO 50
      END
```

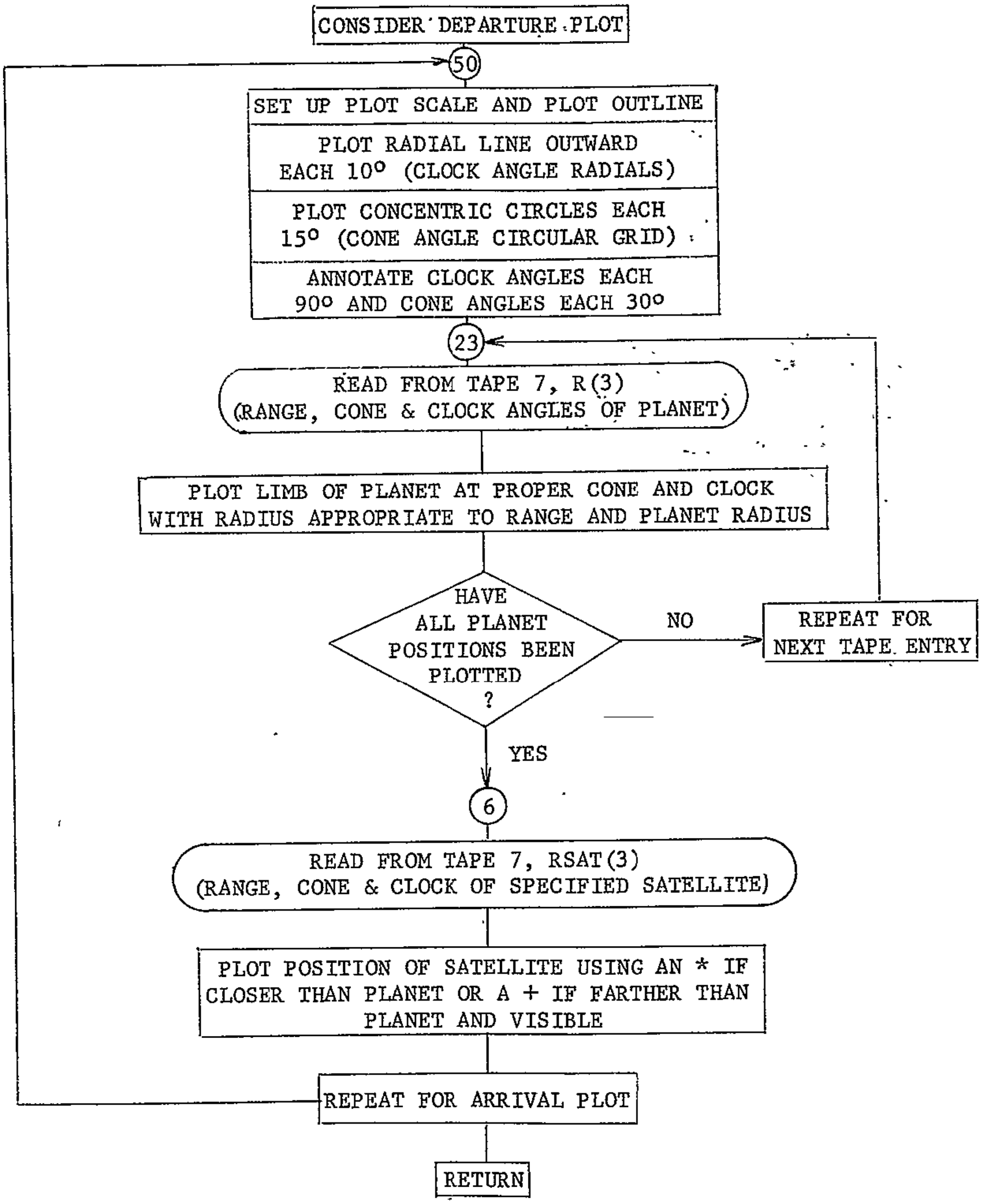


Fig. 6-4 GEOPLT Flow Diagram

6.3.5 INPUT

Purpose: To provide a method of reading all input data required for a given mission

Calling Sequence: CALL INPUT(NP)

Input/Output:

I/O	FORTRAN Name	Definition
I	TITLE (13)	Mission title (78 possible characters)
I	XZ (97)	Input variable value
O	VAR (97)	Input variable name (six letter word)
O	ITEST (97)	Input flag set to 1 when input is read in; otherwise set to 0 for canned input.
I	I	Input index 1-97 input variable = 98 end of case flag = 99 end of run flag
O	CCONE (200)	User input cone angle
O	CCLOK (200)	User input clock angle
O	TCONS (200)	User input time from periapsis (days)
O	NP	Numer of cone, clock, & time inputs read
O	IFLT	Pointing mode flag = 0 instrument points to center of planet or halfway between terminator and lit limb if center is on dark side. = 1 if instrument is pointed by user's cone and clock angles

Subprograms Required: STERM, VECTOR

Approximate Storage Required (octal): 556

Discussion: None

Table 6-5 presents the INPUT listing; Fig. 6-5 is the INPUT flow diagram.

Table 6-5 INPUT Listing

```
      SUBROUTINE INPUT(NP)
      COMMON/HEDING/TITLE(13)
      COMMON/FLAGS/IFLT,IFLPL,IFLPR
      COMMON/TCONS/TCONS(200)
      COMMON/CAM/CCONE(200),CCLOK(200)
      COMMON/TEST/ITEST(97)
      COMMON /TRA/TRACE,COMENT(13)
      COMMON/TNME/TVNAME(5)
      COMMON /PLTBUF/BUF(20)
C INPUT QUANTITIES
      COMMON/XZ/XZ(97) /VAR/VAR(97)
      DATA XZ/14*0.,3.,8*0.,2*1H ,72*0./
      DATA VAR/6H      ,6HPLANET,6HSATEL ,6HDATE  ,6HORIENT,6H      ,
     *6HSTATE1,6HSTATE2,6HSTATE3,6HSTATE4,6HSTATE5,6HSTATE6,6H      ,
     *6H      ,6H      ,6H      ,6H      ,6H      ,6H      ,6H      ,
     *6H      ,6HINSTID,6H      ,6H      ,6H      ,6H      ,6H      ,
     *6H      ,6H      ,6H      ,6HNAME-1,6HNAME-2,6HNAME-3,6HNAME-4,
     *6HNAME-5,6HCHAR-1,6HCHAR-2,6HCHAR-3,6HCHAR-4,6HCHAR-5,6HCHAR-6,
     *6HCHAR-7,6HCHAR-8,6HCHAR-9,6HCHAR10,6HCHAR11,6HCHAR12,6HCHAR13,
     *6H      ,6H      ,6H      ,6H      ,6H      ,6H      ,6H      ,
     *6H      ,6H      ,6H      ,6H      ,6H      ,6HTSTAR ,6HTSTOP ,
     *6HDTINST,6H      ,6H      ,6HBITRAT,6H      ,6H      ,6HCALCST,
     *6H      ,6H      ,6H      ,6H      ,6H      ,6H      ,6H      ,
     *6H      ,6H      ,6H      ,6H      ,6HDPLOT ,6HDPRNT ,6H      ,
     *6H      ,6H      ,6HSATRNG,6H      ,6H      ,6H      ,6H      ,
     *6H      ,6H      ,6H      ,6H      ,6H      ,6H      ,6H      /
      DATA TVNAME/5*6H      /
C
C EJECT PAGE AND READ TITLE CARD
      PRINT 998
      READ(5,1000)TITLE
 1000 FORMAT(13A6)
      WRITE(6,1001)TITLE
 1001 FORMAT(5X,13A6,/)
C
      N1=0
      N2=0
      NP=0
      ITFLB=0
      ITFLA=0
      IFLT=0
      IF(XZ(71).NE..0) GO TO 1
      DO 10 I=1,97
10    ITEST (I)=0
  250 CONTINUE
C
C  READ INPUT CARDS
1     READ(5,994)              I,XV,WORD,(COMENT(J),J=1,10)
  260 CONTINUE
      IF(I.LT.1 .OR. I.GT.97) GO TO 2
      PRINT 995, I,XV,VAR(I),(COMENT(J),J=1,10)
      IF(I.GE.31.AND.I.LE.35) TVNAME(I-30)=WORD
```

Table 6-5 (cont)

```
      XZ(I)=XV
      ITEST(I)=1
      IF(I.NE.22) GO TO 200
      IF(IFIX(XV/100.).NE.3) ITFLB=1
  200 CONTINUE
      IF(I.EQ.23) GO TO 300
      GO TO 1
  300 CONTINUE
      IFLT=XV
      IF(XV.NE.1) GO TO 250
      IF(ITFLB.EQ.1) GO TO 310
      WRITE(6,993) COMMENT
      GO TO 250
  310 ITFLA=1
  320 READ(5,994) I,XV,WORD,(COMENT(J),J=1,10)
      IF(I.LT.24.OR.I.GT.30) GO TO 260
      IF(I.EQ.24) N1=N1+1
      IF(I.EQ.25) N2=N2+1
      IF(I.EQ.26) NP=NP+1
      ITEST(I)=1
      XZ(I)=XV
      PRINT 995,I,XV,VAR(I),(COMENT(J),J=1,10)
      IF(I.EQ.24) CCONE (N1)=XV
      IF(I.EQ.25) CCLOK(N2)=XV
      IF(I.EQ.26) TCONS(NP)=XV
      GO TO 320
15    XZ(I) = WORD
      PRINT 991, I,WORD,VAR(I),(COMENT(J),J=1,10)
      GO TO 1
2     PRINT 996, I,WORD,(COMENT(J),J=1,10)
      IF(I.EQ.98) GO TO 5
      IF(I.EQ.99) GO TO 4
      DO 3 I=1,97
3     XZ(I) = 0.
      XZ(15) = 3.
      XZ(24) = 1H
      XZ(25) = 1H
      GO TO 1
C FINISHES PLOT TAPE
4     CALL STERM(BUF)
      STOP
5     CALL VECTOR1(WORD)
      IF(TRACE.EQ.5HTRACE) PRINT 999
C
      PRINT 998
      WRITE(6,1001)TITLE
      WRITE(6,1002)
 1002 FORMAT(24H XZ(I) VALUES IN STORAGE,/)
      DO 9 I=1,96,2
      K=I+1
    9 WRITE(6,1003)I,ITEST(I),XZ(I),K,ITEST(K),XZ(K)
 1003 FORMAT(I4,I4,2X,E17.10,13X,I4,I4,2X,E17.10)
```

Table 6-5 (concl)

```
C
      RETURN
6     READ(5,992) COMENT
      WRITE(6,993) COMENT
      GO TO 1
991   FORMAT(5X,I2,3X,A6,11X,10A6,A2)
992   FORMAT(13A6)
993   FORMAT(* ERROR*******13A6,*******ERROR THIS CARD IGNORED*)
994   FORMAT(I2,G15.8,10A6,A2)
995   FORMAT(5X,I2,1PG20.8,10A6,A2)
996   FORMAT(5X,I2,20X,10A6,A2)
997   FORMAT(1P4(/1X,5(3X,A6,G15.8))/1X,3(3X,A6,G15.8),2(3X,A6,6X,A6,3X
     X ),15(/1X,5(3X,A6,G15.8)))
998   FORMAT(*1*)
999   FORMAT(*1INPUT SUBROUTINE*)
      END
```

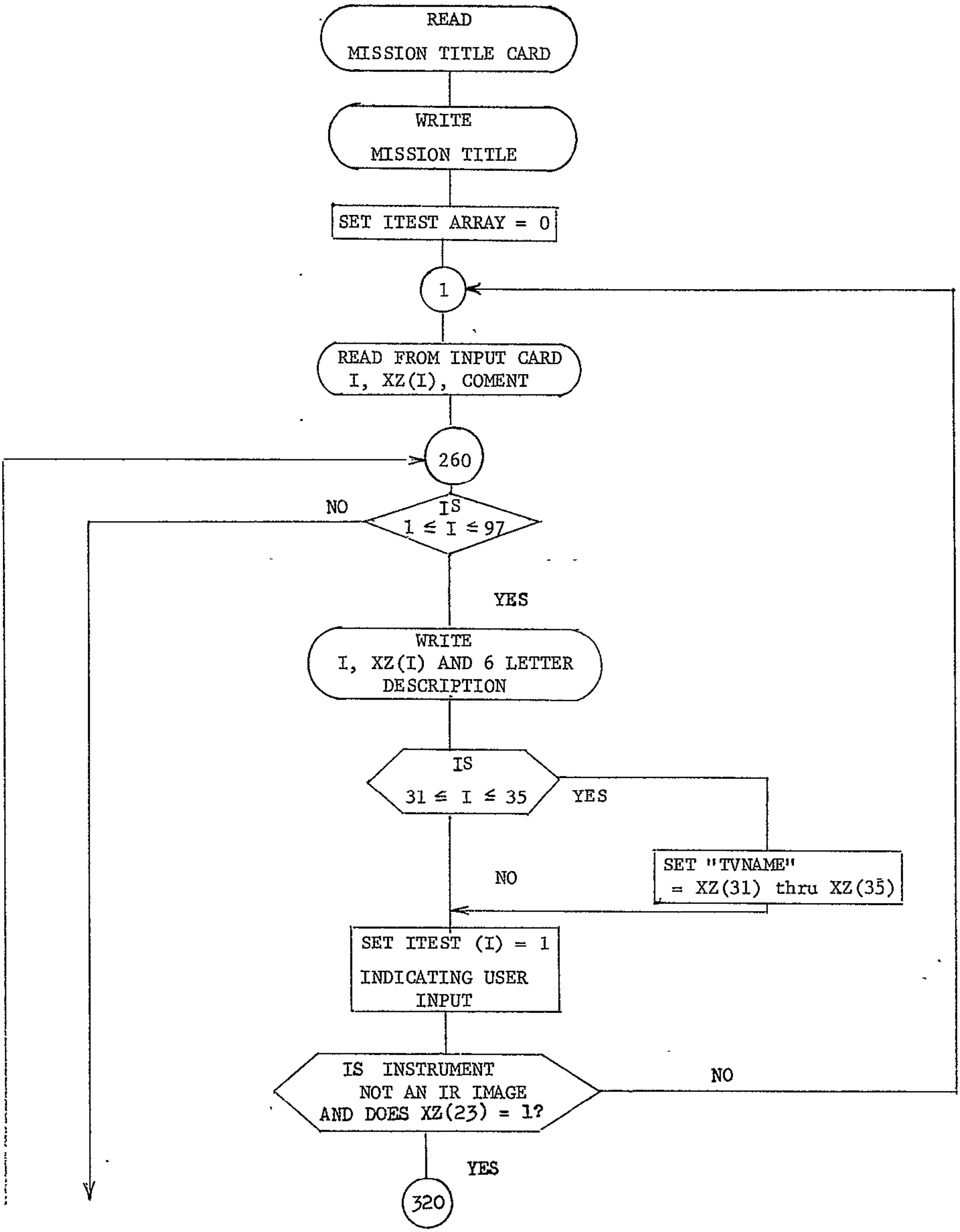


Fig. 6-5 INPUT Flow Diagram

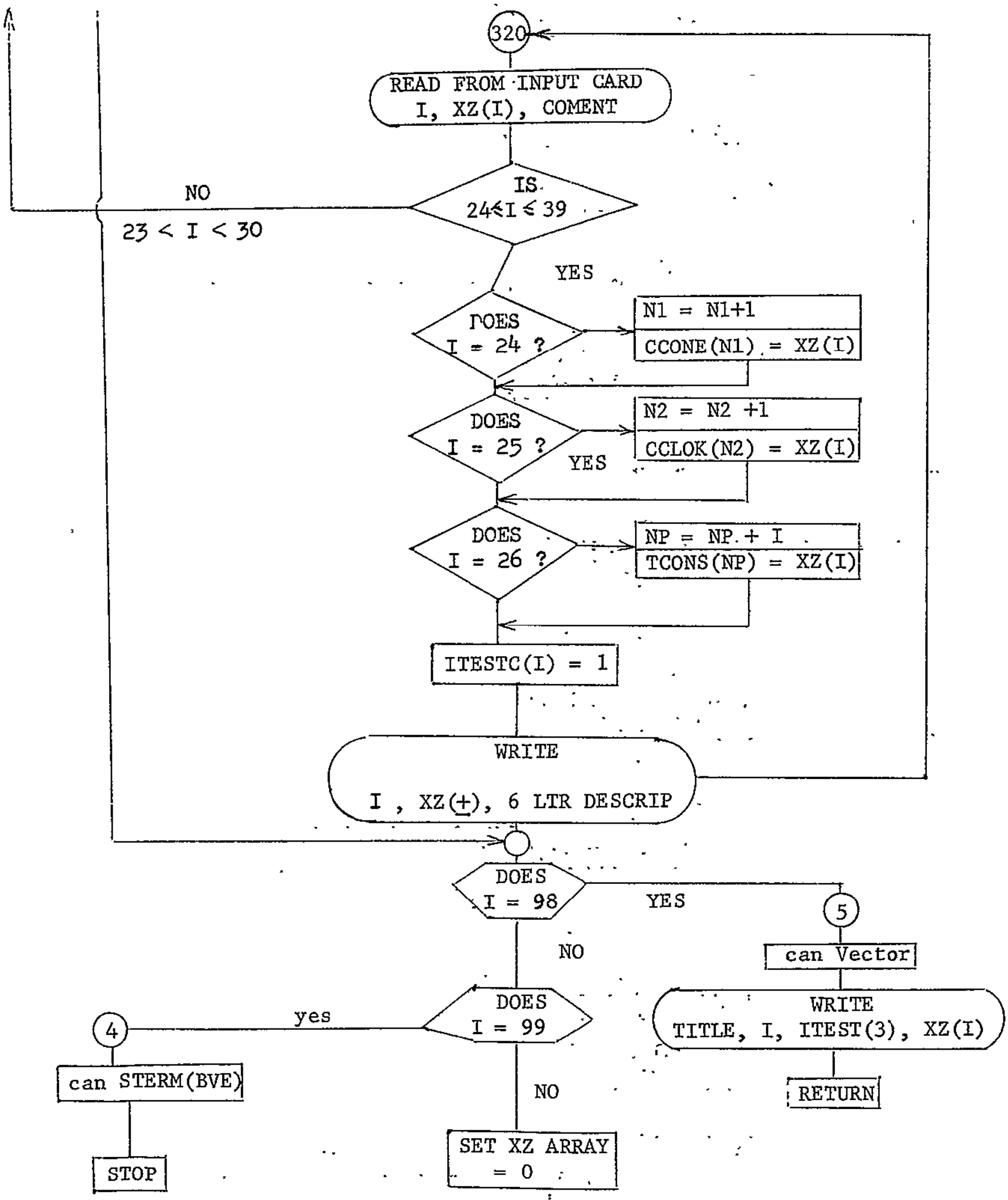

320
READ FROM INPUT CARD
I, XZ(I), COMENT
NO
IS
24≤I≤39
23 < I < 30
YES
DOES
I = 24 ?
N1 = N1+1
CCONE(N1) = XZ(I)
DOES
I = 25 ?
YES
N2 = N2 +1
CCLOK(N2) = XZ(I)
DOES
I = 26 ?
NP = NP + I
TCONS(NP) = XZ(I)
ITESTC(I) = 1
WRITE
I , XZ(+), 6 LTR DESCRIP
DOES
I = 98
YES
5
NO
can Vector
WRITE
TITLE, I, ITEST(3), XZ(I)
RETURN
DOES
I = 99
yes
4
can STERM(BVE)
STOP
NO
SET XZ ARRAY
= 0

Fig. 6-5 (concl)

6.3.6 MRLC

Purpose: To provide a plot of the number of TV pictures that have a resolution of better than four separate input levels as a function of time.

Calling Sequence: CALL MRLC (DIREC)

Input/Output:

I/O	FORTRAN Name	Definition
I	DIREC	Program Mode - plot accumulated Results if DIREC = 'plot', calculate only if not.
I	PLNAME(2,12)	Planet name
I	TIM	Maximum time span for plot scale (Days)
I	REC(9)	Resolution at picture center (Km/Pixel)
I	P(60)	P(17, P(18), P(19) & P(20) are the four required resolution levels (Km/Pixel)
O	PICS (4,9)	Number of frames better than four levels for eight possible satellites and the planet of interest.

Subprograms Required: ADV, MAPSSL, CHAROP, ABSBEM, SYMBOL, PSCONS, LINE, POINT & NUMBER.

Approximate Storage Required (octal): 565

Discussion: None

Table 6-6 presents the MRLC listing; Fig. 6-6 is the MRLC flow diagram.

Table 6-6 MRLC Listing

```
      SUBROUTINE MRLC(DIREC)
      COMMON/PLCONS/DUMMYD(28), PLNAME(2,12),DUMMYE(104)
      COMMON/RAZOR/PICS(4,9),SPPICS(4)
      DIMENSION SYM(4)
      DATA SYM/3HO$.,3HX$.,3H+$.,3HI$./
      COMMON/HANDO/TIM,REC(9),REL(9),RES,SMR(9),SMRR(9),SMRS(9),TSMR(9)
     X,SPSMR,SPSMRS,SPTSMR,SPSMRR
      DIMENSION SDUNIT(4),NS(9),ACPIC(4,8),ION(4,8)        ,THING(3),PP(4)
      DATA THING(3)/2H$./
      DATA ION/32*0/
      COMMON/FOROUT/DUMMYB(13),IPCN,DUMMYC(4)
      COMMON/CHARLIE/NCODE,NREC,IP,IVS,NSC,IFLST,NCB
      COMMON/CHARAC/P(60)
      COMMON/JDAYS/XJDV,TT
      DATA NS/0,0,1,2,4,7,5,2,0/
      DATA SDUNIT/0.1,1.0,0.1,1.0/
      IF(DIREC.EQ.4HPLOT) GO TO 300
      IF(IFR.NE.0) GO TO 100
      REWIND 12
      PMAX=0.0
      IFR=1
C
C        MINIMUM RESOLUTION LEVEL CALCULATION (MRLC) SUBROUTINE
C        WILL COMPUTE THE NUMBER OF PICTURES VS TIME WITHIN A
C        GIVEN RESOLUTION RANGE (SEE IVCHAR P(17) THRU P(20)).
C     BECAUSE CALCULATIONS TAKE PLACE DURING CALC STEPS ACCURACY
C        LIES WITHIN PLUS OR MINUS P(39).
C
      NSP=NS(IP)
      NSX=NSP+1
  100 DO 140 I=1,NSX
      DO 130 J=1,4
      IF(ION(J,I).NE.0) GO TO 110
      IF(REC(I).GT.P(16+J)) GO TO 130
      ION(J,I)=IPCN
      ACPIC(J,I)=1.
  110 CONTINUE
      IF(REC(I).GT.P(16+J)) GO TO 120
      ACPIC(J,I)=ACPIC(J,I)+IPCN-ION(J,I)
120   ION(J,I)=IPCN
      PICS(J,I)=ACPIC(J,I)
      IF(PICS(J,I).GT.PMAX) PMAX=PICS(J,I)
  130 CONTINUE
  140 CONTINUE
      TP=TT/86400.
      WRITE(12) TP,PICS,SPPICS
      RETURN
```

Table 6-6 (concl)

```
C
C         PLOT OF PICTURES PER MRL VS TIME
C
  300 CONTINUE
      DO 400 I=1,NSX
      REWIND 12
      IFR=0
      CALL FRAME
      CALL MAPSSL(-TIM,TIM,1.0,PMAX,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT
     *(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.02)
      CALL SYMBOL(56HNUMBER OF PICTURES WITHIN SPECIFIED RESOLUTION LEVE
     XLS $.)
      IF(I.NE.1) GO TO 310
      DO 305 K=1,2
  305 THING(K)=PLNAME(K,IP)
      GO TO 315
  310 CALL PSCONS(THING,4HNAME,I-1,IP)
  315 CONTINUE
      CALL SYMBOL(THING )
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS)$. )
      CALL CHAROPT(0,0,0,0,0)
  320 READ(12) T,PICS,SPPICS
      IF(EOF,12) 360,325
  325 IF(IFR.EQ.0) GO TO 340
      DO 330 J=1,4
      IF(PP(J).LE.0.0.OR.PICS(J,I).LE.0.0) GO TO 330
      CALL LINE(TP,PP(J),T,PICS(J,I))
  330 CONTINUE
      GO TO 350
  340 CONTINUE
      IFR=1
      DO 345 J=1,4
      IF(PICS(J,I).LE.0.0) GO TO 345
      CALL POINT(TP,PICS(J,I))
      CALL SYMBOL(SYM(J))
  345 CONTINUE
  350 CONTINUE
      DO 355 J=1,4
  355 PP(J)=PICS(J,I)
      TP=T
      GO TO 320
  360 DO 365 J=1,4
      IF(PP(J).LE.0.0) GO TO 365
      CALL POINT(TP,PP(J))
      CALL SYMBOL(SYM(J))
      XHT=0.98-(J-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(J))
      CALL SYMBOL(3H $.)
      CALL NUMBER(P(16+J), 4HF5.1)
      CALL SYMBOL(5H KM$.)
  365 CONTINUE
  400 CONTINUE
      RETURN
      END
```

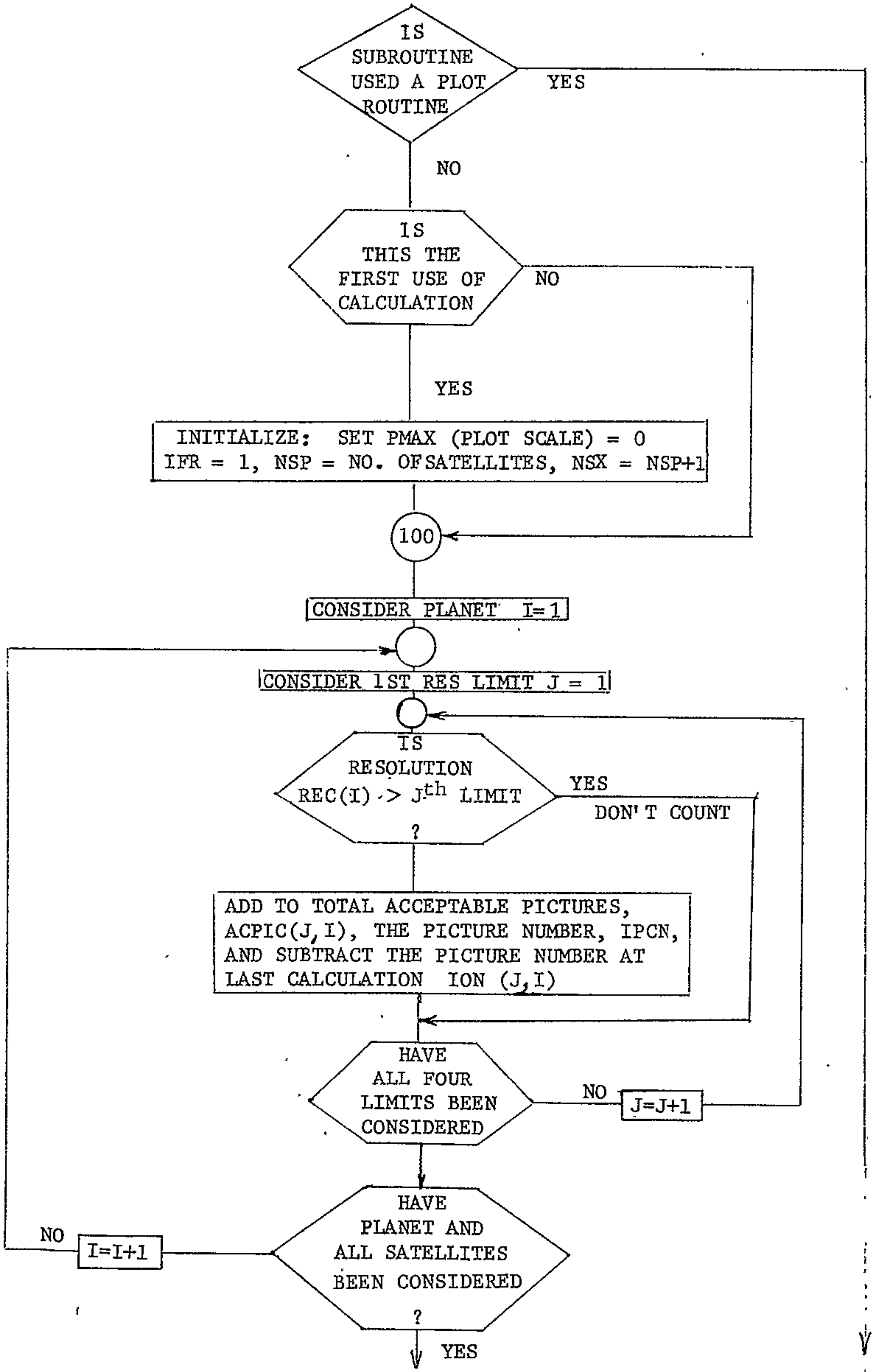


Fig. 6-6 MRLC Flow Diagram

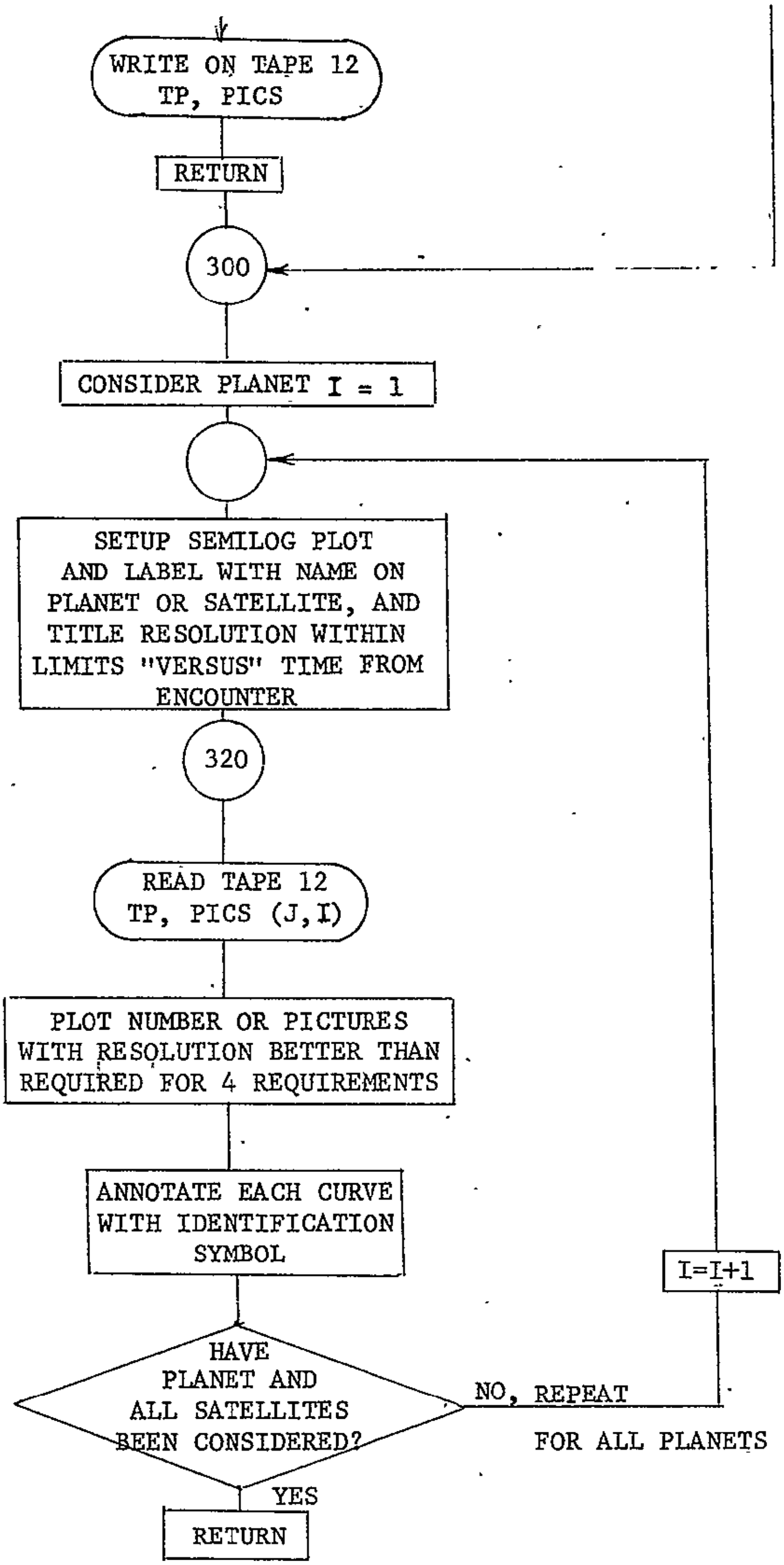

WRITE ON TAPE 12 TP, PICS
RETURN
300
CONSIDER PLANET I = 1
SETUP SEMILOG PLOT AND LABEL WITH NAME ON PLANET OR SATELLITE, AND TITLE RESOLUTION WITHIN LIMITS "VERSUS" TIME FROM ENCOUNTER
320
READ TAPE 12 TP, PICS (J,I)
PLOT NUMBER OR PICTURES WITH RESOLUTION BETTER THAN REQUIRED FOR 4 REQUIREMENTS
ANNOTATE EACH CURVE WITH IDENTIFICATION SYMBOL
HAVE PLANET AND ALL SATELLITES BEEN CONSIDERED?
NO, REPEAT
FOR ALL PLANETS
I=I+1
YES
RETURN

Fig. 6-6 (concl)

6.3.7 OTJAZ

Purpose: To plot various parameters as a function of time for all of the satellites of the subject planet. The plots provided are:

1) Satellite Range versus Time;
2) Satellite Earth Phase Angle (deg) versus Time;
3) Satellite Cone Angle (deg) versus Time;
4) Satellite Clock Angle (deg) versus Time;
5) Sun-Sat-S/C Angle (deg) versus Time;
6) Satellite Half Angle (deg) versus Time;
7) Time required to switch to Satellite at Rate of 1 Deg/Sec versus Time

Calling Sequence: CALL OTJAZ (DIREC)

Input/Output:

I/O	FORTRAN Name	Definition
I	DIREC	Program Mode - Plot results if DIREC = 'PLOT', Calculate only if DIREC = 'CALC'
I	SSV(8)	Sun-satellite-S/C angle for all satellites
I	SAF(8)	Half angle subtended by all satellites
I	DCNCLK(8)	The sum of the difference of the cone angles of the planet and satellite and the difference of the clock angles of the planet and satellites
I	PHASE(8)	Phase angle of all satellites
I	XRCC(3,8)	Range, cone & clock of all satellites from S/C.
I	IP	Planet index number
I	TT	Time from periapsis (sec)

Subprograms required: ORBPOS, ABSV, MAPSSL, CHAROP, ABSBEM, SYMBOL, POINT, LINE, PSCONS, ADV, & MAPS

Approximate Storage Required (octal): 2130

Discussion: None

Table 6-7 presents the OTJAZ listing; Fig. 6-7 is the OTJAZ flow diagram.

Table 6-7 OTJAZ Listing

```
      SUBROUTINE OTJAZ(DIREC)
      COMMON/POBEAR/SSV(8),SAF(8),HANG2,DCNCLK(8)
      DIMENSION DUMMYC(13)
      COMMON/PHASER/PHASE(8),XRCC(3,8)
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      COMMON/JDAYS/JDV,TT
      DIMENSION NS(9),THING(3)
      DIMENSION SYM(8)
      DATA SYM/3H0$.,3HX$.,3H+$.,3HI$.,3HH$.,3HT$.,3HZ$.,3HP$./
      DATA THING(3)/2H$./
      DATA IFR,ITR1/0,11/
      DATA NS/0,0,1,2,4,7,5,2,0/
      REALJDV
      DIMENSION X(3),XP(3),XSAT(3),XE(3),R(3),      RADP(3),TARGE(3),
     *TARGP(3) ,RSAT(3),        XSS(3),SDJNIT(4)   ,RDMIN(3)
      DATA SDUNIT/0.1,1.0,0.1,1.0/
      COMMON/HANDO/TIM,REC(9),REL(9),RES,SMR(9),SMRR(9),SMRS(9),TSMR(9)
     X,SPSMR,SPSMRS,SPTSMR,SPSMRR
      IF(DIREC.EQ.4HPLOT) GO TO 120
      IF(IFR.EQ.1) GO TO 100
      REWIND 9
      RPD=3.14159/180.
      CALL ORBPOS(RDMIN,0.0)
      RAMIN=ABSV(RDMIN)
      NSP=NS(IP)
      REWIND ITR1
      IFR=1
  100 CONTINUE
      T=TT/86400.
      WRITE(ITR1) T,PHASE,XRCC
      WRITE(9) T,SSV,SAF,HANG2,DCNCLK
      RETURN
  120 CONTINUE
C
C  SATELLITE RANGE VS TIME PLOT
C
      REWIND 7
      READ(7)DUMMYC,R,A
      RP=R(1)
      REWIND 7
      CALL FRAME
      CALL MAPSSL(-TIM,TIM,RAMIN,RP,SDUNIT(1),SDUNIT(2),SDUNIT(3),
     *SDUNIT(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(25HRANGE OF SATELLITE (KM)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
      CALL CHAROPT(0,0,0,0,0)
      DO 150 ITB=1,NSP
      REWIND ITR1
      READ(ITR1) TP,PHASE,XRCC
      RP=XRCC(1,ITB)
```

Table 6-7 (cont)

```
      IF(RP.LE.0.0) GO TO 150
      CALL POINT(TP,RP)
      CALL SYMBOL(SYM(ITB))
  130 CONTINUE
      READ(ITR1) T,PHASE,XRCC
      IF(EOF,ITR1)148,140
  140 CONTINUE
      CALL LINE(TP,RP,T,XRCC(1,ITB))
      TP=T
      RP=XRCC(1,ITB)
      GO TO 130
  148 CONTINUE
      CALL POINT(TP,RP)
      CALL SYMBOL(SYM(ITB))
      XHT=0.98-(ITB-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(ITB))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,ITB,IP)
      CALL SYMBOL(THING)
  150 CONTINUE
      REWIND ITR1
      CALL FRAME
C
C  SATELLITE PHASE ANGLE VS TIME PLOT
C
      CALL MAPS (-TIM,TIM,0.0,210.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT
     X(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(29HSATELLITE PHASE ANG_E (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS)$. )
      CALL CHAROPT(0,0,0,0,0)
      DO 153 ITB=1,NSP
      REWIND ITR1
      READ(ITR1) TP,PHASE,XRCC
      PHP=PHASE(ITB)/RPD
      CALL POINT(TP,PHP)
      CALL SYMBOL(SYM(ITB))
  151 CONTINUE
      READ (ITR1) T,PHASE,XRCC
      IF(EOF,ITR1)1053,152
  152 CONTINUE
      PHASES=PHASE(ITB)/RPD
      CALL LINE(TP,PHP,T,PHASES)
      TP=T
      PHP=PHASES
      GO TO 151
 1053 CONTINUE
      CALL POINT(TP,PHP)
      CALL SYMBOL(SYM(ITB))
      XHT=0.98-(ITB-1)*0.03
      CALL ABSBEAM(0.75,XHT)
```

Table 6-7 (cont)

```
      CALL SYMBOL(SYM(ITB))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,ITB,IP)
      CALL SYMBOL(THING)
  153 CONTINUE
      CALL FRAME
C
C  SATELLITE CONE ANGLE VS TIME PLOT
C
      CALL MAPS(-TIM,TIM,0.0,210.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT(
     X4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(28HSATELLITE CONE ANGLE (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
      CALL CHAROPT(0,0,0,0,0)
      DO 156 ITB=1,NSP
      REWIND ITR1
      READ(ITR1) TP,PHASE,XRCC
      RP=XRCC(2,ITB)
      CALL POINT(TP,RP)
      CALL SYMBOL(SYM(ITB))
  154 CONTINUE
      READ (ITR1) T,PHASE,XRCC
      IF(EOF,ITR1)1056,155
  155 CONTINUE
      CALL LINE(TP,RP,T,XRCC(2,ITB))
      TP=T
      RP=XRCC(2,ITB)
      GO TO 154
 1056 CONTINUE
      CALL POINT(TP,RP)
      CALL SYMBOL(SYM(ITB))
      XHT=0.98-(ITB-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(ITB))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,ITB,IP)
      CALL SYMBOL(THING)
  156 CONTINUE
      CALL FRAME
C
C  SATELLITE CLOCK ANGLE VS TIME PLOT
C
      CALL MAPS(-TIM,TIM,0.0,390.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT(
     X4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(29HSATELLITE CLOCK ANGLE (DEG)$.)
```

Table 6-7 (cont)

```
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS)$. )
      CALL CHAROPT(0,0,0,0,0)
      DO 159 ITB=1,NSP
      REWIND ITR1
      READ(ITR1) TP,PHASE,XRCC
      RP=XRCC(3,ITB)
      CALL POINT(TP,RP)
      CALL SYMBOL(SYM(ITB))
  157 CONTINUE
      READ (ITR1) T,PHASE,XRCC
      IF(EOF,ITR1)1059,158
  158 CONTINUE
      CALL LINE(TP,RP,T,XRCC(3,ITB))
      TP=T
      RP=XRCC(3,ITB)
      GO TO 157
 1059 CONTINUE
      CALL POINT(TP,RP)
      CALL SYMBOL(SYM(ITB))
      XHT=0.98-(ITB-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(ITB))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,ITB,IP)
      CALL SYMBOL(THING)
  159 CONTINUE
      REWIND ITR1
      CALL FRAME
      CALL MAPS(-TIM,TIM,   0.0,210.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),
     XSDUNIT(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(25HSUN-SAT-S/C ANGLE (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
      CALL CHAROPT(0,0,0,0,0)
      DO 360 ITB=1,NSP
      REWIND 9
      IFR=0
  310 CONTINUE
      READ(9) T,SSV,SAF
      IF(EOF,9) 350,320
  320 IF(IFR.EQ.0) GO TO 330
      CALL LINE(TP,SSP,T,SSV(ITB))
  330 CONTINUE
      IF(IFR.NE.0) GO TO 340
      IFR=1
      CALL POINT(T,SSV(ITB))
      CALL SYMBOL(SYM(ITB))
  340 CONTINUE
```

Table 6-7 (cont)

```
      TP=T
      SSP=SSV(ITB)
      GO TO 310
  350 CONTINUE
      CALL POINT(TP,SSP)
      CALL SYMBOL(SYM(ITB))
      XHT=0.98-(ITB-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(ITB))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,ITB,IP)
      CALL SYMBOL(THING)
  360 CONTINUE
      CALL FRAME
      CALL MAPS(-TIM,TIM,-03.,07.,SDUNIT(1),SDUNIT(2),SDUNIT(3),
     XSDUNIT(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(28HSATELLITE HALF ANGLE (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
      CALL CHAROPT(0,0,0,0,0)
      DO 460 ITB=1,NSP
      REWIND 9
      IFR=0
  410 CONTINUE
      READ(9) T,SSV,SAF
      IF(EOF,9) 450,420
  420 IF(IFR.EQ.0) GO TO 430
      CALL LINE(TP,SSP,T,SAF(ITB))
      IF(IFR.NE.0) GO TO 440
  430 CONTINUE
      IFR=1
      CALL POINT(T,SAF(ITB))
      CALL SYMBOL(SYM(ITB))
  440 CONTINUE
      TP=T
      SSP=SAF(ITB)
      GO TO 410
  450 CONTINUE
      CALL POINT(TP,SSP)
      CALL SYMBOL(SYM(ITB))
      XHT=0.98-(ITB-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(ITB))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,ITB,IP)
      CALL SYMBOL(THING)
  460 CONTINUE
```

Table 6-7 (concl)

```
      IFR=0
      CALL FRAME
      CALL MAPS(-TIM,TIM,0.,360.,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT
     X(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,.05)
      CALL SYMBOL(60HTIME REQUIRED TO SWITCH TO SATELLITE AT RATE OF 1.
     XDEG/SEC$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
      CALL CHAROPT(0,0,0,0,0)
      DO 500  ITB=1,NSP
      REWIND 9
      IFR=0
465   CONTINUE
      READ(9) T,SSV,SAF,HANG2,DCNCLK
      IF(EOF,9) 485,470
470   IF(IFR.EQ.0) GO TO 475
      CALL LINE(TP,DCNCL1,T,DCNCLK(ITB))
      IF(IFR.NE.0) GO TO 480
475   CONTINUE
      IFR=1
      CALL POINT(T,DCNCLK(ITB))
      CALL SYMBOL(SYM(ITB))
480   CONTINUE
      TP=T
      DCNCL1=DCNCLK(ITB)
      GO TO 465
485   CONTINUE
      CALL POINT(TP,DCNCL1)
      CALL SYMBOL(SYM(ITB))
      XHT=0.98-(ITB-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(ITB))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,ITB,IP)
      CALL SYMBOL(THING)
500   CONTINUE
      IFR=0
      CALL FRAME
      RETURN
      END
```

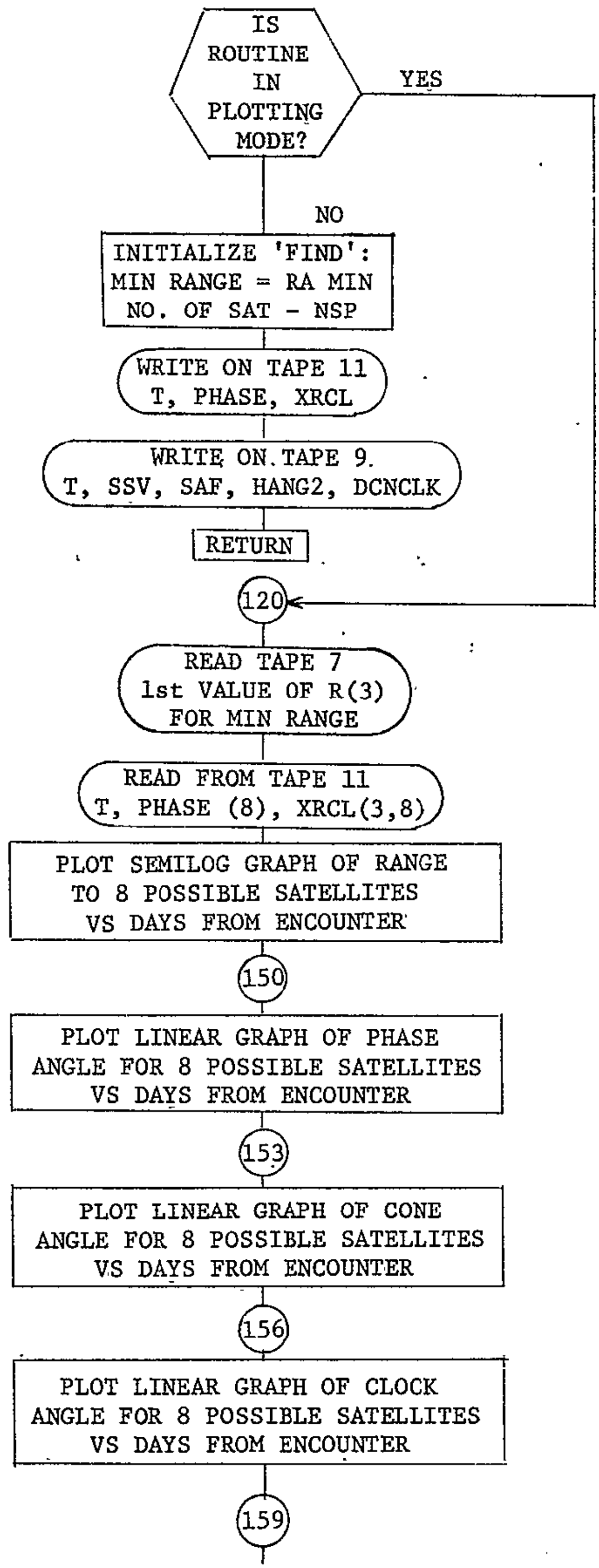


Fig. 6-7 OTJAZ Flow Diagram

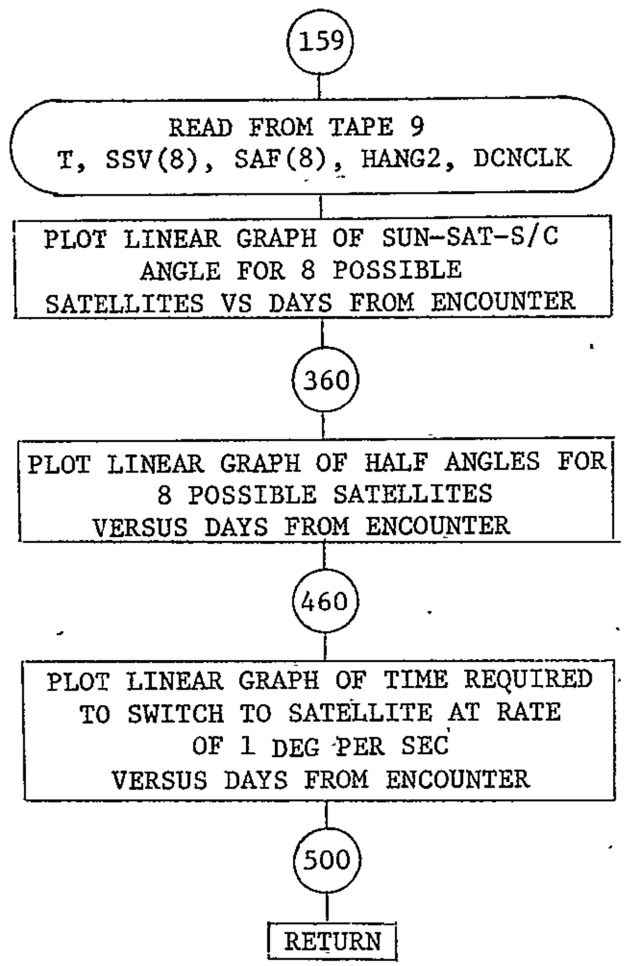

159
READ FROM TAPE 9
T, SSV(8), SAF(8), HANG2, DCNCLK
PLOT LINEAR GRAPH OF SUN-SAT-S/C
ANGLE FOR 8 POSSIBLE
SATELLITES VS DAYS FROM ENCOUNTER
360
PLOT LINEAR GRAPH OF HALF ANGLES FOR
8 POSSIBLE SATELLITES
VERSUS DAYS FROM ENCOUNTER
460
PLOT LINEAR GRAPH OF TIME REQUIRED
TO SWITCH TO SATELLITE AT RATE
OF 1 DEG PER SEC
VERSUS DAYS FROM ENCOUNTER
500
RETURN

Fig. 6-7 (concl)

6.3.8 OTPLT

Purpose: To plot planet (and specified satellite) range, phase angle, cone angle, and clock angle versus time.

Calling Sequence: CALL OTPLT

Input/Output:

I/O	FORTRAN Name	Definition
I	RSAT(3)	Range, cone & clock angles of satellite from S/C
I	PHASE	Planet phase angle
I	PHASES	Satellite phase angle
I	TIM	Maximum time range for plot (days)

Subprograms Required: ORBPOS, MAPSSL, MAPS, CHAROP, ABSBEM, SYMBOL, LINE, & ADV

Approximate Storage Required (octal): 1654

Discussion: None

Table 6-8 presents the OTPLT listing; Fig. 6-8 is the OTPLT flow diagram.

Table 6-8 OTPLT Listing

```
      SUBROUTINE OTPLT
      REALJOV
      DIMENSION X(3),XP(3),XSAT(3),XE(3),R(3),       RADP(3),TARGE(3),
     *TARGP(3) ,RSAT(3),        XSS(3),SDUNIT(4)   ,RDMIN(3)
      COMMON/HANDO/TIM,REC(9),REL(9),RES,SMR(9),SMRR(9),SMRS(9),TSMR(9)
     X,SPSMR,SPSMRS,SPTSMR,SPSMRR
      DATA SDUNIT/0.1,1.0,0.1,1.0/
      RPD=3.14159/180.
C
C   PLANET RANGE PLOT
C
      REWIND 7
      CALL ORBPOS(RDMIN,0.0)
      RAMIN=ABSV(RDMIN)
      READ(7)TX,X,XP,XSAT,XE,R
      RP=R(1)
      TP=TX/86400.
      CALL MAPSSL(-TIM,TIM,RAMIN,RP,SDUNIT(1),SDUNIT(2),SDUNIT(3),
     * SDUNIT(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(23HRANGE OF PLANET (KM) $.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
  100 CONTINUE
      READ(7)TX,X,XP,XSAT,XE,R
      IF(EOF,7)120,110
  110 T=TX/86400.
      CALL LINE(TP,RP,T,R(1))
      TP=T
      RP=R(1)
      GO TO 100
  120 CONTINUE
      REWIND 7
      CALL FRAME
C
C   PLANET PHASE ANGLE VS TIME PLOT
C
      CALL MAPS(-TIM,TIM,0.0,180.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT(
     X4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(26HPLANET PHASE ANGLE (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS)$. )
      READ(7) TX,X,XP,XSAT,XE,R,PHASE
      TP=TX/86400.
      PHP=PHASE/RPD
  121 CONTINUE
```

Table 6-8 (cont)

```
      READ(7) TX,X,XP,XSAT,XE,R,PHASE
      IF(EOF,7) 123,122
  122 CONTINUE
      PHASE=PHASE/RPD
      T=TX/86400.
      CALL LINE(TP,PHP,T,PHASE)
      TP=T
      PHP=PHASE
      GO TO 121
  123 CONTINUE
      REWIND 7
      CALL FRAME
C
C  PLANET CONE ANGLE VS TIME PLOT
C
      CALL MAPS(-TIM,TIM,0.0,180.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT(
     X4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(25HPLANET CONE ANGLE (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS)$. )
      READ(7) TX,X,XP,XSAT,XE,R,PHASE
      TP=TX/86400.
      RP=R(2)
  124 CONTINUE
      READ(7) TX,X,XP,XSAT,XE,R,PHASE
      IF(EOF,7) 126,125
  125 CONTINUE
      T=TX/86400.
      CALL LINE(TP,RP,T,R(2))
      TP=T
      RP=R(2)
      GO TO 124
  126 CONTINUE
      REWIND 7
      CALL FRAME
C
C  PLANET CLOCK ANGLE VS TIME PLOT
C
      CALL MAPS(-TIM,TIM,0.0,360.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT(
     X4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(26HPLANET CLOCK ANGLE (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS)$. )
      READ(7) TX,X,XP,XSAT,XE,R,PHASE
      TP=TX/86400.
      RP=R(3)
  127 CONTINUE
```

Table 6-8 (cont)

```
      READ(7) TX,X,XP,XSAT,XE,R,PHASE
      IF(EOF,7) 129,128
  128 CONTINUE
      T=TX/86400.
      CALL LINE(TP,RP,T,R(3))
      TP=T
      RP=R(3)
      GO TO 127
  129 CONTINUE
      REWIND 7
      CALL FRAME
C
C  SATELLITE RANGE VS TIME PLOT
C
      READ(7)TX,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT
      RP=RSAT(1)
      TP=TX/86400.
      CALL MAPSSL(-TIM,TIM,RAMIN,RP,SDUNIT(1),SDUNIT(2),SDUNIT(3),
     *SDUNIT(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(25HRANGE OF SATELLITE (KM)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
  130 CONTINUE
      READ(7)TX,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT
      IF(EOF,7)150,140
  140 CONTINUE
      T=TX/86400.
      CALL LINE(TP,RP,T,RSAT(1))
      TP=T
      RP=RSAT(1)
      GO TO 130
  150 CONTINUE
      REWIND 7
      CALL FRAME
C
C  SATELLITE PHASE ANGLE VS TIME PLOT
C
      CALL MAPS (-TIM,TIM,0.0,180.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT
     X(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(29HSATELLITE PHASE ANGLE (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS)$. )
      READ(7) TX,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT,PHASES
      TP=TX/86400.
      PHP=PHASES/RPD
  151 CONTINUE
```

Table 6-8 (cont)

```
      READ(7) TX,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT,PHASES
      IF(EOF,7) 153,152
  152 CONTINUE
      PHASES=PHASES/RPD
      T=TX/86400.
      CALL LINE(TP,PHP,T,PHASES)
      TP=T
      PHP=PHASES
      GO TO 151
  153 CONTINUE
      REWIND 7
      CALL FRAME
C
C  SATELLITE CONE ANGLE VS TIME PLOT
C
      CALL MAPS(-TIM,TIM,0.0,180.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT(
     X4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(28HSATELLITE CONE ANGLE (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
      READ(7) TX,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT,PHASES
      TP=T/86400.
      RP=RSAT(2)
  154 CONTINUE
      READ(7) TX,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT,PHASES
      IF(EOF,7) 156,155
  155 CONTINUE
      T=TX/86400.
      CALL LINE(TP,RP,T,RSAT(2))
      TP=T
      RP=RSAT(2)
      GO TO 154
  156 CONTINUE
      REWIND 7
      CALL FRAME
C
C  SATELLITE CLOCK ANGLE VS TIME PLOT
C
      CALL MAPS(-TIM,TIM,0.0,360.0,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNIT(
     X4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(29HSATELLITE CLOCK ANGLE (DEG)$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS)$. )
      READ(7) TX,X,XP,XSAT,XE,R,PHASE,RADP,TARGE,TARGP,RSAT,PHASES
      TP=TX/86400.
      RP=RSAT(3)
  157 CONTINUE
```

Table 6-8 (concl)

```
      READ(7) TX,X,XP,XSAT,XE,R,PHASE,RAJP,TARGE,TARGP,RSAT,PHASES
      IF(EOF,7) 159,158
  158 CONTINUE
      T=TX/86400.
      CALL LINE(TP,RP,T,RSAT(3))
      TP=T
      RP=RSAT(3)
      GO TO 157
  159 CONTINUE
      REWIND 7
      CALL FRAME
      RETURN
      END
```

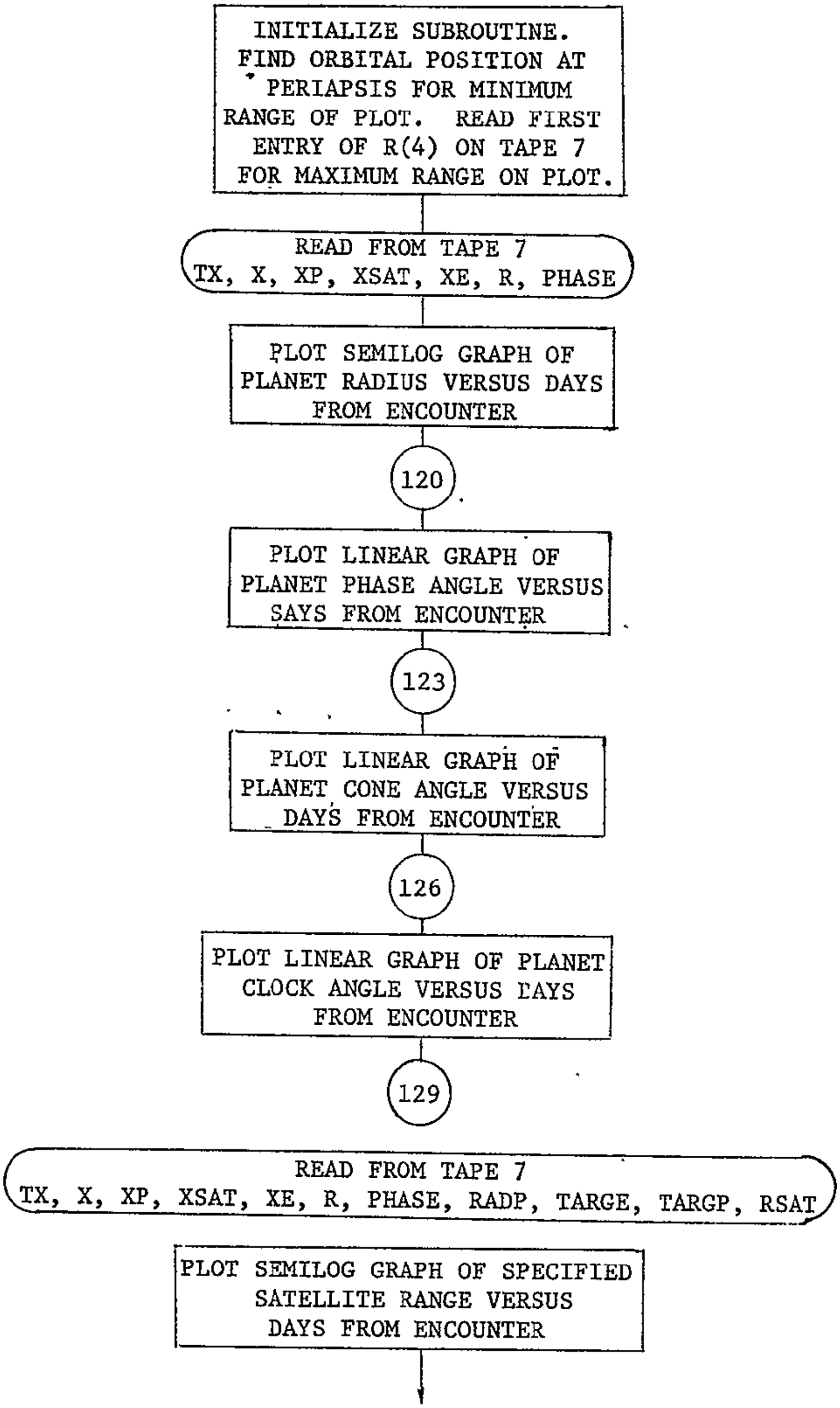


Fig. 6-8 OTPLT Flow Diagram

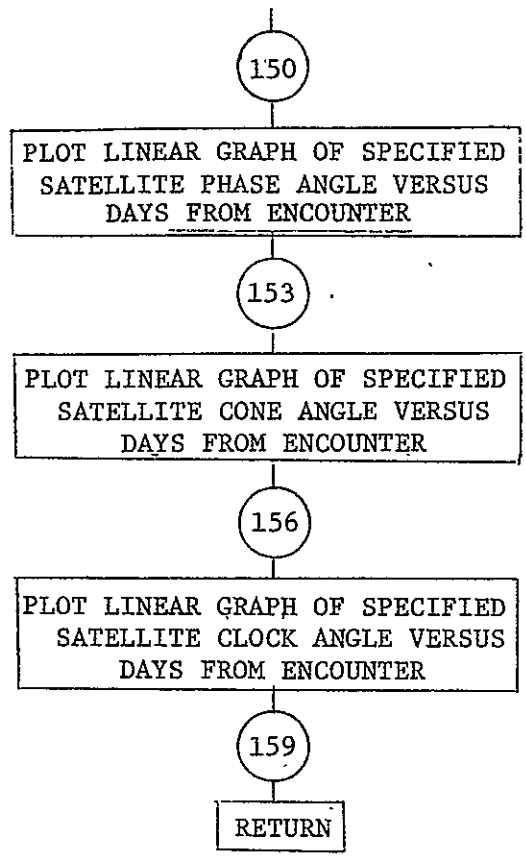

150
PLOT LINEAR GRAPH OF SPECIFIED SATELLITE PHASE ANGLE VERSUS DAYS FROM ENCOUNTER
153
PLOT LINEAR GRAPH OF SPECIFIED SATELLITE CONE ANGLE VERSUS DAYS FROM ENCOUNTER
156
PLOT LINEAR GRAPH OF SPECIFIED SATELLITE CLOCK ANGLE VERSUS DAYS FROM ENCOUNTER
159
RETURN

Fig. 6-8 (concl)

6.3.9 PIC2

Purpose: To plot the principal body (planet or satellite) complete with latitude parallels and lòngitude meridian lines showing the proper perspective in relationship to the distance from the viewing spacecraft and the proper angle of view of the TV camera. In the case of a planet, annotated position of visible satellites is also plotted.

Calling Sequence: CALL PIC2 (ANG2A, ANG2B)

Input/Output:

I/O	FORTRAN Name	Definition
I	ANG2A	Vertical camera field of view (deg)
I	ANG2B	Horizontal camera field of view (deg)
I	IPCN	Picture number
I	TITLE(13)	Mission title (78 characters)
I	IP	Planet index number
I	IVS	Satellite index number (selected)
I	IFLST	Run type flag = 1 - satellite run only = 0 - planet run
I	NCB	Principal body flag = 0 - on planet = 1 - switched to satellite
I	X(3)	Vector to S/C from principal body
I	RPL(12)	Radius of planet (km)
I	VW(3)	Viewpoint vector
I	PM(3)	Prime meridian vector
I	R	Range or planet from S/C (km)
O	TANR	Tangent of 1/2 angle of principal body
O	ASDP	half angle of principal body
O	SCALE	Plotting scale set by frame field of view of principal body half angle

Subprograms Required: TCONV, ARCSTP, PLASAT, PSCONS, VVIEW, VPROJ, SDINIT, LINE, CHAROP, ABSBEN, SYMBOL, NUMBER, SATS, DISK, SHADER, PGLOBE, PARLEL, MERIDN, VSCALE, VUNIT, ADV, VCROSS, ANGV, LOOK, LOCATE & FPLOT

Approximate Storage Required (octal): 665

Discussion: The subroutine plots the planet in the proper angular relationship to the outline of the frame. When the frame is smaller than the planet, the footprint of the frame is plotted on the planet surface.

Table 6-9 presents the PIC2 listing; Fig. 6-9 is the PIC2 flow diagram.

Table 6-9 PIC2 Listing

```
      SUBROUTINE PIC2(ANG2A,AND2B)
      COMMON/FOROUT/DUMMYB(13),IPCN,DUMMYC(4)
      COMMON/HEDING/TITLE(13)
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      COMMON/GEOM/XE(3),XP(3)
      DIMENSION ICLK(3)
      DIMENSION CLOCK(4),LABLE(8),SN(3)
      DIMENSION UX(3),UPV(3),UPM(3)
      COMMON/EXTRA/RPLP,ROT
      COMMON/VW/VWP(3),RIP,ASQP
      COMMON/POINT/VC(3,9),P(3,9)
      COMMON/FLYBY/DUM1(6),X(3),DUM2(6),SV(3)
      COMMON /PLCONS/GMS,GMPL(12),RS,RPL(12),SNAME(2),PLNAME(2,12),
     X SPV(3),PLPV(3,12),SVE(3),PLVE(3,12),SROT,PLROT(12),SOBL,PLOBL(12)
      COMMON /CVPLOT/VT(3),VR(3),VW(3),RPP,TANR,R2,SCALE,R(3),PEN
      COMMON/PCOORD/PV(3),VE(3),PM(3)
      DIMENSION QL(3)
      COMMON/PLTBUF/BUF(20)
      DIMENSION UCK(8)
      DIMENSION SCUNIT(4),COORD(4)
      DATA SCUNIT/0.1428,0.8572,0.1428,0.8572/
      DIMENSION QZ(3)
      COMMON/TRA/TRACE
      REAL LAT,LONG
      DIMENSION GOK(3)
      REAL JDV
      DIMENSION VQ(3,9)
      COMMON/JDAYS/JDV,TT
      DATA PI/3.1415926536/
      DATA QZ/7.0,-90.0,30.0/
      DATA QL/08.,-180.0,45.0/
      ANG2=AMAX1(ANG2A,ANG2B)
      NSAT=0
      CALL TCONV(CLOCK,5HCLOCK,ABS(TT),3HSEC)
  884 FORMAT(*TIME AFTER  ENCOUNTER = *I4,* DAYS *I2,* HRS *I2,* MIN*)
  883 FORMAT(*TIME BEFORE ENCOUNTER = *I4,* DAYS *I2,* HRS *I2,* MIN*)
      CALL ARCSTP(5.)
      TWIST=0.0
      IF(NCB.EQ.1) CALL PLASAT(SV,JDV,IVS,IP)
      RPP=RPL(IP)
      IF(NCB.EQ.1) CALL PSCONS(RPP,6HRADIUS,IVS,IP)
      ANG3=ANG2/2.0
      CALL VVIEW(0,0,0,0,IP)
      CALL VPROJ
      TANR=SQRT(RPP*RPP/(R2*R2-RPP*RPP))
      TANK=ATAN(TANR)*(180.0/PI)
      IF(TANK.GE.ANG3) ANG3=TANK
      SCALE=TAN(ANG3*PI/180.0)
      COORD(1)=-SCALE
      COORD(2)=SCALE
      COORD(3)=-SCALE
      COORD(4)=SCALE
```

Table 6-9 (concl)

```
      CALL SDINIT(BUF,20,COORD,SCUNIT)
      CALL LINE( SCALE, SCALE, SCALE,-SCALE)
      CALL LINE( SCALE,-SCALE,-SCALE,-SCALE)
      CALL LINE(-SCALE,-SCALE,-SCALE, SCALE)
      CALL LINE(-SCALE, SCALE, SCALE, SCALE)
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(17HPICTURE NUMBER $.)
      CALL NUMBER(IPCN,2HI8)
      CALL CHAROPT(0,0,0,0,0)
      CALL ABSBEAM(0.0,0.1)
      DO 100 IFL=1,6
  100 UCK(IFL)=TITLE(IFL)
      UCK(7)=2H$.
      CALL SYMBOL(UCK)
      CALL ABSBEAM(0.0,0.05)
      DO 110 IFL=7,13
  110 UCK(IFL-6)=TITLE(IFL)
      UCK(8)=2H$.
      CALL SYMBOL(UCK)
      IF(IFLST.NE.1.AND.NCB.NE.1) GO TO 1
      NSAT=IVS
1     CALL SATS(1,0,IP,NSAT,JDV,0)
      CALL DISK(IP,NSAT,0,0,0)
      CALL SHADER(50.0,SN)
      CALL PGLOBE(PM,JDV,IP)
      CALL PARLEL(PV,QZ)
      CALL MERIDN (PV,PM,QL)
      CALL VSCALE(SN,XP,-1.)
      IF(TT.LE.0.) ENCODE(49,883,LABLE)(CLOCK(J),J=1,3)
      IF(TT.GT.0.) ENCODE(49,884,LABLE)(CLOCK(J),J=1,3)
      CALL SCPRNA(BUF,890,300,20,90.,LABLE,5)
      IF(TANK.LT.ANG2/2.0) GO TO 10
      SAVE=TRACE
      ASDP=TANK
      CALL VUNIT(UX,X)
      CALL VUNIT(UPV,PV)
      CALL VUNIT(UPM,PM)
      RPLP=RPP
      ROT=PLROT(IP)
      DO 5 I=1,3
    5 VWP(I)=VW(I)
      RIP=R2
      ANG4A=ANG2A/2.0
      ANG4B=ANG2B/2.0
      CALL FRAME1(P,ANG4A,ANG4B)
      LAT=90.0-ANGV(UPV,UX,6HDEGREE)
      CALL VCROSS(GOK,UX,UPV)
      LONG=90.0+ANGV(GOK,UPM,6HDEGREE)
      CALL LOOK(DUM,CONE,CLOCK,LAT,LONG)
      CALL LOCATE(VQ,CONE,CLOCK,TWIST)
      CALL FPLOT(VQ)
      TRACE=SAVE
   10 CONTINUE
      CALL FRAME
      RETURN
      END
```

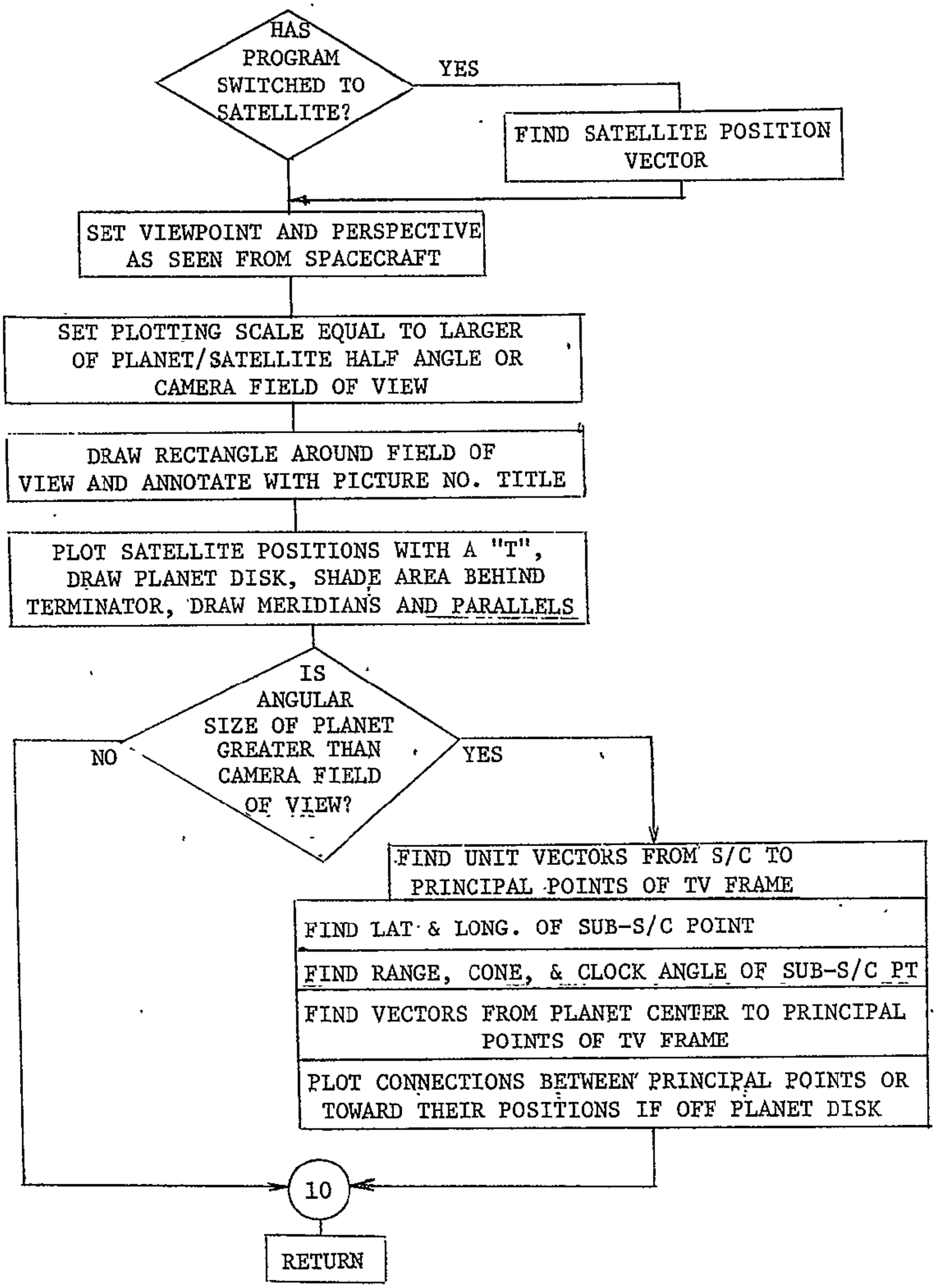


Fig. 6-9 PIC2 Flow Diagram

6.3.10 PICIN

Purpose: To determine if the time interval set for TV frames is compatible with system bit storage, transmission bit rates, and bits per picture, and adjust if necessary.

Calling sequence: CALL PICIN

Input/Output:

I/O	FORTRAN Name	Definition
I	ROTW (3)	Vehicle roll rate (radians/sec)
I/O	TFRAM/P(29)	Frame readout time (seconds)
I/O	TSTAR/P(31)	Start time for TV (days from periapsis)
I/O	TSTOP/P(37)	Stop time for TV (days from periapsis)
I/O	DTUSE/P(33)	User input picture taking interval (min)
I/O	BULK/P(34)	Bulk storage capacity (bits)
I/O	BULKO/P(35)	Storage capacity previously used (bits)
I/O	XBPS/P(36)	Bit rate transmitted (BPS)
I/O	SCIBPS/P(37)	Bit rate required by other instruments (BPS)
I/O	BUFCAP/P(38)	Buffer storage capacity (bits)
I/O	BPPIC/P(49)	Bits per picture (bits)
I/O	P(50)	Frame read rate (bits/sec)
O	STORD	Bulk Data Storage Capacity (picture)
O	TVOUT	Real time picture rate (bps)
O	DTPIC	Minimum picture internal (sec)
O	PICPHR	Maximum picture rate (pics/hr)
O	PICPDA	Maximum picture rate (pic/day)
O	NPIC	Maximum numer of pictures
O	ACTPIC	Actual picture rate (pic/day)
O	NPICA	Actual numberof pictures

Subprograms Required: None

Approximate Storage Required (octal): 562

Discussion: Other than the reset value of picture taking interval, three qualitative statements are output when appropriate:

1) Buffer capacity less than one picture;
2) Picture interval based on roll rate;
3) Roll rate equals zero.

Table 6-10 presents the PICIN listing; Fig. 6-10 is the PICIN flow diagram.

Table 6-10 PICIN Listing

```
      SUBROUTINE PICIN
C
C ROUTINE FOR COMPUTING MIN PICTURE TAKING INTERVAL FROM BIT RATE
C
C
C INPUT BULK   = BULK STORAGE CAPACITY, BITS
C INPUT BULKO  = BITS ALREADY IN BULK STORAGE
C INPUT DTUSE  = USER INPUT PICTURE TAKING INTERVAL, MIN   P(33)
C INPUT XBPS   = TRANSMIT BIT RATE, BPS
C INPUT SCIBPS = BIT RATE FROM OTHER INSTRUMENTS, BPS
C INPUT BUFCAP = BUFFER STORAGE CAPACITY, BITS
C INPUT BPPIC  = BITS PER PICTURE, P(49)
C INPUT TFRAM  = FRAME READOUT TIME, SEC, P(29)
C INPUT TSTAR  = PICTURE SEQUENCE START TIME, DAYS
C INPUT TSTOP  = PICTURE SEQUENCE STOP  TIME, DAYS
C INPUT RPM    = S/C SPIN RATE, ROTW(3) IN RAD/SEC
C
      COMMON/HEDING/TITLE(13)
      COMMON/REVER/ROTW(3)
      COMMON/CHARAC/P(60)
      EQUIVALENCE (P(29),TFRAM),(P(49),BPPIC)
     X,(P(34),BULK),(P(35),BULKO),(P(38),BUFCAP)
     X,(P(36),XBPS),(P(37),SCIBPS)
      DTUSE=P(33)*60.
      RPM=ROTW(3)*57.2957795131/6.
      TSTAR=P(31)
      TSTOP=P(32)
      WRITE(6,107)TITLE
      WRITE(6,104) TSTAR,TSTOP,DTUSE,TFRAM,P(50)
      DTTOT=(TSTOP-TSTAR)*86400.
      TSTAR=TSTAR*86400.
      TSTOP=TSTOP*86400.
      STORD=BULK/BPPIC
      WRITE(6,105)BULK,BPPIC,STORD,BULKO,BUFCAP
      TVOUT=XBPS-SCIBPS
      WRITE(6,108)XBPS,SCIBPS,TVOUT
      IF(BUFCAP-BPPIC)10,11,11
   10 BUFCAP=BPPIC
      WRITE(6,100)BUFCAP
   11 CONTINUE
      TBUFO=BUFCAP/TVOUT
      TBUFI=BUFCAP/P(50)
C
C TEST FOR TOPS OR PIONEER (BULK GREATER THAN 0 FOR TOPS)
C
      IF(BULK)1,1,2
C
C SPINNING S/C WITH BUFFER FOR ONE PICTURE (IE, PIONEER)
    1 DTPIC = TBUFO+ TFRAM
      IF(RPM)8,8,9
    8 WRITE(6,103)
      DTPIC=DTTOT
      GO TO 6
```

Table 6-10 (cont)

```
    9 IF(DTPIC-   60./RPM)3,3,7
    3 NROLL=1
      DTPIC=60./RPM
      WRITE(6,101)DTPIC,NROLL
      GO TO 4
    7 NROLL=DTPIC*RPM/60. +1
      DTPIC=NROLL*60./RPM
      WRITE(6,101)DTPIC,NROLL
      GO TO 4
C
C THREE AXIS STABLE S/C WITH MORE THAN ONE PICTURE STORAGE (IE, TOPS)
C
    2 DTPIC=BPPIC/TVOUT
C
C TEST IF USER INTERVAL LESS THAN MINIMUM
C
    4 CONTINUE
      IF(DTUSE-DTPIC)5,6,6
    5 DTUSE=DTPIC
      P(33)=DTUSE/60.
      WRITE(6,102) DTUSE
C
C MAXIMUM NO OF PICTURES/HOUR
    6 PICPHR=3600./DTPIC
      PICPDA=24.*PICPHR
C
C MAXIMUM NO OF PICTURES THAT CAN BE TAKEN
      NPIC=DTTOT/DTPIC
      ACTPIC=86400./DTUSE
      NPICA=DTTOT/DTUSE
      WRITE(6,106)DTPIC,PICPHR,PICPDA,NPIC,ACTPIC,NPICA
      RETURN
  100 FORMAT(5X,56HBUFFER CAPACITY LESS THAN ONE PICTURE, CAPACITY SET T
     *O    ,E10.3,/)
  101 FORMAT(5X,38HPICTURE INTERVAL BASED ON ROLL RATE = ,E15.8,4H SEC,
     *20H, NO OF ROLLS/PIC =   ,I4,/)
  102 FORMAT(5X,36HREQUESTED DELTA T LESS THAN MINIMUM /
     X       ,5X,32HDELTA T (SEC) HAS BEEN RESET TO ,E15.8)
  103 FORMAT(5X,21HROLL RATE EQUALS ZERO)
  104 FORMAT(    5X,48HCOMPUTATIONS FOR MINIMUM PICTURE TAKING INTERVAL
     X,//5X,28HPICTURE START TIME (DAYS) = ,E15.8/
     X,  5X,28HPICTURE STOP TIME  (DAYS) = ,E15.8/
     X,  5X,28HDELTA T REQUESTED  (SEC)  = ,E15.8/
     X,  5X,28HFRAME READ TIME    (SEC)  = ,E15.8/
     X,  5X,28HFRAME READ RATE    (BPS)  = ,E15.8//)
  105 FORMAT(5X,36HBULK DATA STORAGE CAPACITY (BITS) = ,E15.8/
     X         5X,36HPICTURE SIZE               (BITS) = ,E15.8/
     X         5X,36HBULK DATA STORAGE CAPACITY (PICS) = ,E15.8/
     X         5X,36HBULK CONTENTS IN USE       (BITS) = ,E15.8/
     X         5X,36HBUFFER STORAGE CAPACITY    (BITS) = ,E15.8//)
```

Table 6-10 (concl)

```
106 FORMAT(5X,36H MINIMUM PICTURE INTERVAL   (SEC)   = ,E15.8/
   *         5X,36HMAXIMUM PICTURE RATE  (PICS/HR)   = ,E15.8/
   *         5X,36HMAXIMUM PICTURE RATE  (PIC/DAY)   = ,E15.8/
   *         5X,36HMAXIMUM NUMBER OF PICTURES        = ,I6/
   *         5X,36HACTUAL  PICTURE RATE  (PIC/DAY)   = ,E15.8/
   *         5X,36HACTUAL  NUMBER OF PICTURES        = ,I6//)
107 FORMAT(1H1,4X,13A6,/)
108 FORMAT(5X,36HDATA TRANSMISSION RATE        (BPS) = ,E15.8/
   X         5X,36HRATE FROM OTHER INSTRUMENTS (BPS) = ,E15.8/
   X         5X,36HREAL TIME PICTURE RATE      (BPS) = ,E15.8//)
    END
```

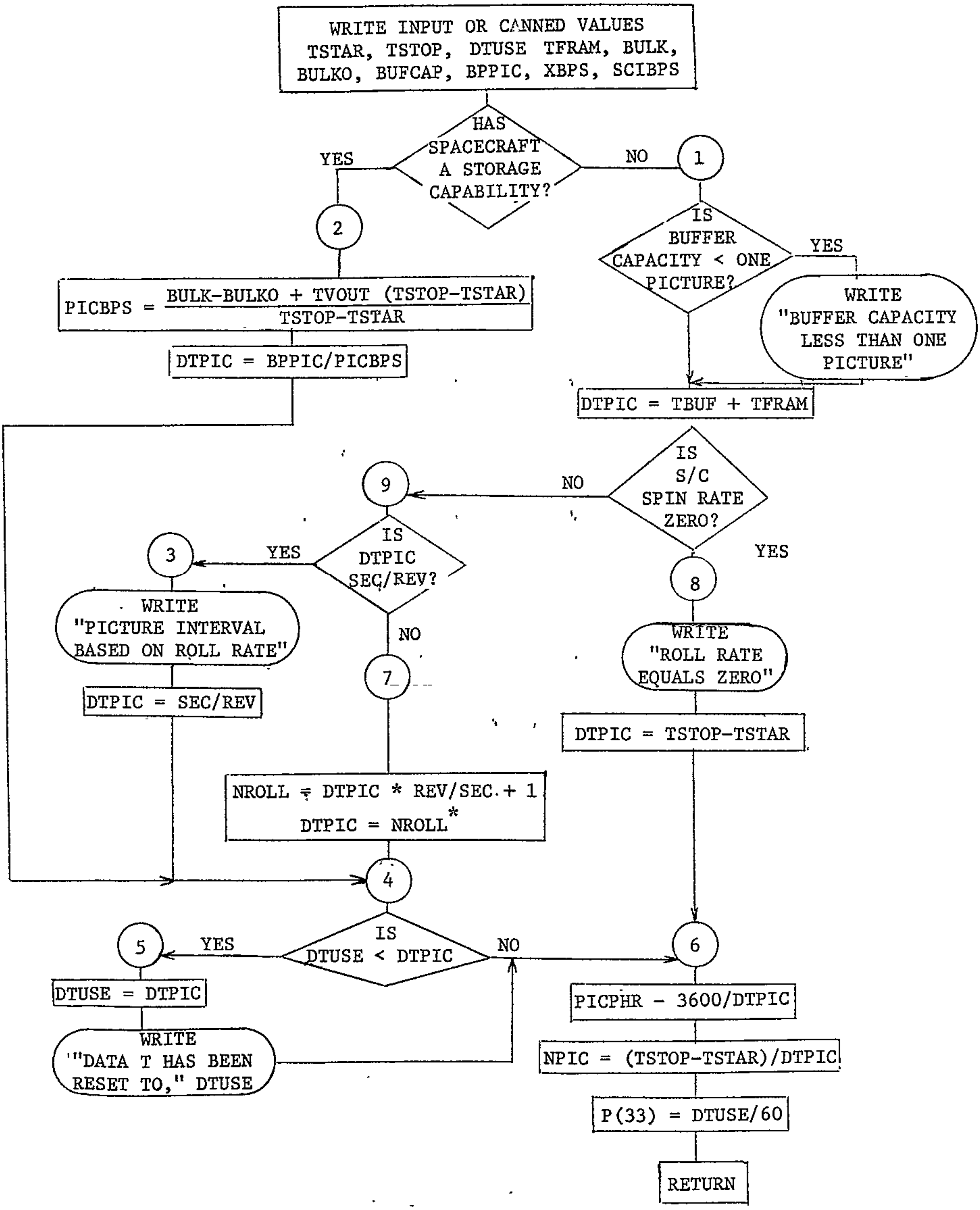


Fig. 6-10 PICIN Flow Diagram

6.3.11 PSATP

Purpose: To plot an overview of the mission, showing the trajectory of the spacecraft, the orbits of the planet's satellites, and vectors toward the sun, Earth, and antiperiapsis point.

Calling Sequence: CALL PSATP

Input/Output:

I/O	FORTRAN Name	Definition
I	IP	Planet index number
I	JDV	Julian date of encounter
O	VT(3)	Vector tangent to spacecraft trajectory at periapsis
O	VR(3)	Vector from planet to spacecraft at periapsis
O	VW(3) } GUSY(3) }	Vector toward trajectory pole

Subprograms required: ARCSTP, PLASAT, ABSV, MAP, VPROJ, ORBVEL, ORBPOS, UCROSS, VEQUAL, VVIEW, ARROCC, SATS, VPPLT, DISK, SINTRP, & ADV

Approximate Storage Required (octal): 315

Discussion: The primary output of this subroutine is the overview plot.

Table 6-11 presents the PSATP listing; Fig. 6-11 is the PSATP flow diagram.

Table 6-11 PSATP Listing

```
       SUBROUTINE PSATP
      COMMON/FLYBY/DUMMYA(3),GUSY(3),DUMMYB(12)
      COMMON /PLTBUF/BUF(20)
      COMMON /PLCONS/GMS,GMPL(12),RS,RPL(12),SNAME(2),PLNAME(2,12),
     X SPV(3),PLPV(3,12),SVE(3),PLVE(3,12),SROT,PLROT(12),SOBL,PLOBL(12)
      COMMON /CVPLOT/VT(3),VR(3),VW(3),RPP,TANR,R2,SCALE,R(3),PEN
       COMMON/PCOORD/PV(3),VE(3),PM(3)
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      COMMON/ORBI/AA,E,P,ORB,C1,TA,TEJ,VV(3),QV(3)
       COMMON/JDAYS/JDV,TT
      COMMON/PLAN/GM
       DIMENSION XM(3),X(3),XF(3),V(3),XS(3)
      COMMON/GEOM/XS,XM
       REAL JD,JDV
       EQUIVALENCE (JD,JDV)
      DIMENSION NS(9)
      DATA NS0/9/
      DATA NS/0,0,1,2,4,7,5,2,0/
       A=AA
      ECC=E
C      NUMBER OF SATELLITES
      RPP=RPL(IP)
       R2=0.
       S=5.
       CALL ARCSTP(S)
       CALL PLASAT(XM,JD,NS(IP),IP)
       SCALE=ABSV(XM)*1.1
      IF(NS(IP).EQ.0) SCALE=15.*RPL(IP)
       CALL MAP(-SCALE,SCALE,-SCALE,SCALE,.0,1.,0.,1.)
      BUF(4)=-SCALE
      BUF(5)=SCALE
      BUF(6)=-SCALE
      BUF(7)=SCALE
      BUF(8)=.0
      BUF(9)=1.
      BUF(10)=0.
      BUF(11)=1.
       CALL VPROJ
      CALL ORBVEL(VR,.0)
      CALL ORBPOS(VT,.0)
      CALL UCROSS(VW,VT,VR)
      CALL VEQUAL(GUSY,VW)
      CALL VVIEW(1,11, 1,15,0)
      DMDT=SQRT(GM/ABS(A)**3)
     CAT=(P/SCALE-1.)/ECC
      AT=ACOS(CAT)
      T=ANOMLY(-AE,AT)/DMDT
     N=ABS(T)/3600.
     JD=JDV
```

Table 6-11 (concl)

```
      N=3.*24.
      N=2*N
      CALL PLPOS(XS,JD,IP)
      CALL PLPOS(XM,JD,3)
      CALL VSCALE(XS,XS,-1.)
      CALL VSUM(XM,XM,XS)
      CALL ARROCC(3,32,5.,4)
      CALL ARROCC(0,64,5.,4)
      CALL SATS(N,0,IP,0,JD,3600.)
      N=N/2
      NN=-N
      DO 233 I=NN,N
      T=I*3600.
      CALL ORBPOS(XS,T)
233   CALL VPPLT(XS,1H*)
       CALL DISK(IP,0,0,0,0)
      CALL SINTRP(BUF)
 310   CALL FRAME
  30  RETURN
       END
```

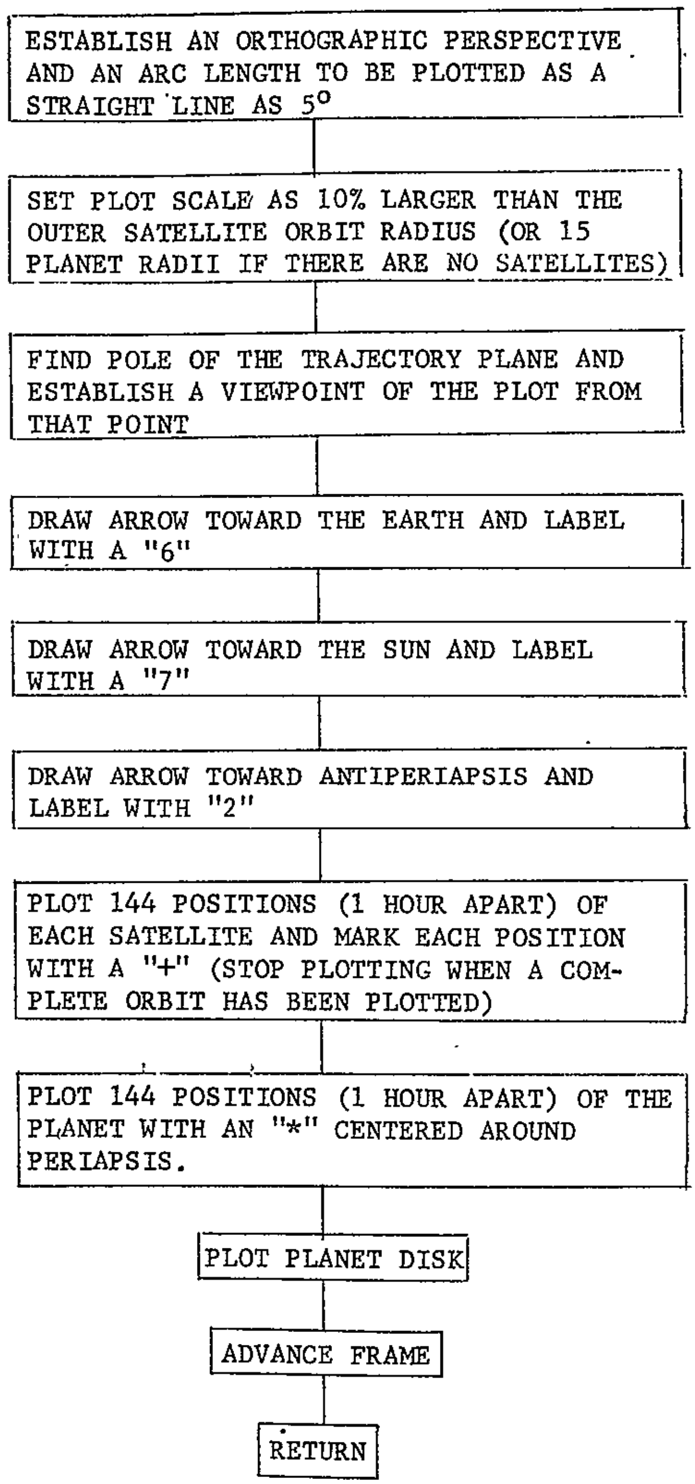


Fig. 6-11 PSATP Flow Diagram

6.3.12 RESO

Purpose: To calculate the resolution of a TV image, given the angular resolution of the lens and screen, and the range of the object. A second purpose is to plot the results versus time from periapsis.

Calling Sequence: CALL RESO (DIREC)

Input/Output:

I/O	FORTRAN Name	Definition
I	XSC(3)	Vector to spacecraft planet or satellite
I	XCL(3)	Vector to picture center from S/C (km)
I	RPL	Radius of planet or satellite (km)
I	DIREC	Program mode - plot results if DIREC='PLOT', calculate only if not
O	TIM	Largest time of interest from periapsis (days)
O	REC(9)	Resolution at the center of the body for planet and all satellites
O	REL(9)	Resolution at the limb of the body for planet and all satellites
O	RES	Resolution at picture center or principal body

Subprograms Required: PSCONS, ABSV, MAPSSL, CHAROP, ABSBEM, SYMBOL, LINE

Approximate Storage Required (octal): 1020

Discussion: None

Table 6-12 presents the RESO listing; Fig. 6-12 is the RESO flow diagram.

Table 6-12 RESO Listing

```
      SUBROUTINE RESO(DIREC)
      COMMON/HANDO/TIM,REC(9),REL(9),RES,SMR(9),SMRR(9),SMRS(9),TSMR(9)
     X,SPSMR,SPSMRS,SPTSMR,SPSMRR
      DIMENSION SDUNIT(4)
      COMMON/GEOM/XE(3),XP(3)
      COMMON/STUFF/XPL(3),VPLV(3),VPLS(3),XSA(3,8),VSAV(3,8),VSAS(3,8),
     XXCL(3),VCLV(3),VCLS(3)
      COMMON/JDAYS/JDV,TT
      COMMON/FLAGS/IFLT,IFLPL,IFLPR
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      COMMON /PLCONS/GMS,GMPL(12),RS,RPL(12),SNAME(2),PLNAME(2,12),
     X SPV(3),PLPV(3,12),SVE(3),PLVE(3,12),SROT,PLROT(12),SOBL,PLOBL(12)
      COMMON/CHARAC/P(60)
      COMMON/XZ/XZ(97)
      COMMON/TRA/TRACE
      REAL NAME(2)
      INTEGER CAMK
      DIMENSION PLACE(3)
      DIMENSION XTRA(3,9),XTRA1(3)
      DIMENSION NS(9)
      DIMENSION RPPS(9)
      DIMENSION THING(3)
      DATA THING(3)/2H$./
      DATA NS/0,0,1,2,4,7,5,2,0/
      DIMENSION SYM(8)
      DATA SYM/3H0$.,3HX$.,3H+$.,3HI$.,3HH$.,3HT$.,3HZ$.,3HP$./
      DATA PLACE/10H( CENTER ),10H(VER EDGE),10H(HOR EDGE) /
      DATA NAME(2)/10H (KM/PX)$.  /
      DATA SDUNIT/0.1,1.0,0.1,1.0/
      DATA PI,RPD/3.1415926535898,1.7453292519943209E-02/
      DATA IFR/0/
      PETE(RADIUS)=SQRT(RADIUS*RADIUS-RPP*RPP)
      IF(IFR.EQ.1) GO TO 60
      ITR1=10
      NSP=NS(IP)
      NXTR=NSP+1
      RPPS(1)=RPL(IP)
      DO 50 I=1,NSP
      CALL PSCONS(RPPS(I+1),6HRADIUS,I,IP)
   50 CONTINUE
   60 CONTINUE
      IF(DIREC.EQ.4HPLOT) GO TO 200
      DO 90 I=1,3
   90 XTRA(I,1)=XPL(I)
      DO 96 ITB=1,NSP
      DO 94 I=1,3
   94 XTRA(I,ITB+1)=XSA(I,ITB)
   96 CONTINUE
      DO 120 ITB=1,NXTR
      CALL VEQUAL(XTRA1,XTRA(1,ITB))
      RCEN=ABSV(XTRA1)
```

Table 6-12 (cont)

```
      IF(RCEN.LE.0.0) GO TO 120
      RPP=RPPS(ITB)
      RLIM=PETE(RCEN)
      RCEN=RCEN-RPP
      IF(IFR.EQ.0) RMAX=RLIM
      IF(IFR.EQ.0) RMIN=RCEN
      IFR=1
      RMAX1=AMAX1(RCEN,RLIM)
      RMIN1=AMIN1(RCEN,RLIM)
      IF(RMAX1.GT.RMAX) RMAX=RMAX1
      IF(RMIN1.LT.RMIN) RMIN=RMIN1
      I=1
      REC(ITB)=RCEN*P(45+I)
      REL(ITB)=RLIM*P(45+I)
  100 CONTINUE
  120 CONTINUE
      RSPE=ABSV(XCL)
      RES=RSPE*P(46)
      RETURN
  200 CONTINUE
      REWIND 7
      REWIND ITR1
      TIM=AMAX1(ABS(P(31)),ABS(P(32)))
      I=1
      RESMAX=RMAX*P(45+I)
      RESMIN=RMIN*P(45+I)
      DO 300 ITB=1,NXTR
      IF(ITB.GT.2) GO TO 204
      CALL FRAME
      CALL MAPSSL(-TIM,TIM,RESMIN,RESMAX,SDUNIT(1),SDUNIT(2),SDUNIT(3),
     $SDUNIT(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(12HRESOLUTION$.)
      CALL SYMBOL(3H $.)
      NAME(1)=PLACE(I)
      CALL SYMBOL(NAME)
      CALL SYMBOL(3H $.)
      IF(ITB.EQ.2) GO TO 202
      DO 201 J=1,2
  201 THING(J)=PLNAME(J,IP)
      GO TO 203
  202 THING(1)=10HSATELLITE
      THING(2)=2H$.
  203 CONTINUE
      CALL SYMBOL(THING)
      CALL ABSBEAM(0.2,0.01)
      CALL CHAROPT(0,0,1,0,0)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
  204 CONTINUE
      IFR=0
      REWIND ITR1
```

Table 6-12 (cont)

```
205 READ(ITR1)TT,REC,REL,RES,SMR,SMRR
    IF(EOF,ITR1)240,210
210 CONTINUE
    IF(IFR.EQ.0) GO TO 220
    IF(REC(ITB).LE.0.0.OR.REL(ITB).LE.0.0) GO TO 240
    CALL LINE(TTP,RECP,TT,REC(ITB))
    CALL LINE(TTP,RELP,TT,REL(ITB))
220 CONTINUE
    IF(ITB.EQ.1) GO TO 225
    IF(IFR.EQ.1) GO TO 225
    CALL CHAROPT(0,0,0,0,0)
    CALL POINT(TT,REC(ITB))
    CALL SYMBOL(SYM(ITB-1))
225 CONTINUE
    IFR=1
    TTP=TT
    RECP=REC(ITB)
    RELP=REL(ITB)
    GO TO 205
240 CONTINUE
    REWIND ITR1
    IF(ITB.EQ.1) GO TO 300
    CALL POINT(TTP,RECP)
    CALL SYMBOL(SYM(ITB-1))
    XHT=0.98-(ITB-2)*0.03
    CALL ABSBEAM(0.75,XHT)
    CALL SYMBOL(SYM(ITB-1))
    CALL SYMBOL(3H $.)
    CALL PSCONS(THING,4HNAME,ITB-1,IP)
    CALL SYMBOL(THING)
300 CONTINUE
    CALL FRAME
    IFR=0
    CALL MAPSSL(-TIM,TIM,RESMIN,RESMAX,SDUNIT(1),SDUNIT(2),SDUNIT(3),
   $SDUNIT(4))
    CALL CHAROPT(0,0,1,1,0)
    CALL ABSBEAM(0.05,0.2)
    CALL SYMBOL(12HRESOLUTION$.)
    CALL SYMBOL(3H $.)
    NAME(1)=PLACE(I)
    CALL SYMBOL(NAME)
    CALL SYMBOL(19H FOR PRIMARY BODY$.)
    CALL ABSBEAM(0.2,0.01)
    CALL CHAROPT(0,0,1,0,0)
    CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
305 READ(ITR1)TT,REC,REL,RES,SMR,SMRR
    IF(EOF,ITR1) 340,310
310 CONTINUE
    IF(IFR.EQ.0) GO TO 320
    CALL LINE(TTP,RESP,TT,RES)
320 CONTINUE
    IFR=1
    TTP=TT
```

Table 6-12 (concl)

```
      RESP=RES
      GO TO 305
  340 CONTINUE
      CALL FRAME
      REWIND ITR1
      IFR=0
      RETURN
      END
```

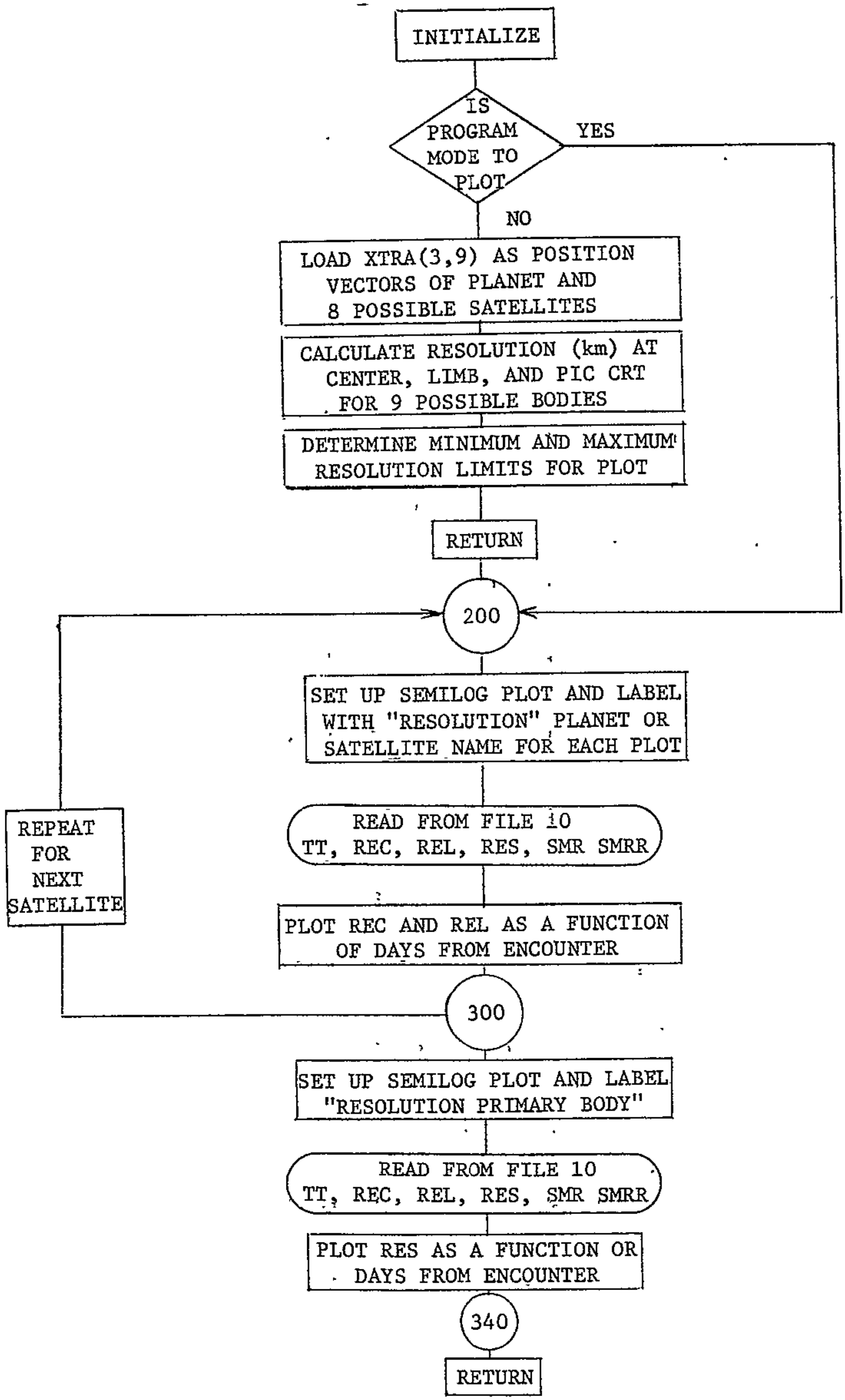


Fig. 6-12 RESO Flow Diagram

6.3.13 SMEAR

Purpose: To calculate smear of an image, given the velocity vector of the scene with respect to the imager, and to plot the result as a function of time from periapsis.

Calling Sequence: CALL SMEAR (DIREC)

Input/Output:

I/O	FORTRAN Name	Definition
I	DIREC	Program mode - plot if DIREC = "PLOT", calculate only if not
I	TT	Time from periapsis (sec)
O	TX	Time from periapsis (days)
I	TIM	Maximum time range for plotting
I	REC(9)	Resolution of sub-spacecraft point at picture center for planet and eight possible satellites (km/pixel)
I	REL(9)	Resolution of limb for planet and eight possible satellites (km/pixel)
I	RES	Resolution at center of primary body (km/pixel)
O	SMR(9)	Velocity of picture center due to orbital motion for planet and eight possible satellites (km/sec)
O	SMRS(9)	Velocity of picture center due to spacecraft rotation for planet and eight possible satellites (km/sec)
O	SMRR(9)	Smear of image for planet and eight possible satellites (pixel)
O	TSMR(9)	Exposure time to limit smear to less than ½ pixel (sec)
O	SPSMR	Velocity of picture center for primary body due to orbital motion (km/sec)
O	SPSMRS	Velocity of picture center for primary body due to spacecraft rotation (km/sec)
O	SPSMRR	Smear of image on primary body (pixels)
O	SPTSMR	Exposure time to reduce smear to ½ pixel on primary body (sec)

Subprograms Required: VSUM, MAPSSL, ABSBEM, CHAROP, SYMBOL, ADV, LINE

Approximate Storage Required (octal): 1760

Discussion: None

Table 6-13 presents the SMEAR listing; Fig. 6-13 is the SMEAR flow diagram.

Table 6-13 SMEAR Listing

```
      SUBROUTINE SMEAR(DIREC)
      DATA INP/0/
      DIMENSION NS(9)
      DATA NS/0,0,1,2,4,7,5,2,0/
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      DIMENSION DUMV(3),RUN(3),VIM(3)
      COMMON/HANDO/TIM,REC(9),REL(9),RES,SMR(9),SMRR(9),SMRS(9),TSMR(9)
     X,SPSMR,SPSMRS,SPTSMR,SPSMRR
      COMMON/CHARAC/P(60)
      DIMENSION SDUNIT(4)
      DATA SDUNIT/0.1,1.0,0.1,1.0/
      COMMON/JDAYS/JDV,TT
      COMMON/FLAGS/IFLT,IFLPL,IFLPR
      COMMON/STUFF/XPL(3),VPLV(3),VPLS(3),XSA(3,8),VSAV(3,8),VSAS(3,8),
     XXCL(3),VCLV(3),VCLS(3)
      DIMENSION THING(3)
      DATA THING(3)/2H$./
      DIMENSION SYM(8)
      DATA SYM/3H0$.,3HX$.,3H+$.,3HI$.,3HH$.,3HT$.,3HZ$.,3HP$./
      IF(DIREC.EQ.4HPLOT) GO TO 200
      ITR2=10
      IF(INP.NE.0) GO TO 90
      NST=NS(IP)
   90 CONTINUE
      SMR(1)=SQRT(VPLV(1)*VPLV(1)+VPLV(2)*VPLV(2))
      SMRS(1)=SQRT(VPLS(1)*VPLS(1)+VPLS(2)*VPLS(2))
      CALL VSUM(VIM,VPLS,VPLV)
      SMRT=SQRT(VIM(1)*VIM(1)+VIM(2)*VIM(2))
      TSMR(1)=REC(1)/(SMRT*2.0)
      SMRR(1)=(SMRT/REC(1))*P(26)
      DO 100I=1,NST
      IF(REC(I+1).LE.0.0) GO TO 100
      SMR(I+1)=SQRT(VSAV(1,I)*VSAV(1,I)+VSAV(2,I)*VSAV(2,I))
      SMRS(I+1)=SQRT(VSAS(1,I)*VSAS(1,I)+VSAS(2,I)*VSAS(2,I))
      CALL VSUM(VIM,VSAS(1,I),VSAV(1,I))
      SMRT=SQRT(VIM(1)*VIM(1)+VIM(2)*VIM(2))
      TSMR(I+1)=REC(I+1)/(SMRT*2.0)
      SMRR(I+1)=(SMRT/REC(I+1))*P(26)
  100 CONTINUE
      TX=TT/86400.
      SPSMR     =SQRT(VCLV(1)*VCLV(1)+VC_V(2)*VCLV(2))
      SPSMRS     =SQRT(VCLS(1)*VCLS(1)+VCLS(2)*VCLS(2))
      CALL VSUM(VIM,VCLS,VCLV)
      SMRT=SQRT(VIM(1)*VIM(1)+VIM(2)*VIM(2))
      SPTSMR     =RES/(SMRT*2.0)
      SPSMRR     =(SMRT/RES)*P(26)
      WRITE(ITR2) TX,REC,REL,RES,SMR,SMRR,SMRS
     X,SPSMR,SPSMRS,SPSMRR
      IF(INP.GT.0) GO TO 120
      INP=INP+1
      SMRN=SPSMR
      SMRX=SPSMR
```

Table 6-13 (cont)

```
      SMRSN=SPSMRS
      SMRSX=SPSMRS
      SMRRN=SPSMRR
      SMRRX=SPSMRR
      NSX=NST+1
  120 CONTINUE
      DO 130 I=1,NSX
      IF(SMR(I).LE.0.0) GO TO 130
      SMRN=AMIN1(SMRN,SPSMR,SMR(I))
      SMRX=AMAX1(SMRX,SPSMR,SMR(I))
      SMRSN=AMIN1(SMRSN,SPSMRS,SMRS(I))
      SMRSX=AMAX1(SMRSX,SPSMRS,SMRS(I))
      SMRRN=AMIN1(SMRRN,SPSMRR,SMRR(I))
      SMRRX=AMAX1(SMRRX,SPSMRR,SMRR(I))
  130 CONTINUE
      RETURN
  200 CONTINUE
      REWIND ITR2
      CALL MAPSSL(-TIM,TIM,SMRN,SMRX,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNI
     $T(4))
      CALL ABSBEAM(0.05,0.2)
      CALL CHAROPT(0,0,1,1,0)
      CALL SYMBOL(27HSMEAR DUE TO VEL (KM/SEC)$.)
      CALL SYMBOL(24H AT SPECIFIED LOCATION$.)
      CALL ABSBEAM(0.2,0.01)
      CALL CHAROPT(0,0,1,0,0)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
      IFR=0
      REWIND ITR2
  210 CONTINUE
      READ(ITR2) TT,REC,REL,RES,SMR,SMRR,SMRS
     X,SPSMR,SPSMRS,SPSMRR
      IF(EOF,ITR2) 230,220
  220 CONTINUE
      IF(IFR.EQ.0) GO TO 225
      CALL LINE(TTP,SMRP,TT,SPSMR)
  225 CONTINUE
      IFR=1
      TTP=TT
      SMRP=SPSMR
      GO TO 210
  230 CONTINUE
      REWIND ITR2
      CALL FRAME
      IFR=0
      CALL MAPSSL(-TIM,TIM,SMRSN,SMRSX,SDUNIT(1),SDUNIT(2),SDUNIT(3),
     *SDUNIT(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(27HSMEAR DUE TO S/C (KM/SEC)$.)
      CALL SYMBOL(24H AT SPECIFIED LOCATION$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
```

Table 6-13 (cont)

```
270 CONTINUE
    READ(ITR2) TT,REC,REL,RES,SMR,SMRR,SMRS
   X,SPSMR,SPSMRS,SPSMRR
    IF(EOF,ITR2)290,280
280 CONTINUE
    IF(IFR.EQ.0)GO TO 285
    CALL LINE(TTP,SMRSP,TT,SPSMRS)
285 CONTINUE
    IFR=1
    TTP=TT
    SMRSP=SPSMRS
    GO TO270
290 CONTINUE
    IFR=0
    CALL FRAME
    REWIND ITR2
    CALL MAPSSL(-TIM,TIM,SMRRN,SMRRX,SDUNIT(1),SDUNIT(2),SDUNIT(3),
   XSDUNIT(4))
    CALL ABSBEAM(0.05,0.2)
    CALL CHAROPT(0,0,1,1,0)
    CALL SYMBOL(26HNUMBER OF PIXELS SMEARED$.)
    CALL SYMBOL(24H AT SPECIFIED LOCATION$.)
    CALL ABSBEAM(0.2,0.01)
    CALL CHAROPT(0,0,1,0,0)
    CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
240 CONTINUE
    READ(ITR2) TT,REC,REL,RES,SMR,SMRR,SMRS
   X,SPSMR,SPSMRS,SPSMRR
    IF(EOF,ITR2) 260,250
250 CONTINUE
    IF(IFR.EQ.0) GO TO 255
    CALL LINE(TTP,SMRRP,TT,SPSMRR)
255 CONTINUE
    IFR=1
    TTP=TT
    SMRRP=SPSMRR
    GO TO 240
260 CONTINUE
    CALL FRAME
    CALL MAPSSL(-TIM,TIM,SMRN,SMRX,SDUNIT(1),SDUNIT(2),SDUNIT(3),SDUNI
   $T(4))
    CALL ABSBEAM(0.05,0.2)
    CALL CHAROPT(0,0,1,1,0)
    CALL SYMBOL(27HSMEAR DUE TO VEL (KM/SEC)$.)
    CALL SYMBOL(3H $.)
    CALL SYMBOL(11HSATELLITE$.)
    CALL CHAROPT(0,0,1,0,0)
    CALL ABSBEAM(0.2,0.01)
    CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
    CALL CHAROPT(0,0,0,0,0)
    DO 330 I=1,NST
    REWIND ITR2
```

Table 6-13 (cont)

```
      IFR=0
 310  CONTINUE
      READ(ITR2) TT,REC,REL,RES,SMR,SMRR,SMRS
     X,SPSMR,SPSMRS,SPSMRR
      IF(EOF,ITR2) 329,320
 320  CONTINUE
      IF(IFR.EQ.0) GO TO 325
      IF(SMRP.LE.0.0) GO TO 330
      CALL LINE(TTP,SMRP,TT,SMR(I+1))
 325  CONTINUE
      IF(IFR.NE.0) GO TO 328
      IF(SMR(I+1).LE.0.0) GO TO 330
      CALL POINT(TT,SMR(I+1))
      CALL SYMBOL(SYM(I))
      IFR=1
 328  CONTINUE
      TTP=TT
      SMRP=SMR(I+1)
      GO TO 310
 329  CONTINUE
      CALL POINT(TTP,SMRP)
      CALL SYMBOL(SYM(I))
      XHT=0.98-(I-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(I))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,I,IP)
      CALL SYMBOL(THING)
 330  CONTINUE
      CALL FRAME
      CALL MAPSSL(-TIM,TIM,SMRSN,SMRSX,SDUNIT(1),SDUNIT(2),SDUNIT(3),
     *SDUNIT(4))
      CALL CHAROPT(0,0,1,1,0)
      CALL ABSBEAM(0.05,0.2)
      CALL SYMBOL(27HSMEAR DUE TO S/C (KM/SEC)$.)
      CALL SYMBOL(3H $.)
      CALL SYMBOL(11HSATELLITE$.)
      CALL CHAROPT(0,0,1,0,0)
      CALL ABSBEAM(0.2,0.01)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
      CALL CHAROPT(0,0,0,0,0)
      DO 390 I=1,NST
      REWIND ITR2
      IFR=0
 370  CONTINUE
      READ(ITR2) TT,REC,REL,RES,SMR,SMRR,SMRS
     X,SPSMR,SPSMRS,SPSMRR
      IF(EOF,ITR2) 388,380
 380  CONTINUE
      IF(IFR.EQ.0)GO TO 385
      IF(SMRSP.LE.0.0) GO TO 390
      CALL LINE(TTP,SMRSP,TT,SMRS(I+1))
 385  CONTINUE
      IF(IFR.NE.0) GO TO 386
      IF(SMRS(I+1).LE.0.0) GO TO 390
```

Table 6-13 (cont)

```
      CALL POINT(TT,SMRS(I+1))
      CALL SYMBOL(SYM(I))
      IFR=1
  386 CONTINUE
      TTP=TT
      SMRSP=SMRS(I+1)
      GO TO 370
  388 CONTINUE
      CALL POINT(TTP,SMRSP)
      CALL SYMBOL(SYM(I))
      XHT=0.98-(I-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(I))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,I,IP)
      CALL SYMBOL(THING)
  390 CONTINUE
      CALL FRAME
      CALL MAPSSL(-TIM,TIM,SMRRN,SMRRX,SDUNIT(1),SDUNIT(2),SDUNIT(3),
     XSDUNIT(4))
      CALL ABSBEAM(0.05,0.2)
      CALL CHAROPT(0,0,1,1,0)
      CALL SYMBOL(26HNUMBER OF PIXELS SMEARED$.)
      CALL SYMBOL(3H $.)
      CALL SYMBOL(11HSATELLITE$.)
      CALL ABSBEAM(0.2,0.01)
      CALL CHAROPT(0,0,1,0,0)
      CALL SYMBOL(29HTIME FROM ENCOUNTER (DAYS) $.)
      CALL CHAROPT(0,0,0,0,0)
      DO 360 I=1,NST
      IFR=0
      REWIND ITR2
  340 CONTINUE
      READ(ITR2) TT,REC,REL,RES,SMR,SMRR,SMRS
     X,SPSMR,SPSMRS,SPSMRR
      IF(EOF,ITR2) 358,350
  350 CONTINUE
      IF(SMRR(I+1).LE.0) GO TO 360
      IF(IFR.EQ.0) GO TO 355
      IF(SMRRP.LE.0.0) GO TO 360
      CALL LINE(TTP,SMRRP,TT,SMRR(I+1))
  355 CONTINUE
      IF(IFR.NE.0) GO TO 356
      IF(SMRR(I+1).LE.0) GO TO 360
      CALL POINT(TT,SMRR(I+1))
      CALL SYMBOL(SYM(I))
      IFR=1
  356 CONTINUE
      TTP=TT
      SMRRP=SMRR(I+1)
      GO TO 340
  358 CONTINUE
```

Table 6-13 (concl)

```
      CALL POINT(TTP,SMRRP)
      CALL SYMBOL(SYM(I))
      XHT=0.98-(I-1)*0.03
      CALL ABSBEAM(0.75,XHT)
      CALL SYMBOL(SYM(I))
      CALL SYMBOL(3H $.)
      CALL PSCONS(THING,4HNAME,I,IP)
      CALL SYMBOL(THING)
  360 CONTINUE
      REWIND ITR2
      CALL FRAME
      IFR=0
      INP=0
      RETURN
      END
```

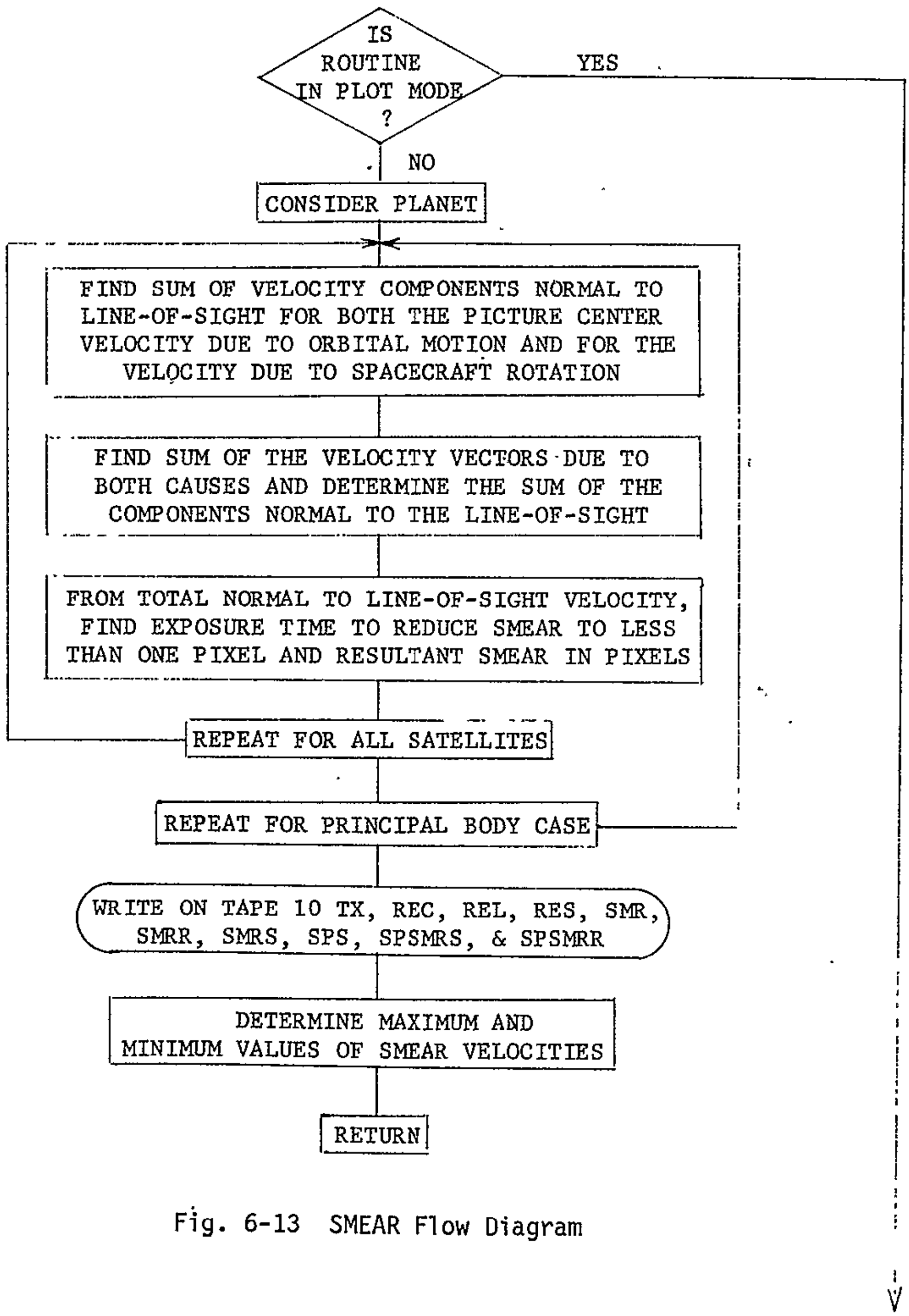


Fig. 6-13 SMEAR Flow Diagram

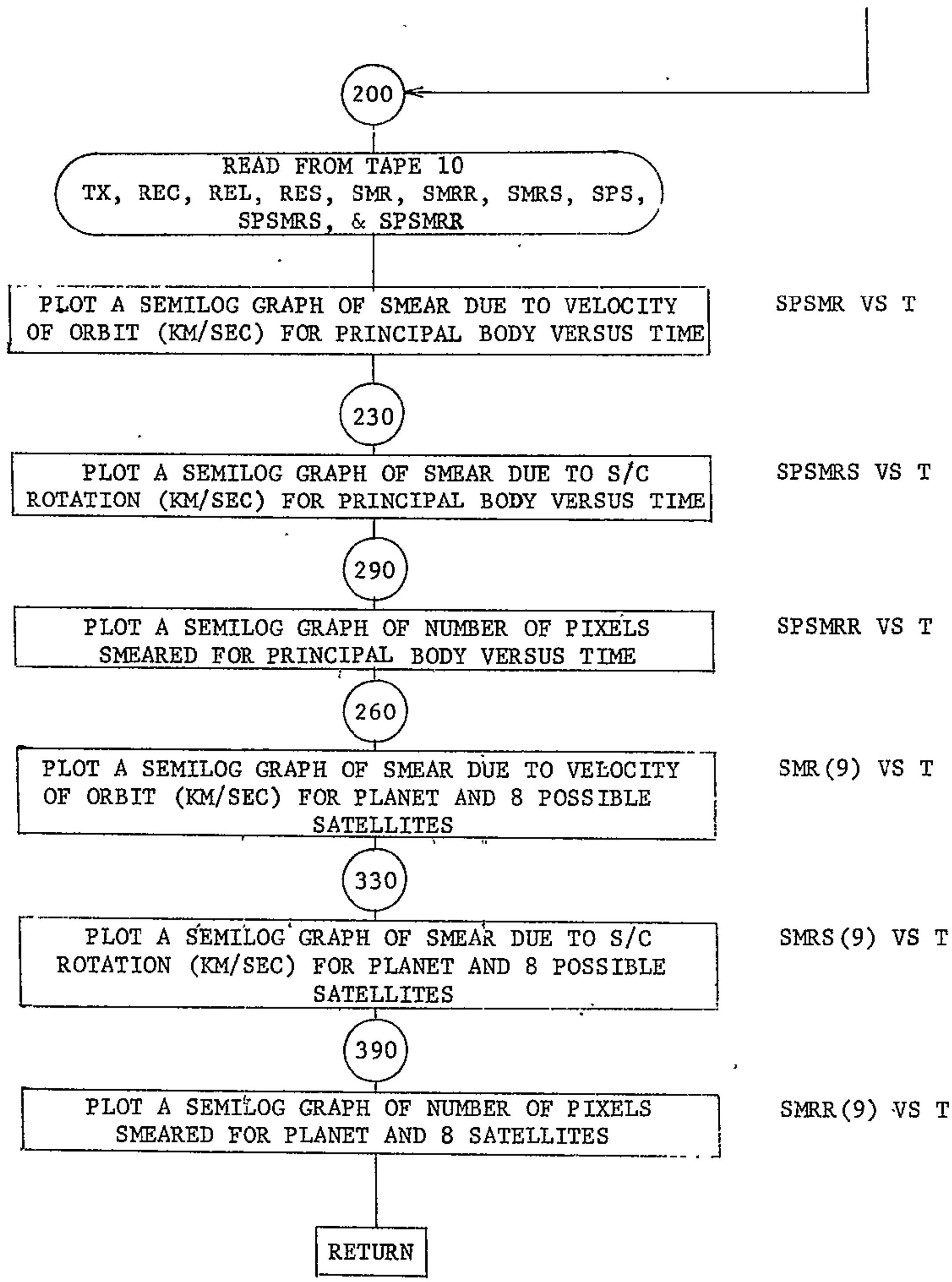


Fig. 6-13 (concl)

6.3.14 TCON

Purpose: To generate time of measurements and to determine when resulting information should be plotted or printed.

Calling Sequence: CALL TCON (NP)

Input/Output:

I/O	FORTRAN Name	Definition
I	P (31)	Start time from periapsis (days)
I	P (32)	Stop time from periapsis (days)
I	P (39)	Measurements per calculation step
I	P (40)	Minutes/measurement
I	XZ (81)	Plot interval
I	XZ (82)	Print interval
O	TCONS (200)	Time for measurement (days from periapsis)
O	NP	Total number of measurement times

Subprograms Required: None

Approximate Storage Required (octal): 316

Discussion: When measurement time and cone and clock angles are read in as inputs, both TCONS and NP are inputs and the subroutine performs no function.

Table 6-14 presents the TCON listing; Fig. 6-14 is the TCON flow diagram.

4

Table 6-14 TCON Listing

```
      SUBROUTINE TCON(NP)
      COMMON/HEDING/TITLE(13)
      COMMON/CHARAC/P(60)
      COMMON/FLAGS/IFLT,IFLPL,IFLPR
      COMMON/TCONS/TCONS(200)
      COMMON/XZ/XZ(97)
      COMMON/TEST/ITEST(97)
C
C P(31)=START TIME FROM PERIAPSIS IN DAYS
C P(32)= STOP TIME FROM PERIAPSIS IN DAYS
C P(39)=PICTURES PER CALC STEP
C P(40)=MINUTES/PICTURE
      TCALC=P(39)*P(40)
      DELT=TCALC/1440.
      START=ABS(P(31))
      STOP=P(32)
C NP=NUMBER OF CALC STEPS
C
CIFLPL=CALC STEPS PER PLOT
      IFLPL=ABS(P(31))/DELT + 1
      IF(ITEST(81).EQ.1) IFLPL=XZ(81)
      DTPPL=IFLPL*DELT*1440.
C
CIFLPR=CALC STEPS PER PRNT
      IFLPR=1
      IF(ITEST(82).EQ.1) IFLPR=XZ(82)
      DTPPR=IFLPR*DELT*1440.
C
      WRITE(6,1001)TITLE
 1001 FORMAT(1H1,13A6,/)
      WRITE(6,1020) IFLPL,DTPPL,IFLPR,DTPPR
      WRITE(6,1000)TCALC
C
      IF(IFLT.EQ.1) GO TO 120
      IF(ITEST(69).NE.1) GO TO 5
      NP=1+IFIX((START+STOP)/DELT)
      IF(NP.GT.200)NP=200
      GO TO 100
    5 CONTINUE
      NP=39
      DO 10 I=1,NP
      S1=FLOAT(I-21)/20.
      S2=FLOAT(I-21)
   10 TCONS(I)=START*ABS(S1)*S1*S1*SIGN(1.0,S2)
      GO TO 120
  100 CONTINUE
      DO 110 I=1,NP
      TCONS(I)=FLOAT(I-1)*DELT+P(31)
  110 CONTINUE
  120 CONTINUE
      WRITE(6,1030)
```

Table 6-14 (concl)

```
      DO 130 I=1,NP
      NPIC=(I-1)*P(39)+1
      TH=TCONS(I)*24.
      TM=TH*60.
  130 WRITE(6,1010)NPIC,I,TCONS(I),TH,TM
      RETURN
 1000 FORMAT(/5X,33HTIME INTERVAL FOR CALCULATIONS = ,F7.2,8H MINUTES,/)
 1010 FORMAT(5X,I4,2X,I4,3E16.7)
 1020 FORMAT(     5X,16HPLOT INTERVAL = ,I2,14H CALC STEPS = ,F7.2,7HMINU
     $TES,/,5X,16HPRNT INTERVAL = ,I2,13HCALC STEPS =   ,F7.2,
     $8H MINUTES,/)
 1030 FORMAT(3X,56HPICTURE CALC   TIME (DAYS)     TIME (HRS)     TIME (M
     $IN))
      END
```

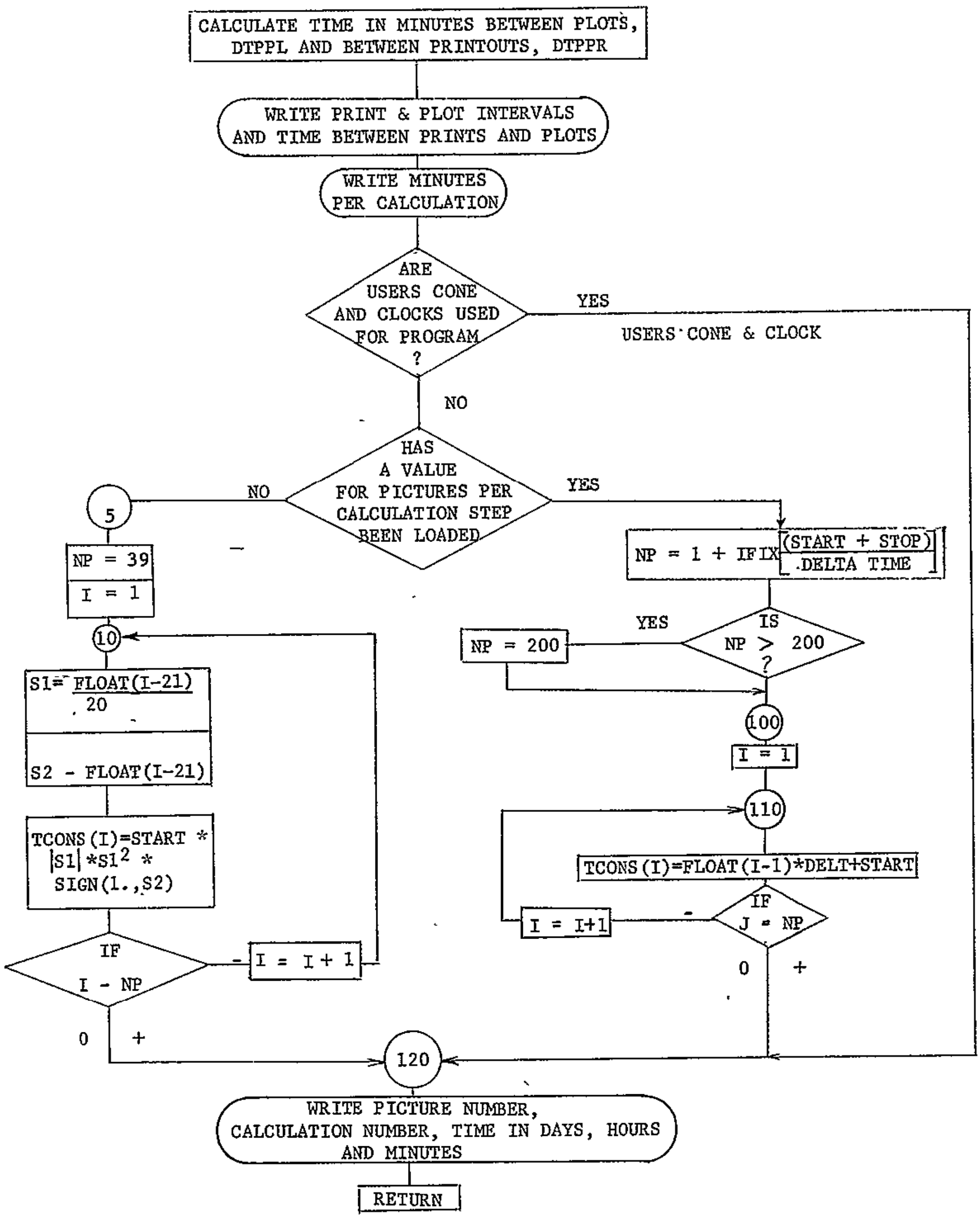


Fig. 6-14 TCON Flow Diagram

6.3.15 TVCHAR

Purpose: To provide a meaning to the block of input variables assigned to the instrument. Whenever specific information is not supplied by data inputs, canned values describing six alternate cameras are provided. A second block of TV variables are computed by the subroutine and the total set of variables is printed out.

Calling Sequence: CALL TVCHAR (INST)

Input/Output:

I/O	FORTRAN Name	Definition
I	XZ (97)	Input variable array XZ (31) thru XZ (90) instrument characteristics
O	P (60)	Output TV characteristics array, which is subdivided as follows: P (1) - P (5) TV' name P (6) - P (7), P(15) - P(16), P(26) - P(28) } Camera type peculiar P (8) - P(14), P(17) - P(25), P(29) - P(39) } Canned TV characteristics not related to specific camera type P(40) - P(52) Values computed from other inputs P(53) - P(60) Spare

Subprograms Required: FOLDER, PICIN

Approximate Storage Required (octal): 1104

Discussion: See subroutine listing (Table 6-15) for meaning of the members of the TV characteristics array.

Figure 6-15 is the TVCHAR flow diagram.

Table 6-15 TVCHAR Listing

```
      SUBROUTINE TVCHAR(INST
      DIMENSION TVAR(55)
      COMMON/HEDING/TITLE(13)
      COMMON/TNME/TVNAME(5)
      COMMON/XZ/XZ(97)
      COMMON/CHARAC/P(60)
      COMMON/TEST/ITEST(97)
      DATA PI/3.14159/
      DATA TVAR/ 6HFOC LN,6HDIAM  ,6HOBSCUR,6HTRANS ,6HSIZE-V,6HSIZE-H,
     X  6HLINES ,6HPIXELS,6HBIT/PX,6HMINEXP,6HMAXEXP,6HRESO-1,6HRESO-2,
     X  6HRESO-3,6HRESO-4,6HSENSOR,6HB FILT,6HG FILT,6HR FILT,6HP FILT,
     X  6HDT EXP,6H OPEN ,6H OPEN ,6HT READ,6HERASET,6HTSTART,6HTSTOP ,
     X  6HDEL T ,6HBULK  ,6HBULK-0,6HBITRAT,6HSCIBPS,6HBUFFER,6HPIC/ST,
     X  6HDEL TF,6HF/STOP,6HA BLOC,6HDIF LM,6HFOV-V ,6HFOV-H ,6HRES-C ,
     X  6HRES-V ,6HRES-H ,6HBIT/PC,6HRR-BPS,6HT/NUM ,6HT/NUMF,6H OPEN ,
     X  6H OPEN ,6H OPEN ,6H OPEN ,6H OPEN ,6H OPEN ,6H OPEN ,6H OPEN /
C
C INSTR FROM INPUT 22=INST=IJK, I SPECIFIES INSTRUMENT TYPE, I=1 FOR TV
C                             J SPECIFIES WHICH TV IN SUBROUTINE
C INSTR = INST-100 = JK       K IS UNUSED
C IF J=0, J IS SET TO 1
C USER MAY CHANGE TV-J VALUES WITH INPUTS NO 31 THRU 69
C
C 1 THRU 5, INSTRUMENT NAME  (INPUTS 31 THRU 35)
C
C 6 FOCAL LENGTH, CM (36)
C 7 CLEAR APERTURE DIAMETER, CM  (37)
C 8 OBSCURATION DIAMETER, CM  (38)
C 9 OPTICS TRANSMISSION,REFL + ABSORP, FRACTION  (39)
C SENSOR DIMENSIONS,  CM
C 10 VERTICAL, ACROSS SCAN LINES  (40)
C 11 HORIZONTAL, ALONG SCAN LINE  (41)
C 12 NUMBER OF SCAN LINES  (42)
C 13 NUMBER OF PIXELS PER SCAN LINE (43)
C 14 NUMBER OF BITS PER PIXEL (44)
C 15 MINIMUM DETECTABLE EXPOSURE, ERGS/SQ CM
C 16 MAXIMUM OR HIGHLIGHT EXPOSURE, ERGS/SQ CM (46)
C 17 THRU 20, MINIMUM RESOLUTION LEVELS
C 21 SENSOR RESPONSE CURVE IDENTIFICATION (51)
C 22 BLUE FILTER TRANSMISSION CURVE ID (52)
C 23 GREEN FILTER                        (53)
C 24 RED FILTER                          (54)
C 25 POLARIZING FILTER                            (55)
C 26 EXPOSURE TIME          (56)
C 27 THRU 28 OPEN    (57 THRU 58)
C 29 FRAME READ TIME, SECONDS (59)
C 30 FRAME ERASE TIME, SECONDS (60)
C 31 PICTURE TAKING START TIME (DAYS)  (61)
C 32 PICTURE TAKING STOP TIME (DAYS)  (62)
C 33 DELTA T PER PICTURE   (MIN)    (63)    CAN BE CHANGED BY PROGRAM
C 34 BULK STORAGE CAPACITY   (BITS)  (64)
C 35 BULK ZERO STATE         (BITS)  (65)
C 36 TRANSMISSION BIT RATE,BPS   (66)
C 37 OTHER INSTRUMENT BIT RATES   (67)
C 38 BUFFER CAPACITY   (BITS)   (68)
C 39 PICTURES PER CALCULATION STEP   (69)
```

Table 6-15 (cont)

```
C P(40) THRU P(51) ARE COMPUTED
C 40 DELTA T PER PICTURE   (MIN)    (70)    FIXED
C 41 F/NUMBER OR F/STOP
C 42 AREA BLOCKAGE, FRACTION
C 43 DIFFRACTION LIMIT, RADIANS
C 44 FIELD OF VIEW, VERTICAL, DEGREES
C 45 FIELD OF VIEW, HORIZONTAL, DEGREES
C ANGULAR RESOLUTION, RADIANS/PIXEL
C 46 AT CENTER
C 47 AT VERTICAL EDGE,CENTER
C 48 AT HORIZONTAL EDGE,CENTER
C 49 BITS PER PICTURE
C 50 READ RATE, BPS
C 51 T/NUMBER (CAN BE CHANGED BY PROGRAM
C 52 T/NUMBER-FIXED
C
      INSTR=INST-100
      J=INSTR/10
      IF(J.LT.1.OR.J.GT.6) J=1
      GO TO (1,2,3,4,5,6),J
    1 CONTINUE
C
C  NARROW ANGLE SIT, INSTR=110
C
      P(1)=6HSIT NA
      P(2)=6HRROW A
      P(3)=6HNGLE C
      P(4)=6HAMERA
      P(5)=6H(CAND)
      P(6)=200.0
      P(7)=23.0
      P(15)=8.E-6
      P(16)=P(15)*250.
      P(21)=1.0
      P(26)=0.001
      P(27)=0.
      P(28)=0.
      P(29)=40.
      P(30)=0.0
      GO TO 7
    2 CONTINUE
C
C  WIDE ANGLE SIT, INSTR=120
C
      P(1)=6HSIT WI
      P(2)=6HDE ANG
      P(3)=6HLE CAM
      P(4)=6HERA
      P(5)=6H(CAND)
      P(6)=20.
      P(7)=5.0
```

Table 6-15 (cont)

```
      P(15)=8.E-6
      P(16)=P(15)*250.
      P(26)=0.0001
      P(27)=0.001
      P(28)=0.010
      GO TO 7
    3 CONTINUE
C
C NARROW ANGLE SILICON VIDICON    INSTR=130
C
      P(1)=6HSI NAR
      P(2)=6HROW AN
      P(3)=6HGLE CA
      P(4)=6HMERA
      P(5)=6H
      P(6)=200.0
      P(7)=23.0
      P(16)=0.5
      P(15)=P(16)/250.
      P(26)=.05
      P(27)=.10
      P(28)=.50
      GO TO 7
    4 CONTINUE
C
C   WIDE ANGLE SILICON VIDICON    INSTR=140
C
      P(1)=6HSI WID
      P(2)=6HE ANGL
      P(3)=6HE CAME
      P(4)=6HRA
      P(5)=6H
      P(6)=20.0
      P(7)=5.0
      P(16)=0.5
      P(15)=P(16)/250.
      P(26)=.1
      P(27)=.5
      P(28)=1.
      GO TO 7
    5 CONTINUE
C
C  NARROW SEC VIDICON   INSTR=150
C
      P(1)=6HOPE 2
      P(2)=6HDEG FO
      P(3)=6HV CAME
      P(4)=6HRA SEC
      P(5)=6H  TUBE
      P(6)=50.8
      P(7)=7.62
      P(15)=0.0001
```

Table 6-15 (cont)

```
      P(16)=0.01
      P(26)=0.4
      P(27)=.1
      P(28)=.01
      GO TO 7
    6 CONTINUE
C
C   WIDE  SEC VIDICON   INSTR=160
C
      P(1)=6HSEC WI
      P(2)=6HDE ANG
      P(3)=6HLE CAM
      P(4)=6HERA
      P(5)=6H
      P(7)=1.524
      P(6)=10.16
      P(15)=0.0001
      P(16)=0.01
      P(26)=.1
      P(27)=0.01
      P(28)=0.001
      CONTINUE
      P(8)=0.0
      P(9) = 0.85
      P(10)=1.60
      P(11)=1.60
      P(12)=800.
      P(13)=800.
      P(14)=8.0
      P(17)=300.
      P(18)=100.0
      P(19)= 50.0
      P(20)=10.0
      P(21) = 1.0
      P(22) = 1.0
      P(23)=1.
      P(24)=1.0
      P(25)=-1.0
      P(29) =40.
      P(30) = 0.1
      P(31)=-1.5
      P(32)=1.5
      P(33)=15.0
      P(34)=1000000000.
      P(35)=0.0
      P(36)=47400.
      P(37)=100.
      P(38)=P(14)*P(12)*P(13)
      P(39)=4.
```

Table 6-15 (concl)

```
C OVERLAYS SYTEM VALUES WITH USER INPUTS
      DO 77 I=31,35
      IF(ITEST(I).EQ.1) P(I-30)=TVNAME(I-30)
   77 CONTINUE
      DO 8 I=36,70
      IF(ITEST(I).EQ.1) P(I-30)=XZ(I)
    8 CONTINUE
      P(40)=P(33)
      P(41)=P(6)/P(7)
      P(42)=((P(8)/P(7))**2)
      P(43)=(6.71E-05)/P(7)
      P(45)=2.*ATAN(P(11)/(2.*P(6)))*180./PI
      P(44)=2.*ATAN(P(10)/(2.*P(6)))*180./PI
      U=P(10)/(P(12)*P(6))
      V=COS(P(44))**2
      P(46)=ATAN(U/(1.-2.*(U**2)))
      P(47)=ATAN(U*V/(1.-2.*((U*V)**2)))
      U=P(11)/(P(13)*P(6))
      V=COS(P(45))**2
      P(48)=ATAN(U*V/(1.-2.*((U*V)**2)))
      P(49)=P(14)*P(12)*P(13)
      P(50) = P(49)/P(29)
      P(51)=P(41)/SQRT((1.+P(42))*P(9))
      P(52)=P(51)
      DO 40 I=1,5
   40 TVNAME(I)=P(I)
      DO 10 I=53,60
   10 P(I)=0.
      WRITE(6,1040)TITLE
      WRITE(6,1010) (P(I),I=1,5)
      DO 20 I= 6,40
      IF(ITEST(I+30).EQ.1) WRITE(6,1020)I,TVAR(I-5),P(I)
   20 IF(ITEST(I+30).NE.1) WRITE(6,1000)I,TVAR(I-5),P(I)
      DO 30 I=41,60
   30 WRITE(6,1030)I,TVAR(I-5),P(I)
C
      CALL FOLDER(INST)
      CALL PICIN
      RETURN
 1000 FORMAT(5X,I2,5X,A6,2H= ,E15.8)
 1010 FORMAT(5X    ,23HTV CHARACTERISTICS FOR ,5A6/)
 1020 FORMAT(5X,I2,5X,A6,2H= ,E15.8,5X,11H USER INPUT )
 1030 FORMAT(5X,I2,5X,A6,2H= ,E15.8,5X,15H COMPUTED VALUES )
 1040 FORMAT(1H1,13A6,/)
      END
      SUBROUTINE UNORM(V,A,U)
      DIMENSION V(3)
      U=SQRT(V(1)**2+V(2)**2+V(3)**2)
      X=A/U
      V(1)=X*V(1)
      V(2)=X*V(2)
      V(3)=X*V(3)
      RETURN
      END
```

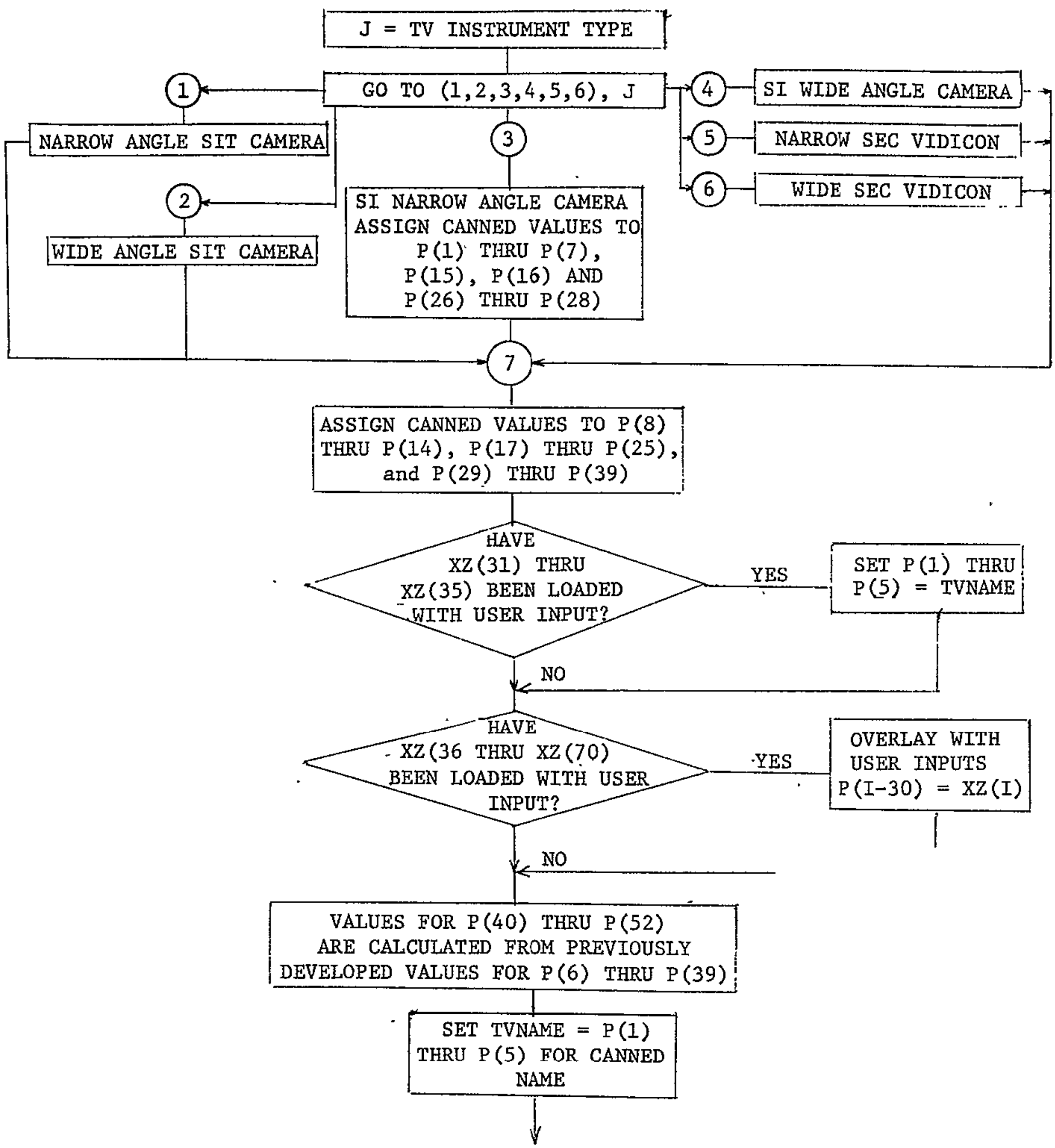


Fig. 6-15 TVCHAR Flow Diagram

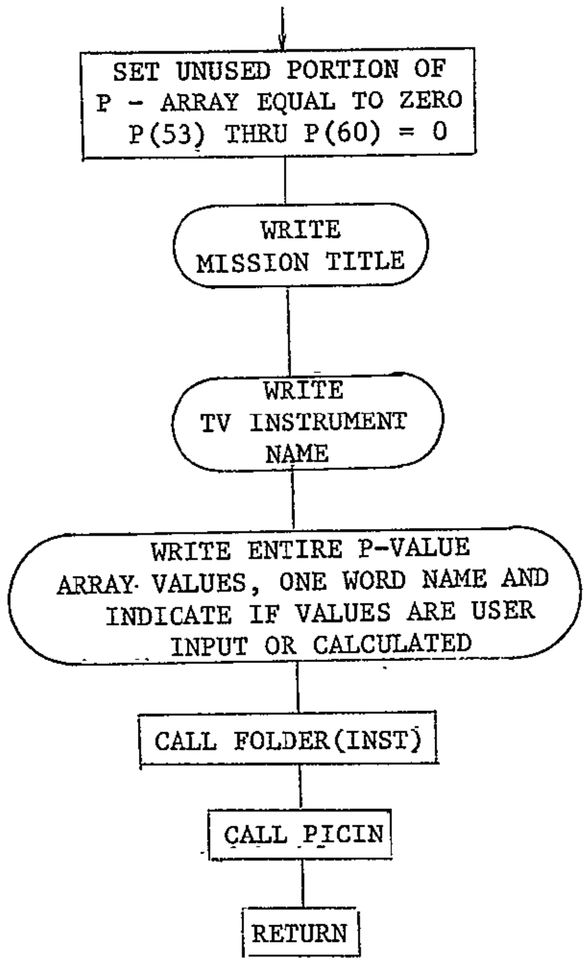


Fig. 6-15 (concl)

6.3.16 XLIMB

Purpose: To plot the outline of the planet disk on the cone and clock angle plot.

Calling Sequence: CALL XLIMB

Input/Output:

I/O	FORTRAN Name	Definition
I	THA	Half angle of planet disk (radians)
I	R(3)	Range, cone, and clock angle to planet center from spacecraft
I	DEI	Inches per degree on cone and clock angle plot
I	K	Direction Index: = 1 departure plot zero cone angle in center; = 2 arrival plot 180 deg cone angle in center.

Subprograms Required: SPHERE, VECTR1, SETBEM

Approximate Storage Required (octal): 246

Discussion: The only subroutine output is the plotted planet outline.

Table 6-16 presents the XLIMB listing; Fig. 6-16 is the XLIMB flow diagram.

Table 6-16 XLIMB Listing

```
      SUBROUTINE XLIMB
      COMMON/PTLB/THA,R(3),DEI,K
      DIMENSION FVL(3)
      PI=3.1415926536
      RTD=57.2957795
      DTR=0.01745329
      THE=-PI/18.
      STHA=SIN(R(3)*DTR)
      CTHA=COS(R(3)*DTR)
      SPHI=SIN(R(2)*DTR)
      CPHI=COS(R(2)*DTR)
      PSI=THA
      CPSI=COS(PSI)
      SPSI=SIN(PSI)
      DO 8I=1,37
      THE=THE+PI/18.
      CTHE=COS(THE)
      STHE=SIN(THE)
      AF=CTHA*CPSI*SPHI-STHA*SPSI*CTHE-CTHA*SPSI*CPHI*STHE
      BF=STHA*CPSI*SPHI+CTHA*SPSI*CTHE-STHA*SPSI*CPHI*STHE
      CF=CPSI*CPHI+SPSI*SPHI*STHE
      FVL(1)=AF
      FVL(2)=BF
      FVL(3)=CF
      CALL SPHERE(FVL,FVL,2HTO,5HPOLAR,6HDEGREE)
      A=DEI*ABS(180.*(K-1)-FVL(2))
      ZON=FVL(3)*DTR-PI/2.
      X=A*COS(ZON)
      Y=A*SIN(ZON)
      IF(K.EQ.2.AND.FVL(2).LT.90.)GO TO 10
      IF(K.EQ.1.AND.FVL(2).GT.90.) GO TO 10
      IF(I.EQ.1) CALL POINT(X,Y)
      CALL VECTOR(X,Y)
10    CALL SET BEAM(X,Y)
    8 CONTINUE
      RETURN
      END
```

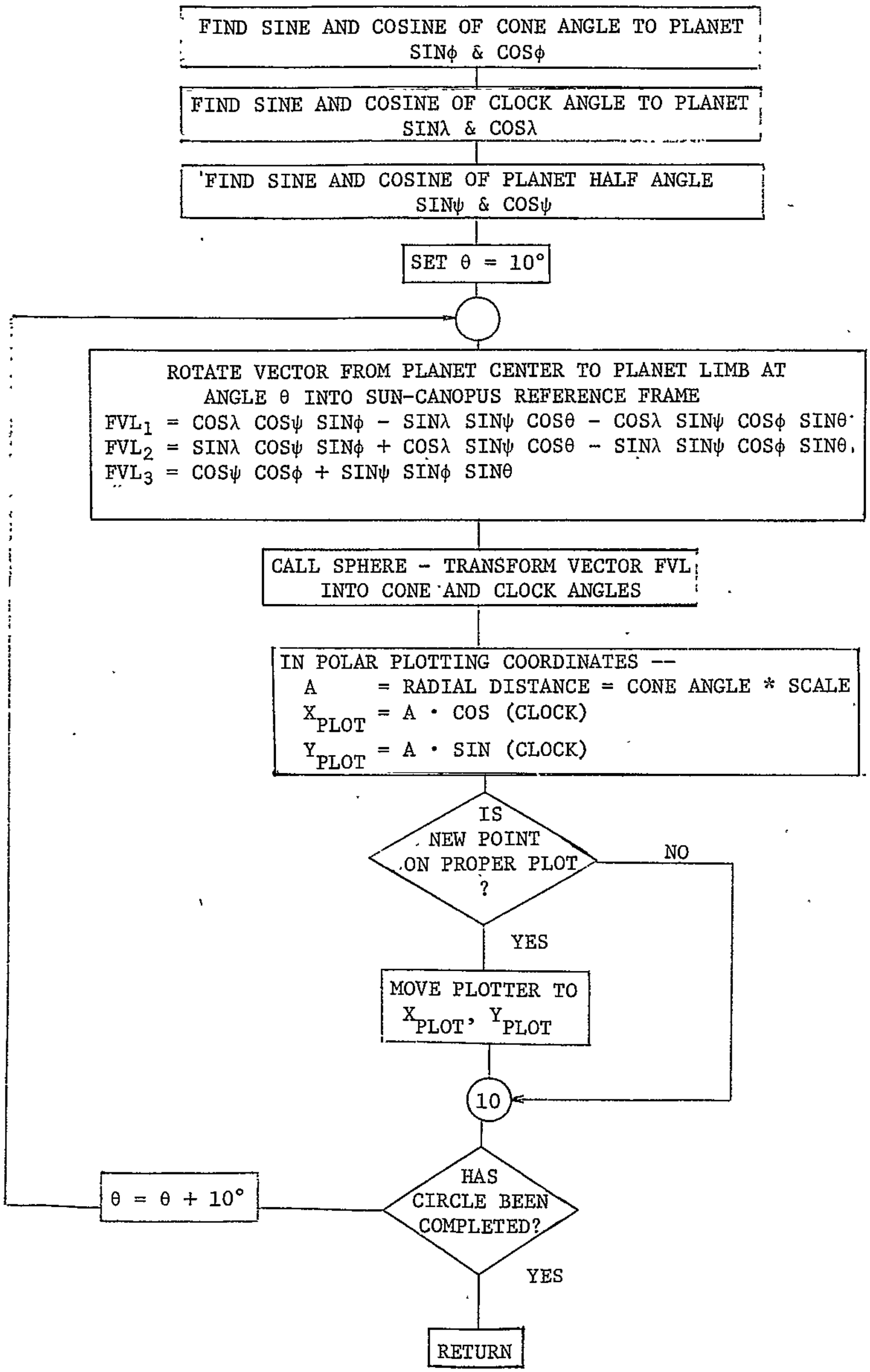


Fig. 6-16 XLIMB Flow Diagram

6.4 Minor Subroutines Developed under Contract

6.4.1 CIRC

Purpose: To plot a polygon approximating a circle on the CDC 280 display.

Calling Sequence: CALL CIRC (R, D)

Input/Output:

I/O	FORTRAN Name	Definition
I	R	Circle radius (distance to vertices of polygon)
I	D	Delta angle (central angle in radians) setting polygon side length

Subprograms Required: LINE

Discussion: The subroutine makes use of subroutine LINE, which plots a straight line from point X_1, Y_1 to point X_2, Y_2. With X_1 set equal to R COS(TH) and Y_1 equal to R SIN(TH), the angle theta, originally set equal to zero, is incremented by the delta angle, D, to produce the second point X_2, Y_2. A line is drawn between the two points, X_1 is made equal to X_2 and Y_1 equal to Y_2 and the process repeated until the angle TH equals 2π. Tests are provided to prevent plotting the circle outside the scale limits of the plot.

Table 6-17 presents the CIRC listing.

Table 6-17 CIRC Listing

```
      SUBROUTINE  CIRC(R,D)
      TH=0
1     X1=R*COS(TH)
      Y1=R*SIN(TH)
10    TH=TH+D
      IF(ABS(Y1).GT.2.625) GO TO 1
      X2=R*COS(TH)
      Y2=R*SIN(TH)
      IF( ABS(Y2).GT.2.625) GO TO 12
      CALL LINE(X1,Y1,X2,Y2)
12    Y1=Y2
      X1=X2
      IF (TH-6.2832)10,11,11
11    RETURN
      END
```

6.4.2 FPCHAR

Purpose: To assign a block of input variables to various field and particles instrument characteristics.

Call Sequence: FPCHAR (INST)

Input/Output:

I/O	FORTRAN Name	Definition
I	INST	Instrument type index to select between variuous sets of canned instrument characteristics
I	XZ(97)	Input characteristics
O	P(60)	Output translation to instruments characteristics

Subprograms Required: None

Approximate Storage Requirements (octal): 206

Discussion: This subroutine assigns two sets of canned values to the first ten values of the XZ array. One set is specifically designed to suit the conditions of a Saturn flyby. In addition, all the values are printed with an indication of whether the input variable was the result of a canned input or a user's input.

Table 6-18 presents the FPCHAR listing.

Table 6-18 FPCHAR Listing

```
      SUBROUTINE FPCHAR(INST)
      COMMON/HEDING/TITLE(13)
      COMMON/TNME/TVNAME(5)
      COMMON/CHARAC/P(60)
      COMMON/TEST/ITEST(97)
      COMMON/XZ/XZ(97)
C
C 1 THRU 5 INSTRUMENT NAME
C 6  MAGNETOPAUSE RANGE (RADII)
C 7 BOW SHOCK RANGE (RADII)
C 8  LAT MAG POLE (DEG)
C 9  LON MAG POLE (DEG)
C 10 INSTRUMENT SENSITIVITY (GAUSS)
C
      P(1)=6HFIELDS
      P(2)=6H AND P
      P(3)=6HARTICL
      P(4)=6HE INST
      P(5)=6HRUMENT
      IF(XZ(2).EQ.6) GO TO 90
      P(6)=51.6
      P(7)=65.8
      P(8)=80.0
      P(9)=190.0
      GO TO 95
   90 CONTINUE
      P(6)=63.0
      P(7)=80.3
      P(8)=90.0
      P(9)=0.0
   95 CONTINUE
      P(10)=0.000002
      IF(ITEST(1).EQ.0) GO TO 110
      DO 100 I=1,5
      P(I)=TVNAME(I)
  100 CONTINUE
  110 CONTINUE
      DO 120 I=6,10
      P(I)=XZ(I+30)
  120 CONTINUE
      DO 130 I=1,5
  130 TVNAME(I)=P(I)
      WRITE(6,1000) TITLE
      WRITE(6,1010) (P(I),I=1,5)
      DO 140 I=6,10
      IF(ITEST(I+30).EQ.1) WRITE(6,1020)  I,P(I)
      IF(ITEST(I+30).NE.1) WRITE(6,1030)  I,P(I)
  140 CONTINUE
      RETURN
 1000 FORMAT(1H1,13A6/)
 1010 FORMAT(5X,23HFP CHARACTERISTICS FOR ,5A6/)
 1020 FORMAT(5X,I2,5X,6X,2H= ,E15.8,5X,10HUSER INPUT)
 1030 FORMAT(5X,I2,5X,6X,2H= ,E15.8)
      END
```

6.4.3 IRCHAR

Purpose: To assign a block on input variables to various infrared imagers.

Call Sequence: CALL IRCHAR (INST)

Input/Output:

I/O	FORTRAN Name	Definition
I	INST	Instrument type index to select between various sets of canned instrument characteristics

Subprograms Required: None

Approximate Storage Requirements (octal): 11

Discussion: This subroutine is a dummy one at the present time and only returns control to calling program when called.

Table 6-19 presents the IRCHAR listing.

Table 6-19 IRCHAR Listing

```
SUBROUTINE IRCHAR(INST)
COMMON/HEDING/TITLE(13)
COMMON/CHARAC/P(60)
RETURN
END
```

6.4.4 NOCHAR

Purpose: To assign a block of input variables to the instrument characteristics array when no instrument is being used, for example, in a planet occultation mission.

Calling Sequence: CALL NOCHAR (INST)

Input/Output:

I/O	FORTRAN Name	Definition
I	INST	Index to assign various possible future uses of the routine.

Subprograms Required: None

Approximate Storage Requirements (octal): 11

Discussion: This subroutine is a dummy one at the present time and only returns control to the calling program when called.

Table 6-20 presents the NOCHAR listing.

Table 6-20 NOCHAR Listing

```
SUBROUTINE NOCHAR(INST)
COMMON/HEDING/TITLE(13)
COMMON/CHARAC/P(60)
RETURN
END
```

6.4.5 OUTS

Purpose: To print data computed by the program for a specific TV frame. Most of the outputs are printed without conversion by the subroutine.

Calling Sequence: CALL OUTS

Input/Output:

I/O	FORTRAN Name	Definition
I/O	TITLE	Mission title (78 Characters)
I/O	IPCN	Picture number
O	ICK(3)	Time from periapsis in days, hours, and minutes
I/O	TJDV	Julian date of picture
I/O	PLNAME	Planet name
O	SATNAM	Satellite name
I/O	R(3)	Range, cone, and clock of planet from S/C
I/O	RSAT(3)	Range, cone, and clock of specified satellite
I/O	PHASED	Phase angle of planet (deg)
I/O	PHASEP	Phase angle of specified satellite (deg)
I/O	HANG2	Angle subtended by planet (deg)
I/O	SAF(8)	Half angle of all satellites (deg)
I/O	RES	Resolution at specified location (km/pixel)
I/O	REC(9)	Resolution of sub-S/C point (km/pixel)
I/O	REL(9)	Resolution at limb (km/pixel)
I/O	SPSMR	Smear due to relative velocities (km/sec)
I/O	SPSMRS	Smear due to S/C attitude rates (km/sec)
O	SMRKM	Object motion for all sources (km)
I/O	SPSMRR	Number of pixels smeared
O	TSMRMS	Exposure time for one-half pixel smeal (millisec)
O	RESSMR	Effective resolution with smear at specified location
O	RECSMR	Effective resolution with smear at sub-S/C point
O	RELSMR	Effective resolution with smear at limb

I/O	TVNAME	Instrument title (30 characters)
O	TEXP	Exposure time (millisec)
I/O	POWLN	Power density (ergs/cm^2/sec) at lens
I/O	POWER	Power density (ergs/cm^2/sec) on sensor
I/O	ENERGY	Exposure (ergs)
I/O	PERCT	Percent of dynamic range for exposure
I/O	NDN	Digital number of exposure
I/O	IND	Total number of available levels
I/O	EPERDN	Exposure per digital number at exposure level
I/O	P(60)	Output is the four resolution levels P(17), P(18), P(19), and P(20)
I/O	PICS(4,9)	Number of pictures with resolution better than four levels for planet and eight satellites
O	ISAT	Satellite number
I/O	XRCC(3,8)	Range, cone, and clock angles for satellite from S/C
O	PHASEZ	Phase angle of unspecified satellites (deg)
I/O	SSVC8	Phase angle of all satellites (deg)

Subprograms Required: PSCONS, TCONV

Approximate Storage Required (octal): 1242

Discussion: The sole purpose of the routine is to print out a set of variables previously calculated by other parts of the program. For the most part, the routine accomplishes this without further processing of the output data. The data handled in this fashion are shown with an I/O indication in the table above. Several outputs are processed; however, the variables, ICK(3), TEXP, and PHASEZ represent simple changes in unit sizes. SATNAM and ISAT originate within the routine while the variables SMRKM, TSMRMS, RESSMR, RECSMR and RELSMR are new variables that are simple combinations of input ones.

Table 6-21 presents the OUTS list.

Table 6-21 OUTS Listing

```
      SUBROUTINE OUTS
      COMMON/RAZOR/PICS(4,9),SPPICS(4)
      DIMENSION SATNAM(2) ,SATELL(2)
      COMMON /PLCONS/GMS,GMPL(12),RS,RPL(12),SNAME(2),PLNAME(2,12),
     X SPV(3),PLPV(3,12),SVE(3),PLVE(3,12),SROT,PLROT(12),SOBL,PLOBL(12)
      COMMON/POBEAR/SSV(8),SAF(8),HANG2,JCNCLK(8)
      COMMON/HEDING/TITLE(13)
      COMMON/TNME/TVNAME(5)
      REAL JDV
      DIMENSION ICK(4)
      COMMON/JDAYS/TJDV,TT
      COMMON/FOROUT/POWER,ENERGY,PERCT,IND,R(3),RSAT(3),RAD1(3),IPCN
     X,FRACT,PHASED,PHASEP   ,POWLN,NDN,EPERDN
      COMMON/FLAGS/IFLT,IFLPL,IFLPR
      COMMON/CHARLIE/NCODE,NREC ,IP,IVS,NSC,IFLST,NCB
      COMMON/FLYBY/DUM1(6),X(3),DUM2(9)
      COMMON/HANDO/TIM,REC(9),REL(9),RES,SMR(9),SMRR(9),SMRS(9),TSMR(9)
     X,SPSMR,SPSMRS,SPTSMR,SPSMRR
      COMMON/PHASER/PHASE(8),XRCC(3,8)
      DIMENSION NS(9)
      COMMON/CHARAC/P(60)
      DATA NS/0,0,1,2,4,7,5,1,0/
      DATA IFP/0/
C
C OUTS-2                                                      OUTS-2
C
      CALL TCONV(ICK,5HCLOCK,ABS(TT),3HSEC)
      CLOCK=ICK(1)
      CLOCK=SIGN(CLOCK,TT)
      ICK(1)=CLOCK
      WRITE(6,1001)TITLE
      WRITE(6,1000)IPCN,(ICK(I),I=1,3) ,TJDV
      CALL PSCONS(SATNAM,4HNAME,IVS,IP)
      WRITE(6,1010) PLNAME(1,IP),PLNAME(2,IP),SATNAM
      WRITE(6,1020) (R(I),RSAT(I),I=1,3)
      WRITE(6,1030) PHASED,PHASEP,HANG2,SAF(IVS)
      IF(NCB.EQ.1) GO TO 100
      WRITE(6,1040) PLNAME(1,IP),PLNAME(2,IP),RAD1(2),RAD1(3)
      GO TO 110
  100 WRITE(6,1045) SATNAM
  110 CONTINUE
      WRITE(6,1060) RES,REC(1),REL(1)
      SMRKM=SPSMRR * RES
      RESSMR=SMRKM + RES
      RECSMR=SMRKM + REC(1)
      RELSMR=SMRKM + REL(1)
      WRITE(6,1070) SPSMR,SPSMRS,SMRKM,SPSMRR
      TSMRMS=SPTSMR*1.E+3
      WRITE(6,1080) TSMRMS
      WRITE(6,1081) RESSMR,RECSMR,RELSMR
      TEXP=P(26)*1.E+3
      WRITE(6,1090) TVNAME,TEXP,POWLN,POWER,ENERGY
      WRITE(6,1091) PERCT,NDN,IND,EPERDN
```

Table 6-21 (cont)

```
      NSP=NS(IP)
      WRITE(6,3000) (P(I),I=17,20)
      WRITE(6,3010) (PICS(I,1),I=1,4)
      NSP=NSP+1
      DO 300 J=2,NSP
      ISAT=J-1
  300 WRITE(6,3020) ISAT,(PICS(I,J),I=1,4)
      NSP=NSP-1
      WRITE(6,5000)
      WRITE(6,5010)
      DO 500 I=1,NSP
      PHASEZ=PHASE(I)*(180./3.14159)
  500 WRITE(6,5020) I,XRCC(1,I),PHASEZ  ,XRCC(2,I),XRCC(3,I)
      WRITE(6,5070)
      DO 530 I=1,NSP
      CALL PSCONS(SATELL,4HNAME,I   ,IP)
  530 WRITE(6,5040) I,SSV(I),SAF(I)
      NSP=NSP+1
      WRITE(6,5030)
      DO 510 I=2,NSP
      NST=I-1
  510 WRITE(6,5040) NST,REC(I),REL(I)
      WRITE(6,5060)
      DO 520 I=2,NSP
      NST=I-1
  520 WRITE(6,5050) SMR(I),SMRS(I),SMRR(I),TSMR(I)
      RETURN
 1000 FORMAT(     5X,15HPICTURE NUMBER ,I5,19X,I3,6H DAYS ,I2,5H HRS ,
     XI2,17H MIN TO ENCOUNTER,10X,3HJD ,F12.3,//)
 1001 FORMAT(1H1,13A6,//)
 1005 FORMAT(//5X,80(1H-)//,5X,15HPICTURE NUMBER ,I5,19X,I3,6H DAYS ,I2,
     X5H HRS ,I2,17H MIN TO ENCOUNTER//)
 1010 FORMAT(5X,7HPLANET ,2A6,20X,10HSATELLITE ,2A6)
 1020 FORMAT(5X,13HRANGE (KM)   =,E16.8,11X,13HRANGE (KM)   =,E16.8/
     X        5X,13HCONE   (DEG) =,F7.2 ,20X,13HCONE   (DEG) =,F7.2 /
     X        5X,13HCLOCK  (DEG) =,F7.2 ,20X,13HCLOCK  (DEG) =,F7.2 )
 1030 FORMAT(5X,13HPHASE  (DEG) =,F7.2 ,20X,13HPHASE  (DEG) =,F7.2/
     $        5X,13HDIAM   (DEG) =,F7.2 ,20X,13HDIAM   (DEG) =,F7.2//)
 1040 FORMAT(5X,27HINSTRUMENT DATA FOR PLANET ,2A6,5X,
     $ 21H(PICTURE CENTERED AT  ,F6.2,9H DEG LAT ,F7.2,11H DEG LONG )/)
 1045 FORMAT(5X,30HINSTRUMENT DATA FOR SATELLITE ,2A6/)
 1060 FORMAT(4X,26HRESOLUTION FOR TARGET BODY ,/,
     $ 5X,24HAT SPECIFIED LOCATION    ,F8.1,9H KM/PIXEL /
     $ 5X,24HAT SUBSPACECRAFT POINT   ,F8.1,9H KM/PIXEL /
     $ 5X,24HAT LIMB OF TARGET BODY  ,F8.1,9H KM/PIXEL /)
 1070 FORMAT(5X,34HSMEAR DUE TO RELATIVE VELOCITIES  ,F8.2,7H KM/SEC /
     $        5X,34HSMEAR DUE TO S/C ATTITUDE RATES   ,F8.2,7H KM/SEC /
     $        5X,34HOBJECT MOTION FROM ALL SOURCES    ,F8.2,3H KM /
     $        5X,34HNUMBER OF PIXELS SMEARED          ,F8.5,7H PIXELS //)
 1080 FORMAT(4X,39HEXPOSURE TIME FOR ONE-HALF PIXEL SMEAR ,F8.3,
     $ 9H MILLISEC //)
```

Table 6-21 (concl)

```
1081 FORMAT(4X,31HEFFECTIVE RESOLUTION WITH SMEAR /
    $ 5X,24HAT SPECIFIED LOCATION   ,F8.1,9H KM/PIXEL /
    $ 5X,24HAT SUBSPACECRAFT POINT  ,F8.1,9H KM/PIXEL /
    $ 5X,24HAT LIMB OF TARGET BODY  ,F8.1,9H KM/PIXEL /)
1090 FORMAT(4X,18HEXPOSURE TIME FOR  ,5A6,3H = ,F8.3,9H MILLISEC /
    $ 5X,16HPOWER DENSITY =  ,E10.4,24H ERGS/SQ CM/SEC AT LENS  /
    $ 5X,16HPOWER DENSITY =  ,E10.4,26H ERGS/SQ CM/SEC ON SENSOR    /
    $ 5X,16HEXPOSURE      =  ,E10.4,11H ERGS/SQ CM  //)
1091 FORMAT(5X,26HPERCENT OF DYNAMIC RANGE =  ,F7.3 /
    $       5X,26HDIGITAL NUMBER OF EXPOS  =  ,I4,4H (OF,I4,8H LEVELS)/
    $       5X,26HONE DIGITAL NUMBER EQUALS  ,E10.4,12H ERGS/SQ CM //)
3000 FORMAT(4X,46HNUMBER OF PICTURES WITH RESOLUTION BETTER THAN   /
    $ 5X,17HRESOLUTION LEVELS ,3X,4(F5.1,10X) ,9H KM/PIXEL  /)
3010 FORMAT(5X,06HPLANET,11X,4(F10.1,5X))
3020 FORMAT(5X,10HSATELLITE ,I1,9X,4(F10.4,5X))
5000 FORMAT(1H1)
5010 FORMAT(     5X,14HSATELLITE DATA//7X,3HNUM,5X,10HRANGE (KM),10X,
    X11HPHASE (DEG),4X,10HCONE (DEG),5X,11HCLOCK (DEG)/)
5020 FORMAT(8X,I1,4X,E15.8,6X,F7.2,8X,F7.2,8X,F7.2)
5030 FORMAT(   /5X,39HRESOLUTION (KM/PIX) DATA FOR SATELLITES   /
    X,7X,3HNUM,3X,23HRES ALONG RADIUS VECTOR,12X,21HRES AT SATELLITE LI
    XMB/)
5040 FORMAT(8X,I1,8X,E15.8,20X,E15.8)
5050 FORMAT(4(5X,E15.8))
5060 FORMAT(    /,5X,26HSMEAR DATA FOR SATELLITES  /
    X5X,12HSMEAR DUE TO ,8X,12HSMEAR DUE TO,10X,09HNUMBER OF,9X,13HTIME
    X TO SMEAR/
    X5X,12HVEL (KM/SEC),08X,12HS/C (KM/SEC),08X,14HPIXELS SMEARED,6X,
    X15H1/2 PIXEL (SEC)/)
5070 FORMAT(   /,7X,3HNUM,7X,17HEARTH-SAT-S/C ANG ,
    $ 17X,18HSATELLITE ANG DIAM/)
     END
```

6.4.6 SCROT

Purpose: To provide spacecraft rotation rates from a set of canned values or from user inputs.

Calling Sequence: CALL SCROT (ROT W)

Input/Output:

I/O	FORTRAN Name	Definition
I	ROT(6)	Two sets of canned S/C rotation rates (deg/sec)
I	XZ (19)	User input X axis rate (deg/sec)
I	XZ(20)	User input Y axis rate (deg/sec)
I	XZ(21)	User input Z axis rate (deg/sec)
O	ROTW(3)	Spacecraft rotation rate vector (radians/sec)

Subprograms Required: None

Approximate Storage Required (octal): 101

Discussion: None

le 6-22 presents the SCROT listing.

Table 6-22 SCROT Listing

```
      SUBROUTINE SCROT(ROTW)
      COMMON/XZ/XZ(97)
      DIMENSION ROT(6),ROTW(3)
      DATA ROT/.01,.01,.01,.01,.01,90./
      DATA PI/3.14159/
      NS=XZ(18)
      IF(NS.LT.0) NS=0
      NS=NS+1
      GO TO (1,2,3,3,3)NS
1     DO 5 I=1,3
5     ROTW(I)=ROT(I)*(PI/180.)
      RETURN
2     DO 6 I=4,6
6     ROTW(I-3)=ROT(I)*(PI/180.)
      RETURN
3     DO 7 I=1,3
7     ROTW(I)=XZ(I+18)*(PI/180.)
      RETURN
      END
```

6.4.7 UVCHAR

Purpose: To assign a block of input variables to various ultraviolet spectrometer characteristics.

Call Sequence: CALL UVCHAR (INST)

Input/Output:

I/O	FORTRAN Name	Definition
I	INST	Instrument type index to select between various sets of canned instrument characteristics

Subprograms Required: None

Approximate Storage Requirements (octal): 11

Discussion: This subroutine is a dummy one at the present time, and only returns control to calling program when called.

Table 6-23 presents the UVCHAR listing.

Table 6-23 UVCHAR Listing

```
SUBROUTINE UVCHAR(INST)
COMMON/HEDING/TITLE(13)
COMMON/CHARAC/P(60)
RETURN
END
```

6.5 Plotting Conversion Subroutines - During development of the model, the use of certain system plotting routines was necessary for operation on the CDC 6500 computer. A group of simple subroutines was written, which performed the same plotting functions, using a second group of system plotting subroutines peculiar to the plotting operation of the UNIVAC 1108 computer. A brief description of these subroutines is given below:

6.5.1 ABSBEM

Purpose: To position the plotting beam in absolute units. A value of 1.0 represents either the top or the right edge of the plotting area, while a value of 0.0 represents either the bottom or the left side of the plotting area. The X and Y arguments in the calling sequence are the coordinates of the plot beam after the call.

Calling Sequence: CALL ABSBEM (X, Y)

6.5.2 CHAROP

Purpose: To determine the print size and orientation of the plotting annotations that follow the call. In the argument list, DUM refers to unused dummy arguments. ISIZE is an index of the character size, where:

- 0 provides miniature type;
- 1 provides small type;
- 2 provides medium type;
- 3 provides large type.

IOR is an orientation endex with a value of 0 giving horizontal orientation, and a value of 1 giving vertical orientation.

Calling Sequence: CALL CHAROP (DUM, DUM, ISIZE, IOR, DUM)

6.5.3 LINE

Purpose: To plot a line from point X_1, Y_1 to point X_2, Y_2.

Calling Sequence: CALL LINE (X1, Y1, X2, Y2)

6.5.4 MAP

Purpose: To establish a correspondence between the user's coordinate system and the SC-4020 raster array. The arguments XMIN, XMAX, YMIN, and YMAX are the extreme values expected to be plotted in the user's units. The arguments XMI, XMA, YMI, and YMA are the absolute units of the plot with the point XMI, YMI describing the lower left hand corner of the plot and the point XMA, YMA the upper left corner.

Calling Sequence: CALL MAP (XMIN, XMAX, YMIN, YMAX, XMI, XMA, YMI, YMA)

6.5.5 MAPS

Purpose: To provide the same scaling function as MAP, but also to provide a frame around the plot marked with appropriate numerical values and tick marks.

Calling Sequence: CALL MAPS (XMIN, XMAX, YMIN, YMAX, XMI, XMA, YMI, YMA)

6.5.6 MAPSSL

Purpose: To provide the same functions as MAPS, except that the X-axis has a semilog scale instead of a linear one.

Calling Sequence: CALL MAPSSL (XMIN, XMAX, YMIN, YMAX, XMI, XMA, YMI, YMA)

6.5.7 POINT

Purpose: To plot a point at a given position. The arguments X and Y are the position of the point in the user's units set up by a preceding call of MAP, MAPS, or MAPSSL.

Calling Sequence: CALL POINT (X,Y)

6.5.8 SETBEM

Purpose: To position the plotting beam with respect to the user's coordinate system. The arguments X and Y are the point at which the plotting beam is positioned.

Calling Sequence: CALL SETBEM (X,Y)

6.5.9 SYMBOL

Purpose: To annotate the plot with a group of Hollerith text. The argument MH....$. is a Hollerith text of M characters. The last two characters must be $., which are not printed but do indicate the end of the string to be printed.

Calling Sequence: CALL SYMBOL (MH.....$.)