77-FM-50 JSC-13094

An Analytical Satellite Orbit Predictor (ASOP)

Mission Planning and Analysis Division

October 1977

(NASA-CR-151583) AN ANALYTICAL SATELLITE ORBIT PREDICTOR (ASOP) (Analytical and Computational Mathematics) 105 p HC A06/MF A01 CSCL 22A

N78-14066

Unclas G3/13 59492

National Aeronautics and Space Administration

Lyndon B. Johnson Space Center
Houston, Texas

77-FM-50 JSC-13094

SHUTTLE PROGRAM

AN ANALYTICAL SATELLITE ORBIT PREDICTOR (ASOP)

By Stephen E. Starke
Analytical and Computational Mathematics, Inc.
Contract Number NAS 9-15171

JSC Task Monitor: George A. Weisskopf
Software Development Branch

Approved: Elric N. McHenry
Elric N. McHenry, Chief
Software Development Branch

Approved: Ronald L. Berry
Ronald L. Berry, Chief
Mission Planning and Analysis Division

National Aeronautics and Space Administration

Lyndon B. Johnson Space Center

Mission Planning and Analysis Division

Houston, Texas

October 1977

CONTENTS

Section Page

TABLES

FIGURES

AN ANALYTICAL SATELLITE ORBIT PREDICTOR (ASOP)

By Stephen E. Starke
Analytical and Computational Mathematics, Inc.

1.0 INTRODUCTION

This report contains the documentation and user's guide for the Analytical Satellite Orbit Predictor (ASOP) computer program. ASOP is based on mathematical methods that represent a new state-of-the-art for rapid orbit computation techniques. The theoretical development of these methods has been carried out during the past few years, and they are now being put into the form of an operational computer program. ASOP is intended to be used for computation of near-Earth orbits including those of the Shuttle/Orbiter and its payloads.

Orbit computation methods can usually be given one of the following two classifications:

a. Numerical methods - The calculations are carried out in a step-by-step manner. High precision is possible, but computer runtime can be excessive.

b. Analytical methods - The calculations are carried out in one step regardless of the prediction interval. Therefore, these methods have extremely fast computation times.

In the past, analytical methods have not been widely used because they were less accurate and required much more computer codes than numerical methods. The Poincaré-Similar Element Method (PSϕ) used in ASOP overcomes these disadvantages. It is possible to compute near-Earth orbits to within an accuracy of a few meters. Recursive equations are used instead of complicated formulas. Execution time of ASOP is on the order of a few milliseconds. It is planned to include in ASOP the perturbations due to atmospheric drag, Sun and Moon gravity and all terms of the geopotential.

In reference 1, Mueller describes the relationship between the PSϕ elements and the Cartesian coordinates and establishes the PSϕ perturbed equations of motion. These elements and the associated equations of motion were used by Bond (ref. 1) to develop a completely analytical solution to the J_2 perturbation problem.

Recently, the analytical solution has been expanded by Scheifele (ref. 3) to include the perturbations due to atmospheric drag. Mueller (ref. 4) has developed the higher order zonal geopotential terms. Time dependent (tesseral) geopotential terms for the solution are under development. Later, it is expected that the third body perturbations of the Sun and Moon will also be added to the solution.

The current version of ASOP includes the J_2 geopotential terms. The coding is documented in this report. It is planned that the additional perturbations mentioned above will be added to ASOP and will be documented in separate reports.

This ASOP program has been designed in two versions: (1) a stand-alone version that can be used interactively to obtain immediate results to a specific problem and (2) a user-subroutine package that can be used by other software systems. Both versions were designed to be small and to execute quickly. The average execution time is 40 milliseconds for the stand-alone program and 5 milliseconds for the user-subroutine package, while using 3111_{10} and 2001_{10} words of storage, respectively.

Both versions of the ASOP program have been written in UNIVAC standard FORTRAN-V and are available to the public on file NUMEG under the qualifier FM6-N08569 on the UNIVAC 1110-Exec 8 system. This document is intended to instruct the user in the operation of the ASOP program on this machine and to document the individual ASOP subroutines.

2.0 PS ELEMENT FORMULATION

The PS (Poincaré-Similar) element formulation is described in the following subsections. An exact development of this element set is described in reference 2 and a description of the variables can be found in Appendix E.

2.1 TRANSFORMATION FROM CARTESIAN COORDINATES TO PS ELEMENTS (XTOPS)

The transformation into the PS element set $(\vec{\sigma},\vec{\rho})$ from a given set of Cartesian coordinates $(\overline{X},\overline{V},r)$ is accomplished within the subroutine XTOPS. It is assumed that these coordinates are the initial conditions so that the physical time t and the independent variable τ both equal zero. The PS elements are then computed using the equations

$$\sigma_1 = \tan^{-1}\left(\frac{X_x + R^*\sigma_3}{X_y + R^*\rho_3}\right) \qquad \rho_1 = G - \sqrt{G^2 + 2r^2V} + \frac{\mu}{\sqrt{2\rho_4}}$$

$$\sigma_2 = Z_2 \cos\sigma_1 - Z_1 \sin\sigma_1 \qquad \rho_2 = Z_2 \sin\sigma_1 + Z_1 \cos\sigma_1$$

$$\sigma_3 = -2G_x/\sqrt{2(G + G_z)} \qquad \rho_3 = 2G_y/\sqrt{2(G + G_z)}$$

$$\sigma_4 = t - \frac{\mu}{(2\rho_4)^{3/2}}\left(E - \phi - \frac{r}{p} Z_2\, Q\sqrt{1 - e^2}\right) \qquad \rho_4 = \frac{\mu}{r} - \frac{1}{2}\vec{V}^2 - V$$

The other required relations are:

$$G_x = X_yV_z - X_zV_y \quad , \quad G_y = X_zV_x - X_xV_z \quad , \quad G_z = X_xV_y - X_yV_x$$

$$G = \sqrt{G_x^{\,2} + G_y^{\,2} + G_z^{\,2}}$$

$$p = 1/\mu\left(G - \rho_1 + \mu/\sqrt{2\rho_4}\right)^2 \quad , \quad q = G - 1/2\ \rho_1 + \mu/2\sqrt{2\rho_4}$$

$$R^* = rR/2G \quad , \quad R = 2GX_z/r\sqrt{2(G + G_z)}$$

$$Z_1 = (p/r - 1)/Q \quad , \quad Z_2 = \dot{r}p/\left(Q(2q - G)\right)$$

$$Q = 1/\mu \left(\rho_4 \ (2\mu/\sqrt{2\rho_4} + G_z - G)\right)^{1/2}$$

$$\dot{r} = \frac{\vec{X}\cdot\vec{V}}{r} \quad , \quad \sqrt{1 - e^2} = p\sqrt{2\rho_4/\mu}$$

$$E - \phi = -2 \tan^{-1}\left(Z_2 Q/(1 + \sqrt{1 - e^2} + Z_1 Q)\right)$$

This transformation is performed only once for a given set of Cartesian coordinates.

2.2 TRANSFORMATION FROM PS ELEMENTS TO CARTESIAN COORDINATES (PSTOX)

This transformation is performed when any intermediate printout is desired or when the final condition is met. Therefore, this transformation should be coded with emphasis on the speed of calculation.

The Cartesian coordinates defined in terms of the PS elements are given by the equations:

$$X_x = r \cos \sigma_1 - R^* \sigma_3 \quad , \quad V_x = \dot{r} \cos \sigma_1 - \frac{G}{r} \sin \sigma_1 - R^* \sigma_3$$

$$X_y = r \sin \sigma_1 - R^* \rho_3 \quad , \quad V_y = \dot{r} \sin \sigma_1 - \frac{G}{r} \cos \sigma_1 - \dot{R}^* \rho_3$$

$$X_z = R^* \sqrt{2(G + H)} \quad , \quad V_z = \dot{R}^* \sqrt{2(G + H)}$$

The other necessary relationships are

$$r = p/(1 + e \cos \phi) \quad , \quad R^* = rR/2G$$

$$\dot{r} = e \sin \phi/p\left(2q - \rho_1 + 1/2 \ (\rho_2{}^2 + \sigma_2{}^2)\right)$$

$$\dot{R}^* = (R\dot{r} + \dot{R}r)/2G \quad , \quad \dot{R} = \frac{G}{r^2}(\rho_3 \cos \sigma_1 - \sigma_3 \sin \sigma_1)$$

$$R = \rho_3 \sin \sigma_1 + \sigma_3 \cos \sigma_1 \quad , \quad G = \rho_1 - 1/2 \ (\rho_2{}^2 + \sigma_2{}^2)$$

$$H = G - 1/2\ (\rho_3^{\,2} + \sigma_3^{\,2})\quad,\quad q = -1/2\ (\rho_2^{\,2} + \sigma_2^{\,2} - \rho_1 - \mu/\sqrt{2\rho_4}\,)$$

$$p = 1/\mu\ \left(\mu/\sqrt{2\rho_4} - 1/2\ (\rho_2^{\,2} + \sigma_2^{\,2})\right)$$

$$e \sin \phi = Q(\rho_2 \sin \sigma_1 + \sigma_2 \cos \sigma_1)$$

$$e \cos \phi = Q(\rho_2 \cos \sigma_1 - \sigma_2 \sin \sigma_1)$$

$$Q = 1/\mu \left\{ \rho_4\left(2\mu/\sqrt{2\rho_4} - 1/2\ (\rho_2^{\,2} + \sigma_2^{\,2})\right)\right\}^{1/2}$$

The physical time t is computed using

$$t = \sigma_4 + \frac{\mu}{(2\rho_4)^{3/2}}(E - \phi - \frac{r}{p}\ e \sin \phi \sqrt{1 - e^2}) \qquad (2.2a)$$

where an expression for $E - \phi$ is given by

$$E - \phi = -2 \tan^{-1} \left[e \sin \phi/(1 + \sqrt{1 - e^2} + e \cos \phi)\right]$$

2.3 TIME TERMINATION PROCEDURE

Because the PS element set uses the true anomaly τ as the independent variable, an iteration procedure is necessary to stop at a specific time t_{final}. Within the ASOP program, this iteration is performed by the TIMEPS subroutine in the following manner:

An expression for the derivative of time with respect to the true anomaly τ is given in the PS theory as

$$\frac{dt}{d\tau} = r^2/q$$

This expression can be linearly approximated by the expression

$$\frac{\Delta t}{\Delta \tau} = r^2/q \qquad (2.3a)$$

where $\Delta t = t_{final} - t_n$ and $\Delta\tau = \tau_{n+1} - \tau_n$. Equation 2.3a then yields a recursive formula for refining an initial estimate of τ of

$$\tau_{n+1} = \tau_n - \Delta t\, q/r^2 \tag{2.3b}$$

Using equation 2.3b, an initial estimate of τ is refined until the associated value of t equals the desired final time t_{final}.

In order to start the iteration, a suitable initial value of τ is necessary. This value is determined by first assuming that a circular orbit is being used. With this assumption, equation 2.2a reduces to

$$\sigma_4 = t$$

and an initial approximation for τ can be written as

$$\tau_o = (t_{final} - \sigma_4)/\frac{\partial\sigma_4}{\partial\tau}$$

The value of $\partial\sigma_4/\partial\tau$ is taken from the analytical theory (see reference 2) during the initialization procedure.

Therefore, the full algorithm is

a. Set the iteration counter n to zero, and compute the initial approximation for $\tau_n = \tau_o$.

b. Determine the PS elements at τ_n.

c. Determine the time t_n at τ_n.

d. If $|t_{final} - t_n| \leq$ TOLerence, then STOP; otherwise $n = n + 1$

e. If $n > n_{max}$, then print a diagnostic message and STOP

f. Compute a new approximation for τ_n using

$$\tau_n = \tau_{n-1} - (t_{n-1} - t_{final})q/r^2$$

g. Go to step b.

Values for TOLerence and n_{max} have been preset within the TIMEPS subroutine to 10^{-7} and 5, respectively.

3.0 DESCRIPTION AND STRUCTURE OF ASOP

The ASOP program was designed as a top-down structured program consisting of 11 subroutines (modules) and a main (driver) program. Within this set of subroutines, a subset of seven subroutines comprises the removable ASOP subroutine package. The package has been designed for easy incorporation into existing user software. This type of division has resulted in some redundant operations but does not seriously affect the overall performance of the ASOP program.

The two sets of subroutines are described in the following subsections and each subroutine is documented in section 3.3.

3.1 THE ASOP PROGRAM

Basically, the ASOP program, shown in figure 1, consists of four segments: a main program or driver, an input routine, an output routine, and a removable ASOP subroutine package. The removable package is described in section 3.2 so that only the first three segments will be described here.

The purpose of the main program is to call all the necessary subroutines (input, output, constants, etc.) in a specific sequence to produce the desired results. In particular, the main program provides for the repetitive call to the ASOP subroutine in order to produce a satellite ephemeris.

All input to the program is controlled by the subroutine INPUT. Its primary functions are

a. Set all default values

b. Accept input from the NAMELIST $INPUT

c. Convert any input values that require conversion

d. Issue normal program termination command

Output from the ASOP Program is performed only by the subroutine OUTPUT. This subroutine contains all the FORMAT specifications used for normal[a] output from ASOP. The output program will also perform any conversions required to make the output more understandable. This involves converting Cartesian coordinates to Keplerian elements, radians to degrees, and time units to days.

---

[a]All error output is controlled by the individual subroutines.

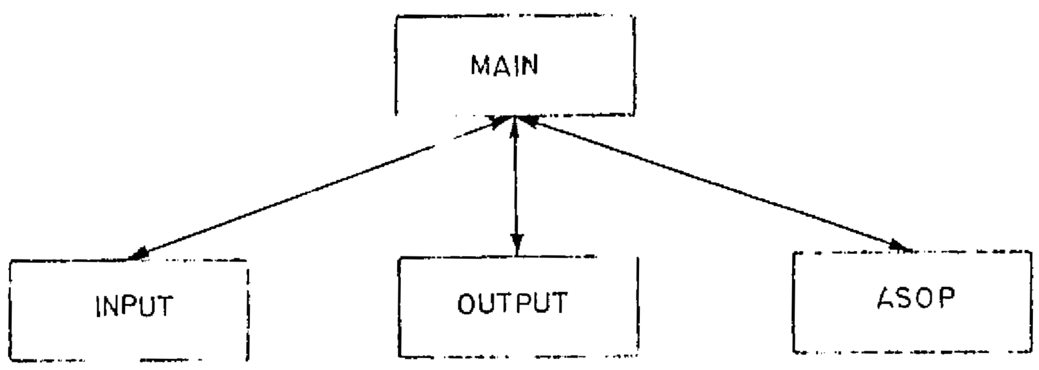


Figure 1.- Structure of the ASOP program.

3.2 THE ASOP SUBROUTINE PACKAGE

The ASOP subroutine package has been designed as an independent segment of the ASOP program so that it can be incorporated into existing software with little or no modifications.

This package, as with the ASOP program, has four basic parts (fig. 2):

a. A driver subroutine (ASOP)

b. Coordinate transformations (XTOPS, PSTOX)

c. A stopping routine (TIMEPS)

d. The analytical theory (PSANS)

Along with the five subroutines mentioned, there are two additional subroutines, POTJ2 and CONST, that perform functions that are used by more than one part so that they cannot be specifically included in one section.

Subroutine ASOP performs the same task as the main program in that it calls all the necessary subroutines as dictated by the input values. In the case of the ASOP subroutine, however, the input values are given through an argument list, and a few COMMON blocks. Therefore, if a satellite ephemeris is desired, the user must supply the necessary coding within his own software.

The basic input to the subroutine is:

a. The Cartesian coordinates (X)

b. A stop value (STOP)

c. A stop flag (ISTOP)

d. A new data flag (NEWX)

This input is fully described in sections 3.3.3 and 4.2.3.

Because the analytical theory has been developed in PS elements, it is necessary to perform transformations to and from the element set. Transformation into the PS elements from Cartesian coordinates is performed by the subroutine XTOPS while the reverse transformation is performed by the subroutine PSTOX.

A time-stop subroutine has been included because the PS elements use the true anomaly as the independent variable and do not use time. This stopping routine is an iterative procedure and is described in section 2.3.

PSANS, the analytical theory subroutine, is the nucleus of the ASOP program. At present, the theory contains only the perturbations due to J_2, and these equations are given in Appendix F. However, the theory will soon be expanded to include other perturbations such as drag (ref. 5), and the higher order geopotential terms.

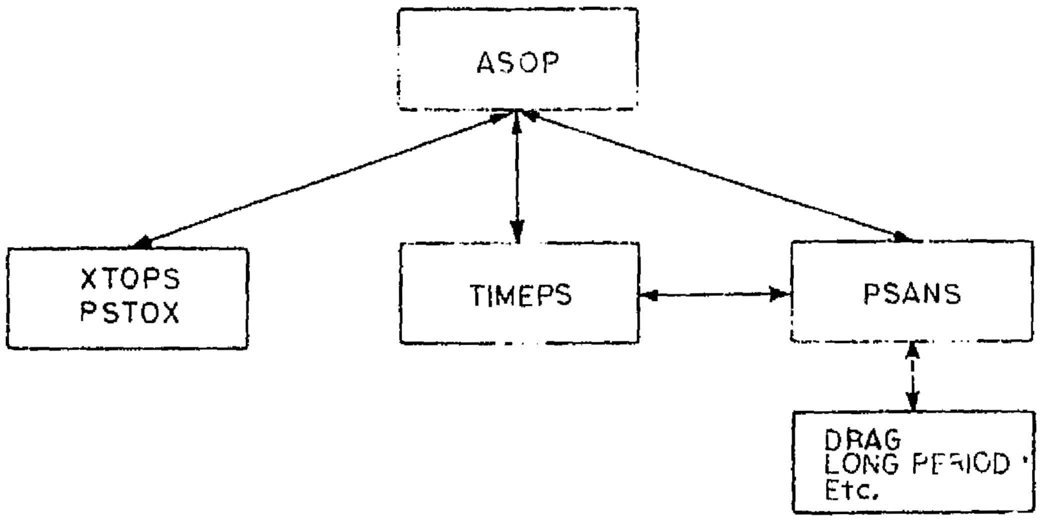


Figure 2.- Structure of the ASOP subroutine.

3.3 MODULE DESCRIPTIONS

This section will give a complete description of the subroutines currently used in the ASOP program. Each description will contain a brief statement as to the purpose or use, of the subroutine as well as a complete description of all variables used within the subroutine. Also included are lists of the named COMMON blocks used, external references to other ASOP subroutines, and other ASOP subroutines that reference the subroutine being described. Information is also available as to the calling sequence of the subroutine, subroutine size, and the average execution time of the subroutine. Each description is followed by a general flow chart of the subroutine (figs. 3 through 14).

Each program is listed alphabetically, with the exception of the MAIN program, which is described first.

3.3.1 MAIN Program (Driver)

Purpose: Driver for the Analytical Satellite Orbit Predictor (ASOP) program

Calling sequence: None

Called by: Operating system

Subroutines/functions called: ASOP, CONST, INPUT, OUTPUT

Named COMMON:		
	/CARTC /	X(8),R
	/CBASIC/	PI,TWOPI,DEG,RAD,DAY
	/CPRINT/	PRINT,IPRINT,IELPRT,IUNITS
	/END /	STOP,ISTOP
	/PS /	SIG(8),TAU

Program data: Size = $124_8(84_{10})$ words compiled.
Execution time = 34 milliseconds to 42 milliseconds, depending upon the input values for PRINT and IPRINT (does not include time for data units conversions and initial condition printout).

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
DAY	1	DP	---	sec/day or min/day or hrs/day	Conversion of days into hours, minutes, or seconds. Its numerical value is controlled by IUNITS.
IPRINT	1	I	I	--	Flag to determine if the intermediate printout is to be given on days or revolutions. = 0 no printout = 1 days = 2 revolutions

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
ISTOP	1	I	I	--	Flag to determine if the program is to terminate after a given number of days or revolutions as specified by STOP. = 1 days = 2 revolutions
NEWX	1	I	I	--	Flag to determine if the ASOP program is to be initialized. = 0 no = 1 yes
PRINT	1	DP	I	days or revs	Increment for which the intermediate printout is desired (valid only if IPRINT $\geq$ 1)
STOP	1	DP	I	days or revs	Value at which the program is to stop execution (units are determined by ISTOP).
STOPDT	1	DP	--	days or revs	Value at which the next intermediate printout is desired (valid only if IPRINT $\geq$ 1).
TAU	1	DP	I	rad	Independent variable of the PS elements; it is defined such that REVS = TAU/2π.
TWOPI	1	DP	I	rad	2π
X	8	DP	I	Defined by input values (see CONST)	X(1) $\rightarrow$ X(3) = $\vec{X}$ X(4) $\rightarrow$ X(6) = $\vec{V}$ X(7) = time X(8) = total energy

ORIGINAL PAGE IS
OF POOR QUALITY

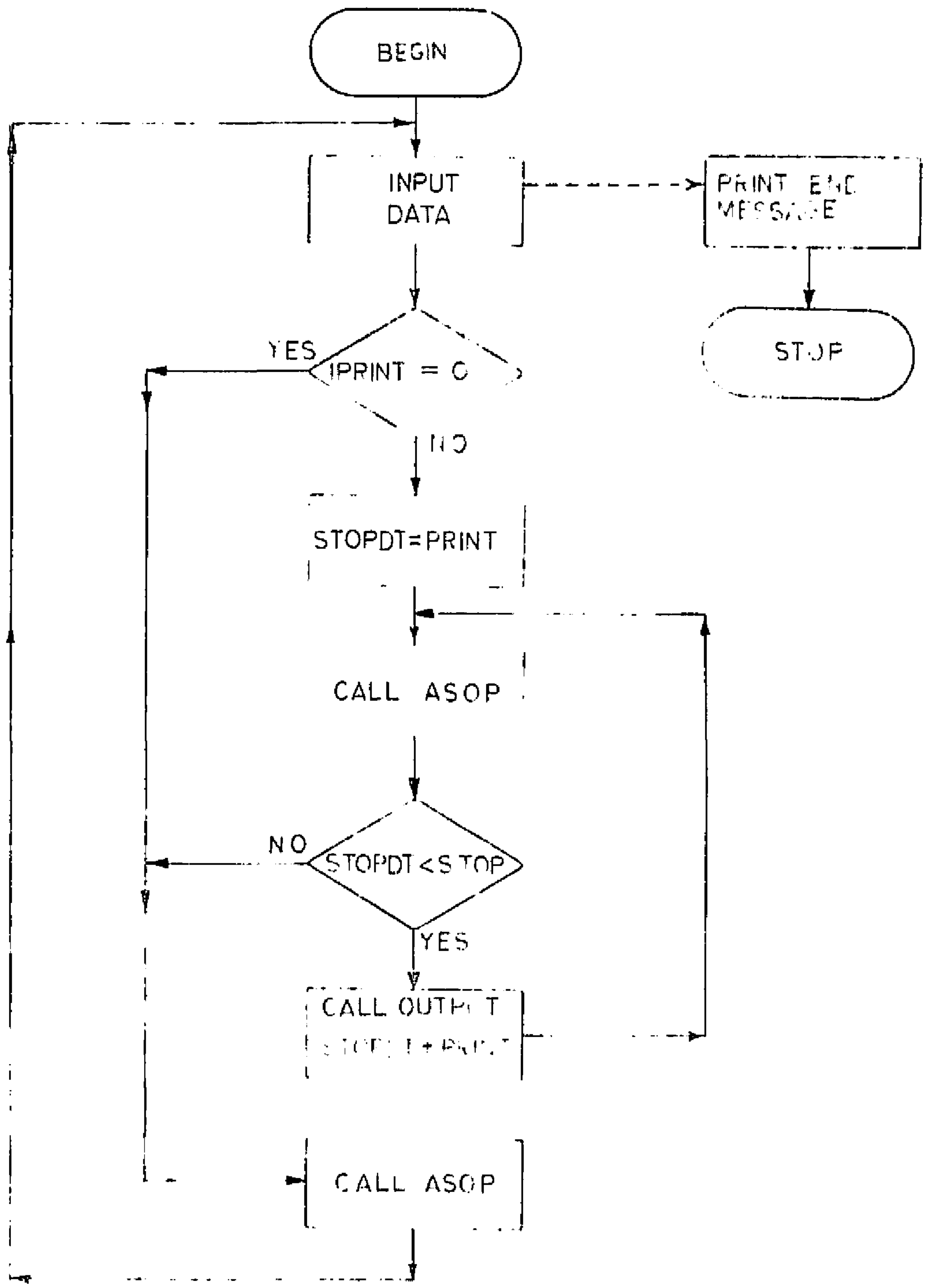


Figure 3.- MAIN program flow charts.

3.3.2 AEIXYZ (Subroutine)

Purpose: Transform the Keplerian elements (a,e,i,ω,Ω,M) into Cartesian coordinates ($\vec{X},\vec{V}$)

Calling sequence: CALL AEIXYZ

Called by: INPUT

Subroutines/functions used: None

Named COMMON:

/CARTC /	X(3),V(3),TIME,ENERGY,R
/KEPLER/	A,E,XI,OMEGA,XNODE,XM
/KONST /	XMU,XMUI,SQTMU,SQTMUI,EPS
/RPOOL /	COMEGA,SOMEGA,CNODE,SNODE,SINC,B(3,3),EA,EAO,SINEA, COSEA,X1(3),V1(3)

Equivalence: (B(3,3),CINC)

Program data: Size = 356_8 (238_{10}) words compiled
execution time = < 1 milliseconds
Subroutine valid only when $|e| < 1.$

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
A	1	DP	I	see appendix A	Semimajor axis of the orbit
B	(3,3)	DP	--	--	Keplerian elements to Cartesian coordinates transformation matrix
CINC	1	DP	--	--	Cosine of the orbital inclination with respect to the Earth's equator (cos i).
CNODE	1	DP	--	--	Cosine of the argument of the ascending node (cos Ω).
COMEGA	1	DP	--	--	Cosine of the argument of pericenter (cos ω).
COSEA	1	DP	--	--	Cosine of the eccentric anomaly (cos E).
E	1	DP	I	--	Orbital eccentricity (e).

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
EA	1	DP	--	rad	The eccentric anomaly of the satellite computed from Kepler's equation.
EAO	1	DP	--	rad	Old value of EA; used when iterating to solve Kepler's equation.
OMEGA	1	DP	I	rad	Argument of pericenter (ω).
R	1	DP	O	see appendix A	Magnitude of the position vector of the satellite.
SINC	1	DP	--	--	Sine of the orbital inclination with respect to the Earth's equator (sin i).
SINEA	1	DP	--	--	Sine of the eccentric anomaly (sin E).
SNODE	1	DP	--	--	Sine of the argument of the ascending node (sin Ω).
SOMEGA	1	DP	--	--	Sine of the argument of pericenter (sin ω).
V	3	DP	O	see appendix A	Velocity vector of the satellite with respect to the Earth's equator V(1) = V_x V(2) = V_y V(3) = V_z
V1	3	DP	--	see appendix A	Velocity vector with respect to the orbital plane V1(1) = V'_x V1(2) = V'_y V1(3) = V'_z = 0
X	3	DP	O	see appendix A	Position vector of the satellite with respect to the Earth's equator X(1) = X X(2) = Y X(3) = Z

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
X1	3	DP	--	see appendix A	Position vector of the satellite with respect to the orbital plane X1(1) = X' X1(2) = Y' X1(3) = Z' = 0
XI	1	DP	I	rad	Orbital inclination to the Earth's equator (i)
XM	1	DP	I	rad	Mean anomaly of the satellite (M)
XMU	1	DP	I	see appendix A	Gravitational constant for the central body (μ)
XNODE	1	DP	I	rad	Argument of the ascending node (Ω)

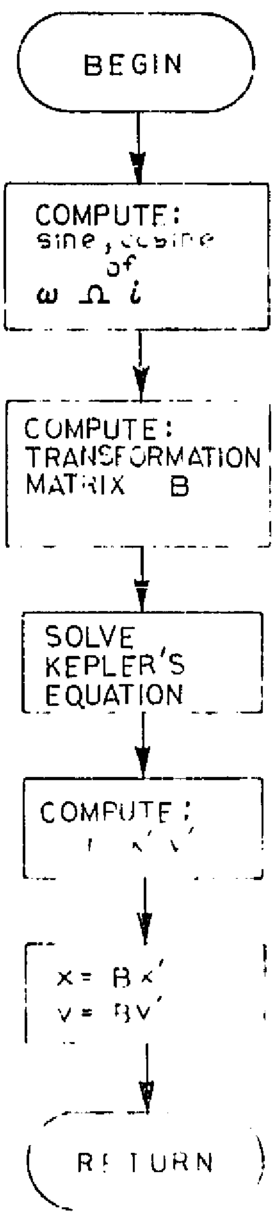


Figure 4.- AEIXYZ flow chart.

3.3.3 ASOP (Subroutine)

Purpose: Driver for the analytical section of the ASOP program; it performs all the operations required by the analytical program PSANS

Calling sequence: CALL ASOP (X,STOP,ISTOP,NEWX)

Called by: MAIN

Subroutines/functions used: CONST, PSANS, PSTOX, TIMEPS, XTOPS

Named COMMON:

/CARTC /	XIN(8),R
/CBASIC/	PI,TWOPI,DEG,RAD,DAY
/PS /	SIG(8),TAU
/PSTIME/	CLO,FAKTPS,XXX

Program data: Size = 127_8 (87_{10}) words compiled
execution time = 2 to 8 milliseconds depending on the value of ISTOP or the eccentricity of the orbit.

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
CLO	1	DP	O	--	Initial value of σ_4 used by the time iteration stopping procedure (TIMEPS).
DAY	1	DP	I	see appendix A	Conversion of days into hours, minutes, or seconds
ISTOP	1	DP	I	--	Flag to determine if the value of STOP is given as days or revolutions. = 1 days = 2 revolutions
NEWX	1	I	I	--	Flag to determine if the ASOP program is to be initialized. = 0 no = 1 yes
SIG	8	DP	--	--	The PS elements $\sigma_1,\sigma_2,\sigma_3,\sigma_4,\rho_1,\rho_2,\rho_3,\rho_4$ where $\rho_1 \rightarrow \rho_4 = \sigma_5 \rightarrow \sigma_8$
STOP	1	DP	I	days or revs	Value at which the final state vector is required; units are set by ISTOP.

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
TAU	1	DP	---	rad	Independent variable of the PS elements.
TWOPI	1	DP	I	rad	2π
X	8	DP	I/O	see appendix A	Initial/final state vector X(1) → X(3) = $\vec{X}$ X(4) → X(6) = $\vec{V}$ X(7) = time X(8) = total energy If initializing NEWX = 1, X(7) and X(8) will be set to 0
XIN	8	DP	I/O	see desription of X	Identical to X but allows the ASOP subroutine to be removed from the stand alone program.

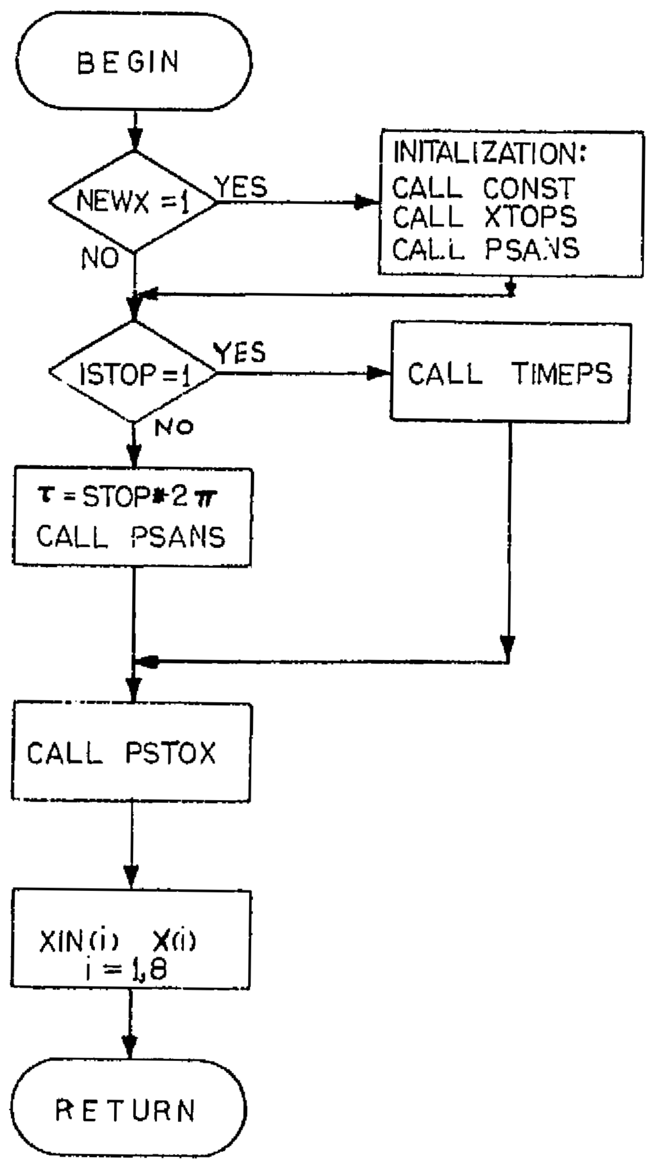


Figure 5.- ASOP flow chart.

ORIGINAL PAGE IS
OF POOR QUALITY

3.3.4 CONST (Subroutine)

Purpose: Initialize the mathematical and physical constants needed to execute the ASOP program.

Calling sequence: CALL CONST

Called by: ASOP, INPUT

Subroutines/functions used: None

Named COMMON:		
	/CBASIC/	PI,TWOPI,DEG,RAD,DAY
	/CPRINT/	X,I(2),IUNITS
	/GEO /	RE,CJ2
	/KONST /	XMU,XMUI,SQTMU,SQTMUI,EPS
	/PERTRB/	IDRAG,ILONG

Program data: Size = 267_8 (784_{10}) words compiled
execution time = < 1 milliseconds

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
CJ2	1	DP	O	--	J_2 coefficient of the central body
DAY	1	DP	O	see appendix A	Value to convert days into seconds, minutes, or hours
DAYS	7	DP	--	see appendix A	Storage array holding the possible values of DAY
DEG	1	DP	O	deg/rad	$180/\pi$
EPS	1	DP	O	--	$3/2 \mu J_2 R_e^2$
ILONG	1	I	I	--	Flag to determine the type of geopotential terms to be used. If ILONG = 0 the two-body orbit is assumed (J_2 = 0)
IUNITS	1	I	I	--	Flag to determine what units are to be used for the calculations. = 1 km,sec = 2 nm,sec = 3 ft,sec = 4 m,sec = 5 km,hr

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
IUNITS (conc'd)					= 6 nm,hr = 7 E.r.,min
PI	1	DP	O	rad	π
RAD	1	DP	O	rad/deg	$\pi/180$
RE	1	DP	O	see appendix A	Central body equatorial radius
RES	7	DP	--	--	Array containing the possible values of RE
SQTMU	1	DP	O	see appendix A	$\sqrt{\mu}$
SQTMUI	1	DP	O	see appendix A	$1/\sqrt{\mu}$
TWOPI	1	DP	O	rad	2π
XMU	1	DP	O	see appendix A	Gravitational constant for the central body (μ)
XMUI	1	DP	O	see appendix A	$1/\mu$
XMUS	7	DP	--	see appendix A	Array containing the possible values of XMU.

ORIGINAL PAGE IS
OF POOR QUALITY

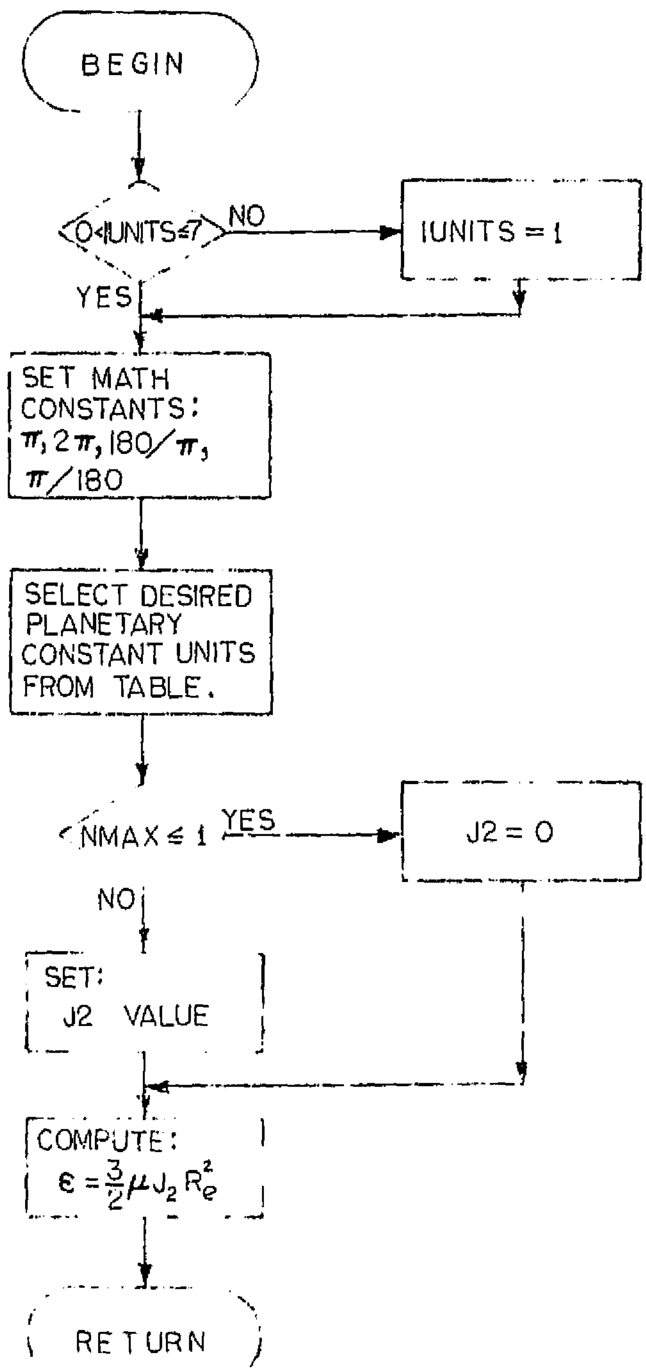


Figure 6.- CONST flow chart.

3.3.5 INPUT (Subroutine)

Purpose: Read the input data from the NAMELIST statement, set the default values, and initialize all required COMMON block variables.

Calling sequence: CALL INPUT ($20)

Called by: MAIN

Subroutines/functions used: AEIXYZ, CONST, OUTPUT, XTOPS

Named COMMON:

/CARTC /	X(6),TIME,ENERGY,R
/CBASIC /	PI,TWOPI,DEG,RAD,DAY
/CPRINT /	PRINT,IPRINT,IELPRT,IUNITS
/DRAG /	CD,AREA,XMASS
/END /	STOP,ISTOP
/EPOCH /	DATE(5),XJDATE
/KEPLER /	EL(6)
/PERTURB/	IDRAG,ILONG
/TESS /	NMAX,MMAX

NAMELIST statements: /INPUT/ EL,IEL,STOP,ISTOP,PRINT,IPRINT,DATE,IDRAG,CD, AREA,XMASS,ITESS,NMAX,MMAX,ILONG,IELPRT,IUNITS

Program data: Size = 301_8 (793_{10}) words compiled

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
AREA	1	DP	I	m^2	Cross sectional surface area of the satellite (valid only if IDRAG $\geq$ 1)
CD	1	DP	I	--	Drag coefficient of the satellite (valid only if IDRAG $\geq$ 1)
DATE[a]	5	DP	I	--	Calendar date of epoch DATE(1) = year (2) = day number (3) = hours (4) = minutes (5) = seconds

[a]To be implemented.

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
EL	6	DP	I/O	see appendix A	Initial conditions of the satellite given in Keplerian elements or Cartesian coordinates; on output it will contain the Keplerian elements EL(1) = X or a (2) = Y or e (3) = Z or i (4) = X or ω (5) = Y or Ω (6) = Z or M
ENERGY	1	DP	O	see appendix A	Total energy of the satellite; Initially set to 0
IDRAG[a]	1	I	I	--	Flag to determine if drag calculations are to be included = 0 no = 1 yes
IEL	1	I	I	--	Flag to determine if the input values of EL are given as Keplerian elements or Cartesian coordinates = 1 Keplerian = 2 Cartesian
IELPRT	1	I	I	--	Flag to determine if the PS elements are to be included with all output = 0 no = 1 yes
ILONG	1	I	I	--	Flag to determine if the long period terms are to be included = 0 no (2 body orbit) = 1 no (J_2 orbit) = 2 yes[a]

[a]To be implemented.

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
IPRINT	1	I	I	--	Flag to deter e if the intermediate printout is to be done at a PRINT value of days or revolutions = 0 no intermediate printout = 1 days = 2 revolutions
ISTOP	1	I	I	--	Flag to determine if the STOP condition is days or revolutions = 1 days = 2 revolutions
IUNITS	1	I	I	--	Flag to determine what calculation constants are to be used = 1 km,sec = 2 nm,sec = 3 ft,sec = 4 m,sec = 5 km,hr = 6 nm,hr = 7 E.r.,min
MMAX[a]	1	I	I	--	Maximum number of tesseral terms to be included (valid only if ILONG $\geq$ 1)
NMAX[a]	1	I	I	--	Maximum number of zonal terms to be included (valid only if ILONG $\geq$ 1)
PRINT	1	DP	I	days or revs	Increment at which the intermediate printout is desired (valid only if IPRINT $\geq$ 1)
RAD	1	DP	I	rad/deg	$\pi/180$
STOP	1	DP	I	days or revs	Final stop value at which output is desired
TIME	1	DP	O	hrs or min or sec	Physical time; initially set to 0

[a]At present only MMAX = 0 and NMAX = 2 are valid inputs.

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
X	6	DP	O	see appendix A	Initial state vector $X(1) \rightarrow X(3) = \vec{X}$ $X(4) \rightarrow X(6) = \vec{V}$
XMASS	1	DP	I	kg	Initial mass of the satellite

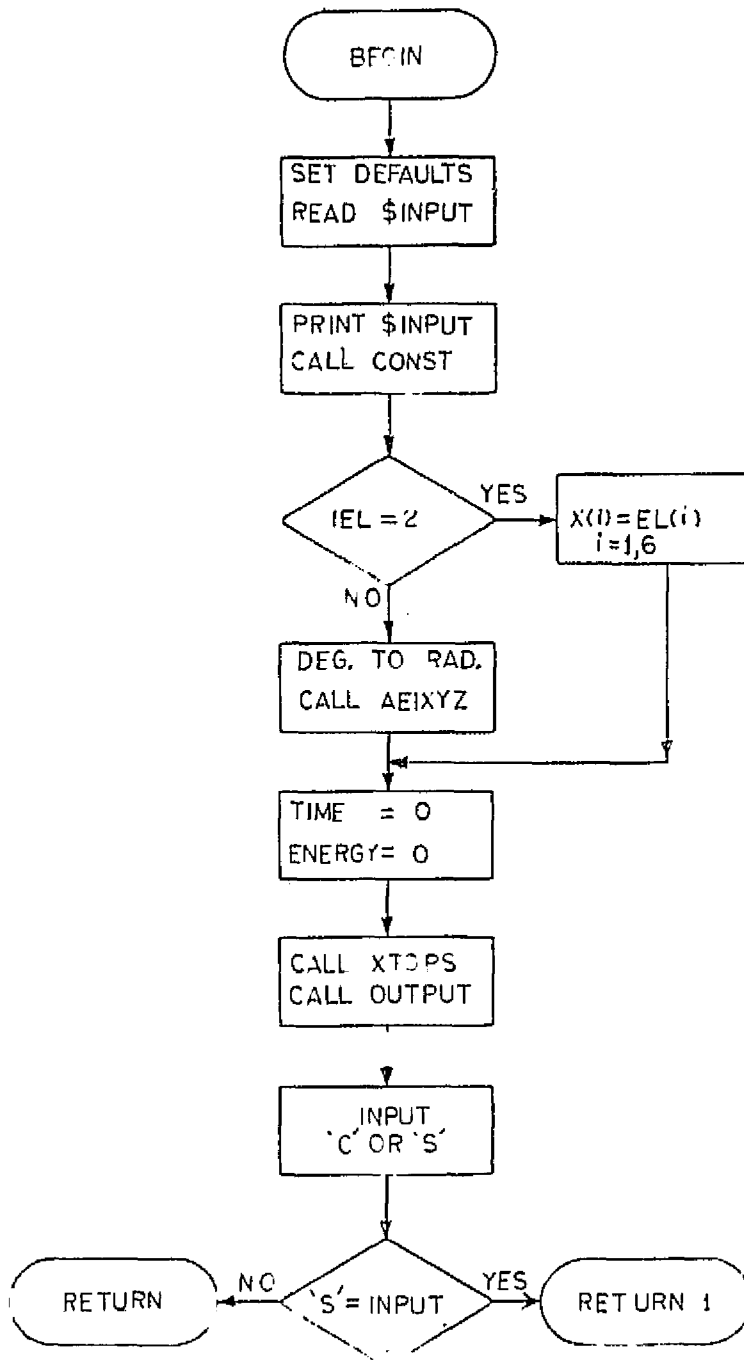


Figure 7.- INPUT flow chart.

3.3.6 OUTPUT (Subroutine)

Purpose: Print all desired output during the execution of the ASOP program; it contains all output formats and does all unit conversions required for output.

Calling sequence: CALL OUTPUT (IFORM)

Called by: INPUT, MAIN

Subroutines/functions used: XYZAEI

Named COMMON:

/CARTC /	X(6),TIME,ENERGY,R
/CBASIC/	PI,TWOPI,DEG,RAD,DAY
/CPRINT/	PRINT,IP,IELPRT,IU
/END /	STOP,ISTOP
/KEPLER/	XKEP(6)
/PS /	SIG(4),RHO(4),TAU

Program data: Size = 573_8 (379_{10}) words compiled.
execution time = 28 milliseconds to 33 milliseconds depending upon the value of IELPRT.

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
AEI	6	DP	--	--	Character array to accompany the Keplerian element output AEI(1) = ' A=' (2) = ' E=' (3) = ' I=' (4) = 'OMEGA=' (5) = 'NODE=' (6) = 'M='
ANG	3	DP	--	--	Character array to accompany any angular output ANG(1) = 'DEG' (2) = 'RAD' (3) = ' '
CHECK	1	DP	--	--	Energy check value = (RHO(4) - ENERGY)/RHO(4)
DAY	1	DP	I	sec/day or min/day or hr/day	Value to convert days into hours, minutes, or seconds.

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
DAYS	1	DP	--	days	Print value (DAYS = TIME/DAY)
DEG	1	DP	I	deg/rad	180/π
DST	7	DP	--	--	Character array to accompany any distance output. DST(1) = 'KM' (2) = 'NM' (3) = 'FT' (4) = 'M ' (5) = 'KM' (6) = 'NM' (7) = 'ER'
ENERGY	1	DP	I	--	Total energy of the physical system.
HMS	4	DP	--	--	Character array to accompany any time output. HMS(1) = 'DAYS:' (2) = ' HRS:' (3) = ' MIN:' (4) = ' SEC:'
IA	3	I	--	--	Character array of blanks and asterisks (*) IA(1) = '**' (2) = ' ' (3) = '**'
IELPRT	1	I	I	--	Flag to determine if the PS elements are to be printed = 0 no = 1 yes
IFORM	1	I	I	--	Flag to determine if the initial or final condition message is to be printed =1 initial condition message =2 no message (intermediate print) =3 final condition message

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
IP	1	I	I	--	Flag to determine what print condition is being used = 1 days = 2 revolutions
ISTOP	1	I	I	--	Flag to determine the final stop condition = 1 days = 2 revolutions
IU	1	I	I	--	Pointer to the VEL and DST arrays
IXP	1	I	--	--	Pointer to the IA array IFORM $\neq$ 3 $\rightarrow$ IXP = IP IFORM = 3 $\rightarrow$ IXP = ISTOP
REVS	1	DP	--	revs	Total number of revolutions predicted = TAU/2π
RHO	4	DP	I	see	PS elements $\rho_1 \rightarrow \rho_4$ RHO(1) = ρ_1 RHO(3) = ρ_3 RHO(2) = ρ_2 RHO(4) = ρ_4
SIG	4	DP	I	see	PS elements $\sigma_1 \rightarrow \sigma_4$ SIG(1) = σ_1 SIG(3) = σ_3 SIG(2) = σ_2 SIG(4) = σ_4
TAU	1	DP	I	rad	Independent variable of the PS elements set
TIME	1	DP	I	hrs or min or sec	Physical time
TWOPI	1	DP	I	rad	2π

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
VEL	7	DP	--	--	Character array to accompany any velocity output VEL(1) = 'KM/S' (2) = 'NM/S' (3) = 'FT/S' (4) = 'M/S ' (5) = 'KM/H' (6) = 'NM/H' (7) = 'ER/M'
X	6	DP	I	see appendix A	Cartesian state vector $X(1) \rightarrow X(3) = \vec{X}$ $X(4) \rightarrow X(6) = \vec{V}$
XKEP	6	DP	I/O	see appendix A	Keplerian elements XKEP(1) = a (2) = e (3) = i (4) = ω (5) = Ω (6) = M
XYZ	6	DP	--	--	Character array to accompany the Cartesian state vector X XYZ(1) = ' X=' (2) = ' Y=' (3) = ' Z=' (4) = 'VX=' (5) = 'VY=' (6) = 'VZ='

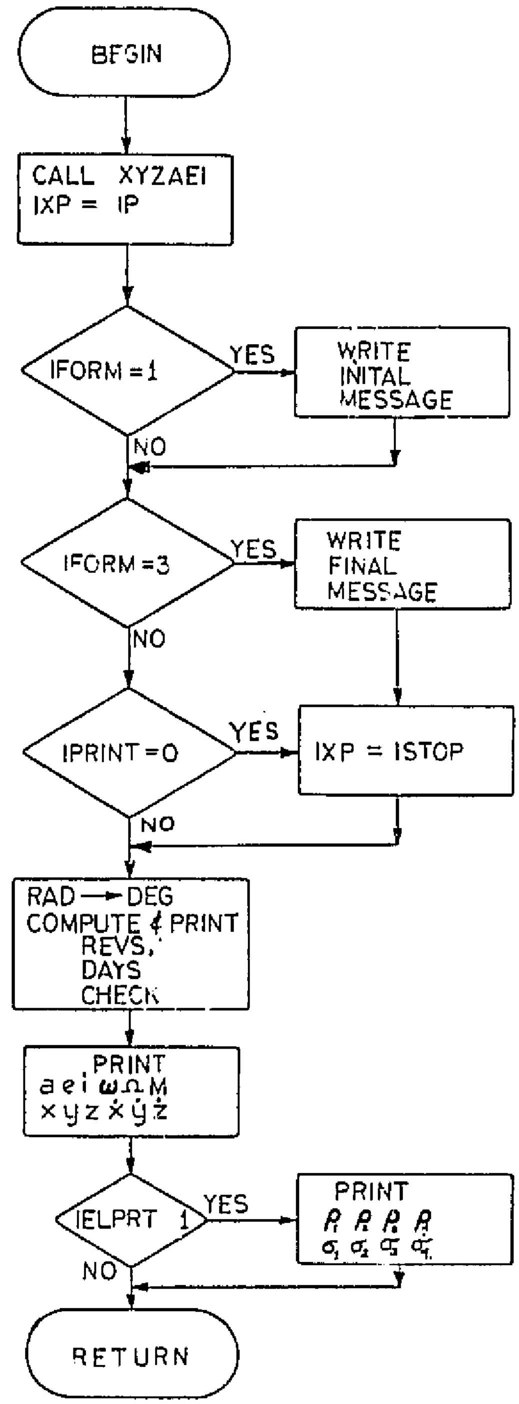


Figure 8.- OUTPUT flow chart.

3.3.7 POT J2 (Subroutine)

Purpose: Compute the potential due to the J_2 term of the central body geopotential

Calling sequence: CALL POTJ2 (POT)

Called by: PSTOX, XTOPS

Subroutines/functions used: None

Named COMMON: /CARTC/ X(8),R
/KONST/ XMU,XMUI,SQTMU,SQTMUI,EPS
/RPOOL/ X32,RI,R2I,R3I,XXX(20)

Program data: Size = 41_8 (33_{10}) words compiled
execution time = <1 milliseconds

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
EPS	1	DP	I	see appendix A	$= 3/2\ \mu\ J_2\ R_e^2$
POT	1	DP	O	--	Potential due to J_2
R	1	DP	I	see appendix A	Magnitude of the position vector
R2I	1	DP	--	--	$1/R^2$
R3I	1	DP	--	--	$1/R^3$
RI	1	DP	--	--	1/R
X	8	DP	I	see appendix A	Cartesian state vector $X(1) \rightarrow X(3) = \vec{X}$ $X(4) \rightarrow X(6) = \vec{V}$ X(7) = time X(8) = total energy
X32	1	DP	--	--	$= Z^2 = X(3)^2$

ORIGINAL PAGE IS
OF POOR QUALITY

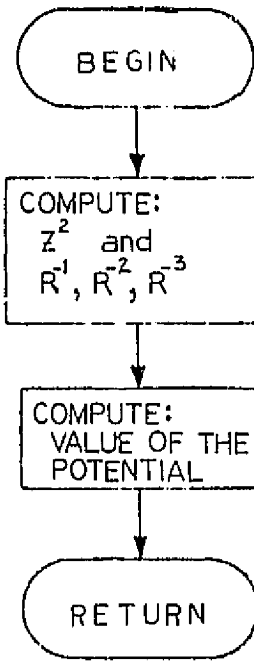


Figure 9.- POTJ2 flow chart.

3.3.8 PSANS (Subroutine)

Purpose: Analytical theory of the ASOP program; it contains only the short period and the first order secular terms.

Calling sequence: CALL PSANS (NN)

Called by: ASOP, TIMEPS

Subroutines/functions used: None

Named COMMON:		
	/CBASIC/	PI,TWOPI,DEG,RAD,DAY
	/IPOOL /	IMARK
	/KONST /	XMU,XMUI,SQTMU,SQTMUI,EPS
	/PS /	SIG(4),RHO(4),TAU
	/PSANSV/	FACTOR(4),SIGINI(8)
	/PSTIME/	CLO,FAKTPS,YYY

Equivalences: (LS,SIG(4)), (L,RHO(4)), (PHI,RHO(1)), (DSF(1),W(1)), (DSB(1),W(5)), (S1(1),GAM3(1)), (Y(1),GAM2(1)), (Q(1),GAM(1)), (HC(1),DEL3(1)), (GC(1),DEL2(1)), (P(1),DEL(1))

Program data: Size = 1770_8 (1016_{10}) words compiled
execution time = 1 millisecond

A description of the mathematical symbols used and their relationship to one another is given in reference 6; a brief description can also be found in Appendixes E and F. Therefore, only a brief mathematical description will be given.

FORTRAN variable	Dimension	Type	Input/ output	Description
BS	1	DP	--	1 - H/G
BY3	1	DP	--	1/3
BY6	1	DP	--	1/6
C	8	DP	--	$\partial c/\partial\sigma_k, \partial c/\partial\rho_k$ k = 1,2,3,4
CN	1	DP	--	±1 depending on the value of NN
COSFC2	1	DP	--	cos (A_{21}) (see appendix F)
COSFC3	1	DP	--	cos (A_{31}) (see appendix F)
CS	1	DP	--	$1/2(G + H)(\rho_3^2 + \sigma_3^2)$ = small 'c'
DEL1	8	DP	--	$\partial\delta_1/\partial\sigma_k, \partial\delta_1/\partial\rho_k$ k = 1,2,3,4

FORTRAN variable	Dimension	Type	Input/ output	Description
DEL2	8	DP	--	$\partial\delta_2/\partial\sigma_k$, $\partial\delta_2/\partial\rho_k$ $k = 1,2,3,4$
DEL3	8	DP	--	$\partial\delta_3/\partial\sigma_k$, $\partial\delta_3/\partial\rho_k$ $k = 1,2,3,4$
DELTA1	1	DP	--	δ_1
DELTA2	1	DP	--	δ_2
DELTA3	1	DP	--	δ_3
DIFF2	1	DP	--	$\rho_2^2 - \sigma_2^2$
DIFF3	1	DP	--	$\rho_3^2 - \sigma_3^2$
DSB	4	DP	--	$\partial b/\partial\beta_k$ $k = 1,2,3,4$
DSF	4	DP	--	$\partial f/\partial\beta_k$ $k = 1,2,3,4$
EPS	1	DP	I	$\varepsilon = 3/2\ (\mu\ J_2\ R_e^2)$
ETA1	1	DP	--	$\sin \sigma_1$
ETA2	1	DP	--	$\sin 2\sigma_1$
ETA3	1	DP	--	$\sin 3\sigma_1$
FACTOR	4	DP	--	Derivatives of the DS Hamiltonian and its combinations (A_1, A_2, A_3 and A_4 in appendix F)
FAK	1	DP	--	$(2L)^{-3/2}$
FAKTPS	1	DP	--	Derivative of the DS Hamiltonian (A_4 in appendix F)
FS	1	DP	--	f
FSSQ	1	DP	--	f^2
G	1	DP	--	G
GAM1	8	DP	--	$\partial\gamma_1/\partial\sigma_k$, $\partial\gamma_1/\partial\rho_k$ $k = 1,2,3,4$
GAM2	8	DP	--	$\partial\gamma 2/\partial\sigma k$, $\partial\gamma 2/\partial\rho_k$ $k = 1,2,3,4$
GAM3	8	DP	--	$\partial\gamma_3/\partial\sigma_k$, $\partial\gamma_3/\partial\rho_k$ $k = 1,2,3,4$
GAMMA1	1	DP	--	γ_1

FORTRAN variable	Dimension	Type	Input/ output	Description
GAMMA2	1	DP	--	γ_2
GAMMA3	1	DP	--	γ_3
GC	8	DP	--	$\partial G/\partial \sigma_k$, $\partial G/\partial \rho_k$ k = 1,2,3,4
GIN	1	DP	--	G^{-1}
GINSQ	1	DP	--	G^{-2}
GM3H	1	DP	--	G - 3H
GPH	1	DP	--	G + H
GSQ	1	DP	--	G^2
H	1	DP	--	H
HC	8	DP	--	$\partial H/\partial \sigma_k$, $\partial H/\partial \rho_k$ k = 1,2,3,4
HOG	1	DP	--	H/G
HSQ	1	DP	--	H^2
IMARK	1	I	--	Flag determining if one or two passes have been made 0 = 1 pass; 1 = 2 passes
IQL	1	DP	--	$\mu/\sqrt{2L}$
L	1	DP	--	$\rho_4 = \sigma_8 = L$
LS	1	DP	--	$\sigma_4 = \ell$
NN	1	I	--	Flag determining if initializing or computing 0 = initializing; 1 = computing
P	8	DP	--	$\partial p/\partial \sigma_k$, $\partial p/\partial \rho_k$ k = 1,2,3,4
P5	1	DP	--	1/2
PHI	1	DP	--	$\phi = \rho_1 = \sigma_5$
PSSQRT	1	DP	--	$\sqrt{PS}$
Q	8	DP	--	$\partial q/\partial \sigma_k$, $\partial q/\partial \rho_k$ k = 1,2,3,4

FORTRAN variable	Dimension	Type	Input/ output	Description
QC	1	DP	--	Q
QCIN	1	DP	--	Q^{-1}
QCSQ	1	DP	--	Q^2
QCV	8	DP	--	$\partial Q/\partial \sigma_k$, $\partial Q/\partial \rho_k$ k = 1,2,3,4
QS	1	DP	--	q
RHO	4	DP	I/O	$\rho_1,\ldots,\rho_4$ (see SIG)
S	8	DP	--	$\partial s/\partial \sigma_k$, $\partial s/\partial \rho_k$ k = 1,2,3,4
S1	4	DP	--	Derivatives of the generating function S_1
SIG	4	DP	--	$\sigma_1,\ldots,\sigma_4$; note the location of SIG and RHO in COMMON. This location makes the equivalence $\rho_1 = \sigma_5,\ldots,\rho_4 = \sigma_8$
SIGINI	8	DP	O	The initial values of the σ's and ρ's
SINFC2	1	DP	--	sin $(A_2\tau)$ (see appendix F)
SINFC3	1	DP	--	sin $(A_3\tau)$ (see appendix F)
SQTMU	1	DP	I	$\sqrt{\mu}$
SQTMUI	1	DP	I	$\mu^{-1/2}$
SS	1	DP	--	s
SUM2	1	DP	--	$1/2\ (\sigma_2^{\ 2} + \rho_2^{\ 2})$
SUM3	1	DP	--	$1/2\ (\sigma_3^{\ 2} + \rho_3^{\ 2})$
TAU	1	DP	I/O	independent variable τ
TWO3	1	DP	--	2/3
W	8	DP	--	$\partial w/\partial \sigma_k$, $\partial w/\partial \rho_k$ k = 1,2,3,4
WS	1	DP	--	w
XMU	1	DP	I	μ

FORTRAN variable	Dimension	Type	Input/ output	Description
XMUI	1	DP	I	μ^{-1}
Y	8	DP	--	$\partial y/\partial \sigma_k$, $\partial y/\partial \rho_k$ $k = 1,2,3,4$
YS	1	DP	--	y
ZET1	1	DP	--	$\cos \sigma_1$
ZET2	1	DP	--	$\cos 2\sigma_1$
ZET3	1	DP	--	$\cos 3\sigma_1$

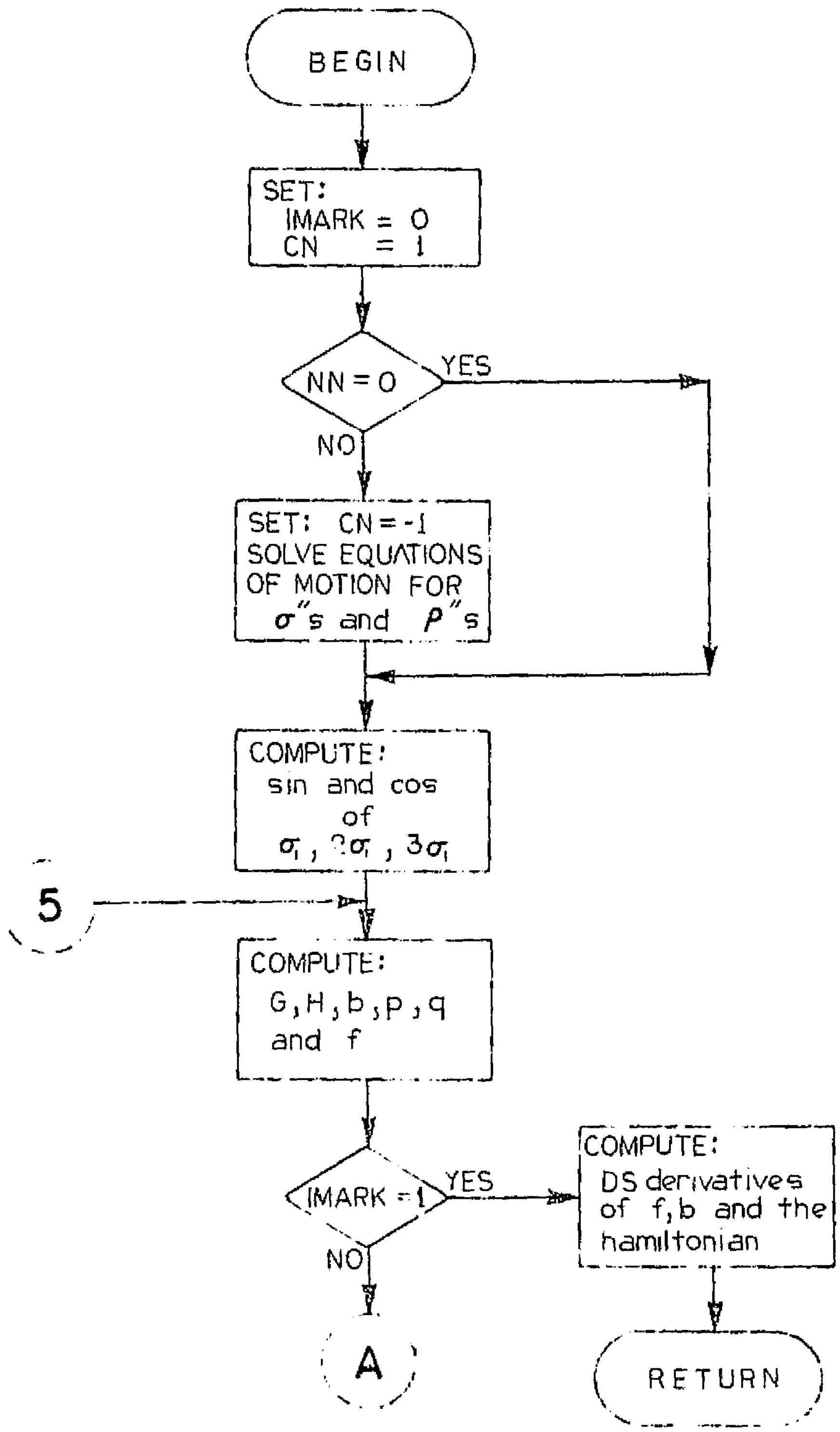


Figure 10.- PSANS flow chart.

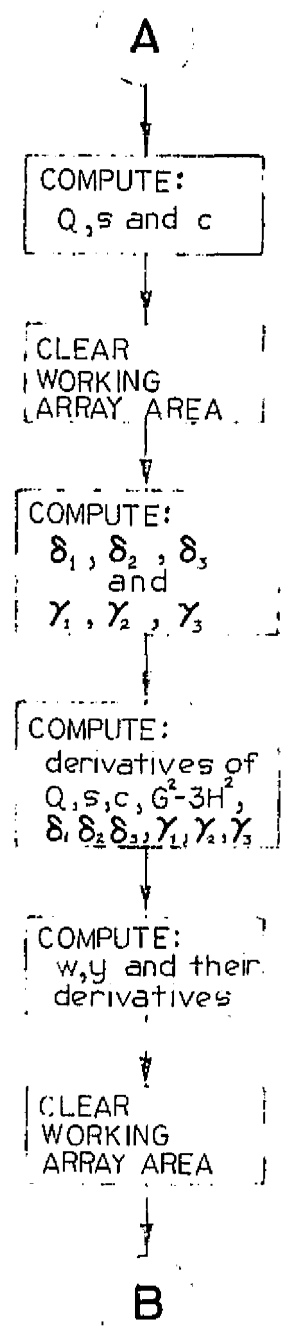


Figure 10.- Continued.

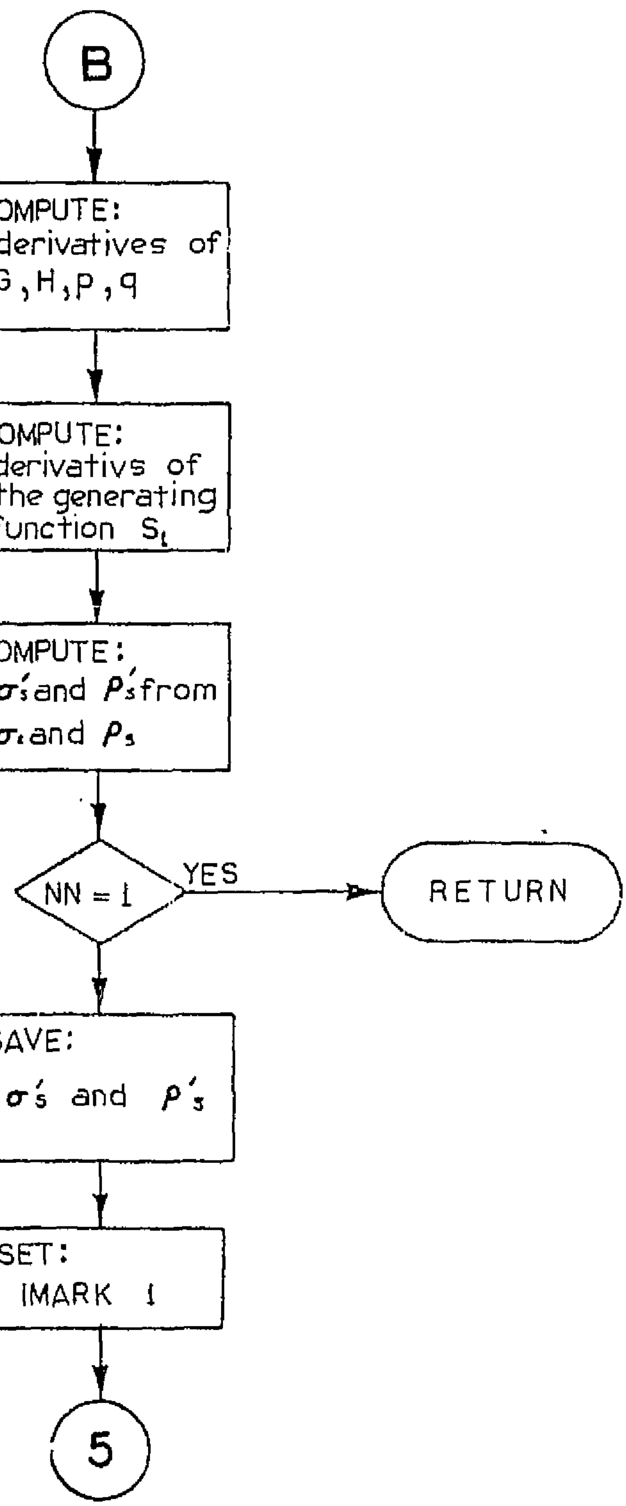


Figure 10.- Concluded.

3.3.9 PSTOX (Subroutine)

Purpose: Transform the PS (Poincaré-Similar) elements into the Cartesian coordinates $(\vec{X},\vec{V})$. The subroutines will also compute the physical time for the time iteration stopping procedure.

Calling sequence: CALL PSTOX (ITIME)

Called by: ASOP

Subroutines/functions used: POTJ2

Named COMMON:

/CARTC /	X1,X2,X3,V1,V2,V3,TIME,ENERGY,R
/KONST /	XMU,XMUI,SQTMU,SQTMUI,EPS
/PS /	SIG(4),RHO(4),TAU
/PSTIME/	CLO,FAKPS,Q
/RPOOL /	SUM2,T2L,SQT2LI,CS1G1,SS1G1,QCAP,ECOSPH,ESINPH,P, ROP,EROOT,EMINPH,FAKT,GC,GCIN,RCAP,SUM3,X3ROOT,RDOT, RCAPDT,XXX(4)

Program data: Size = 363_8 (243_{10}) words compiled
execution time = <1 millisecond

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
CS1G1	1	DP	--	--	$\cos \sigma_1$
ECOSPH	1	DP	--	--	$E \cos \phi$
EMINPH	1	DP	--	--	$E - \phi$
ENERGY	1	DP	O	--	total energy of the system
EROOT	1	DP	--	--	$\sqrt{2\rho_4 \; p/\mu}$
ESINPH	1	DP	--	--	$E \sin \phi$
FAKT	1	DP	--	--	$(2L)^{-3/2}$
GC	1	DP	--	--	$\rho_1 - 1/2\ (\sigma_2^2 + \rho_2^2) = G$
GCIN	1	DP	--	--	G^{-1}
ITIME	1	I	I	--	Flag determining if only the physical time is to be computed 0 = no; 1 = yes

ORIGINAL PAGE IS
OF POOR QUALITY

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
P	1	DP	--	dist.	p (see section 2.2)
P5	1	DP	--	--	1/2
POT	1	DP	I	--	Value of the Earth's potential due to J_2
Q	1	DP	O	--	q (see section 2.2)
QCAP	1	DP	--	--	Q (see section 2.2)
R	1	DP	O	see appendix A	Magnitude of the position vector
RCAP	1	DP	--	--	R (see section 2.2)
RCAPDT	1	DP	--	--	$\partial R/\partial t$ (see section 2.2)
RDOT	1	DP	--	--	Magnitude of the velocity vector
RHO	4	DP	I	--	$\rho_1,\ldots,\rho_4$
RI	1	DP	--	see appendix A	Inverse magnitude of the position vector
ROP	1	DP	--	--	r/P (see section 2.2)
SIG	4	DP	I	--	$\sigma_1,\ldots,\sigma_4$
SQT2LI	1	DP	--	--	$\mu/\sqrt{2L}$ (see section 2.2)
SSIG1	1	DP	--	--	$\sin \sigma_1$
SUM2	1	DP	--	--	$1/2\ (\sigma_2^2 + \rho_2^2)$
SUM3	1	DP	--	--	$\sigma_3^2 + \rho_3^2$
T2L	1	DP	--	--	$2\rho_4$
TIME	1	DP	O	time	t
V1	1	DP	O	see appendix A	Components of the velocity vector V1 = X component V2 = Y component V3 = Z component
V2	1	DP	O		
V3	1	DP	O		

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
X1	1	DP	O	see appendix A	Components of the position vector X1 = X component X2 = Y component X3 = Z component
X2	1	DP	O		
X3	1	DP	O		
X3ROOT	1	DP	--	--	$\sqrt{4G - \sigma_3^2 - \rho_3^2/G}$ (see section 2.2)
XMU	1	DP	I	see appendix A	μ
XMUI	I	DP	I	see appendix A	μ^{-1}

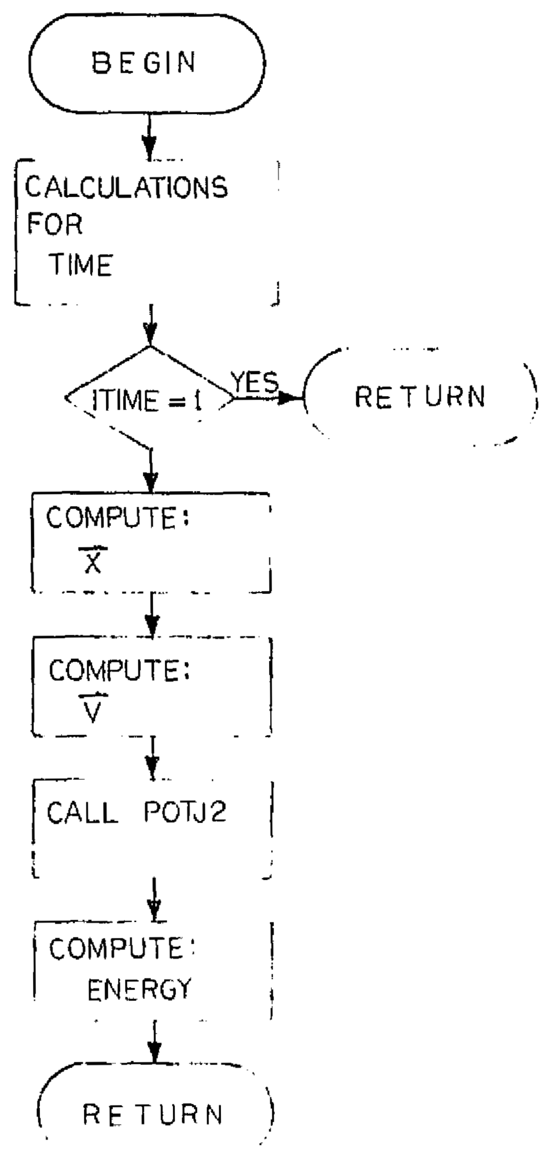


Figure 11.- PSTOX flow chart.

3.3.10 TIMEPS (Subroutine)

Purpose: Iteration procedure to stop the PS elements at a desired value of the physical time.

Calling sequence: CALL TIMEPS (TFIN)

Called by: ASOP

Subroutines/functions used: PSANS,PSTOX

Named COMMON:

/CARTC /	X(6),TIME,ENERGY,R
/CBASIC/	PI,TWOPI,DEG,RAD,DAY
/KONST /	XMU,XMUI,SQTMU,SQTMUI,EPS
/PS /	SIG(8),TAU
/PSTIME/	CLO,FAKT,QS

Program data: Size = 202_8 (130_{10}) words compiled
execution time = 4 to 9 milliseconds depending upon the orbital eccentricity.

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
CLO	1	DP	I	time	Initial value of σ_4
DAY	1	DP	I	sec/day or min/day or hr/day	Value to convert days into hours, minutes, or seconds
DAYS	1	DP	--	days	Total number of days elapsed
FAKT	1	DP	I	--	FAKTPS from PSANS = A_4 (see appendix F)
IERR	1	I	--	--	Error counter
ITER	1	I	--	--	Total number of iterations allowed
QS	1	DP	I	--	q (see section 2.2)
R	1	DP	I	see appendix A	Magnitude of the position vector
TAU		DP	O	rad	New value of the independent variable (τ)
TFIN	1	DP	I	sec or min or hrs	Final time desired for stopping the iteration

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
TIME	1	DP	I	sec or min or hrs	Computed value of the physical time
TOL	1	DP	--	time	Allowable TOLerance between TFIN and TIME that will stop the iteration $\lvert TFIN - TIME \rvert \leq TOL$
TWOPI	1	DP	I	rad	2π

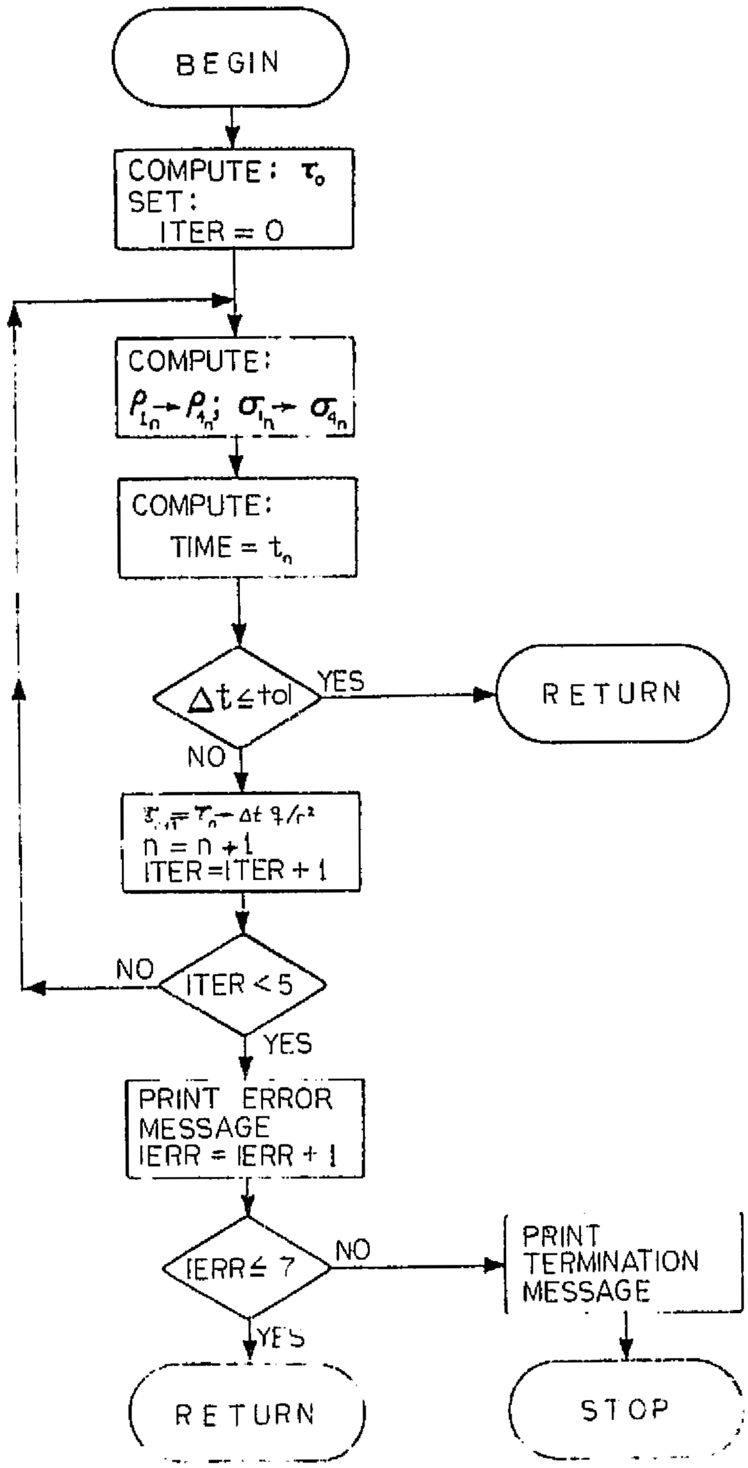


Figure 12.- TIMEPS flow chart.

3.3.11 XTOPS (Subroutine)

Purpose: Transform the Cartesian coordinates $(\vec{X},\vec{V})$ into the PS (Poincaré-Similar) elements $(\vec{\sigma},\vec{\rho})$

Calling sequence: CALL XTOPS

Called by: ASOP, INPUT

Subroutines/functions used: POTJ2

Named COMMON:

/CARTC/	X1,X2,X3,V1,V2,V3,TIME,ENERGY,R
/KONST/	XMU,XMUI,SQTMU,SQTMUI,EPS
/PS /	SIG(4),RHO(4),TAU
/RPOOL/	G1SQ,G2SQ,SQT2LI,ESINPH,EROOT,EMINPH,SQTGHI,GCIN,GCSQT, SUM3,RCAP,COS1G1,SIS1G1,SUM2,P,Q,QCAP,RDOT,ZCAP1,ZCAP2, ECOSPH,XXX(3)

Equivalence: (HC,G3), (PHIC,RHO(1)), (LC,RHO(4))

Program data: Size = 523_8 (339_{10}) words compiled
execution time = <1 milliseconds

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
					All equations are described in section 2.1
COSIG1	1	DP	--	--	$\cos \sigma_1$
ECOSPH	1	DP	--	rad	$E \cos \phi$
EMINPH	1	DP	--	rad	$E - \Phi$
EROOT	1	DP	--	rad	$\sqrt{1 - 2Q(\Phi - G)}$
ESINPH	1	DP	--	rad	$E \sin \phi$
G1	1	DP	--	dist²/time	$YV_z - ZV_y = G_x$
G1SQ	1	DP	--	--	G_x^2
G2	1	DP	--	dist²/time	$ZV_x - XV_z = G_y$
G2SQ	1	DP	--	--	G_y^2
G3	1	DP	--	dist²/time	$XV_y - YV_x = G_z$

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
GC	1	DP	--	dist^2/time	$G = \sqrt{G_x^2 + G_y^2 + G_z^2}$
GCIN	1	DP	--	time/dist^2	1/G
GCSQ	1	DP	--	--	G^2
GCSQT	1	DP	--	--	$\sqrt{G}$
HC	1	DP	--	--	$H = G_3$
LC	1	DP	O	(dist/time)^2	$L = \rho_4 = \sigma_8$
P	1	DP	--	dist	p (small p)
PHIC	1	DP	O	dist^2/time	$\Phi = \rho_1 = \sigma_5$
POT	1	DP	I	--	Potential due to J_2
Q	1	DP	--	--	$1/2 \left(\frac{\mu}{2L} - G\right)$
QCAP	1	DP	--	--	Q
R	1	DP	O	see appendix A	The magnitude of the position vector (r)
RCAP	1	DP	--	--	R
RDOT	1	DP	--	--	$\partial R/\partial t$
RHO	4	DP	O	--	$\rho_1 \rightarrow \rho_4$ in RHO(1) → RHO(4)
RI	1	DP	--	--	Inverse magnitude of the position vector magnitude (1/r)
RSQ	1	DP	--	--	Square of the position vector (r^2)
SIG	4	DP	O	see ref. 1	$\sigma_1 \rightarrow \sigma_4$ in SIG(1) → SIG(4)
SIS1G1	1	DP	--	--	$\sin \sigma_1$
SQT2LI	1	DP	--	--	$\mu/\sqrt{2L}$
SQTGHI	1	DP	--	--	$-\sqrt{2/(G + H)}$

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
SUM2	1	DP	--	$dist^2/time$	Φ - G
SUM3	1	DP	--	$dist^2/time$	2(G - H)
TAU	1	DP	O	rad	Independent variable; initially set to zero (0)
TIME	1	DP	I	time	Initial physical time
V1	1	DP	O	see appendix A	Components of the velocity vector V1 = X component V2 = Y component V3 = Z component
V2	1	DP	O		
V3	1	DP	O		
X1	1	DP	O	see appendix A	Components of the position vector X1 = X component X2 = Y component X3 = Z component
X2	1	DP	O		
X3	1	DP	O		
XMU	1	DP	I	see appendix A	μ
XMUI	1	DP	I		$1/\mu$
ZCAP1	1	DP	--	--	Z_1
ZCAP2	1	DP	--	--	Z_2

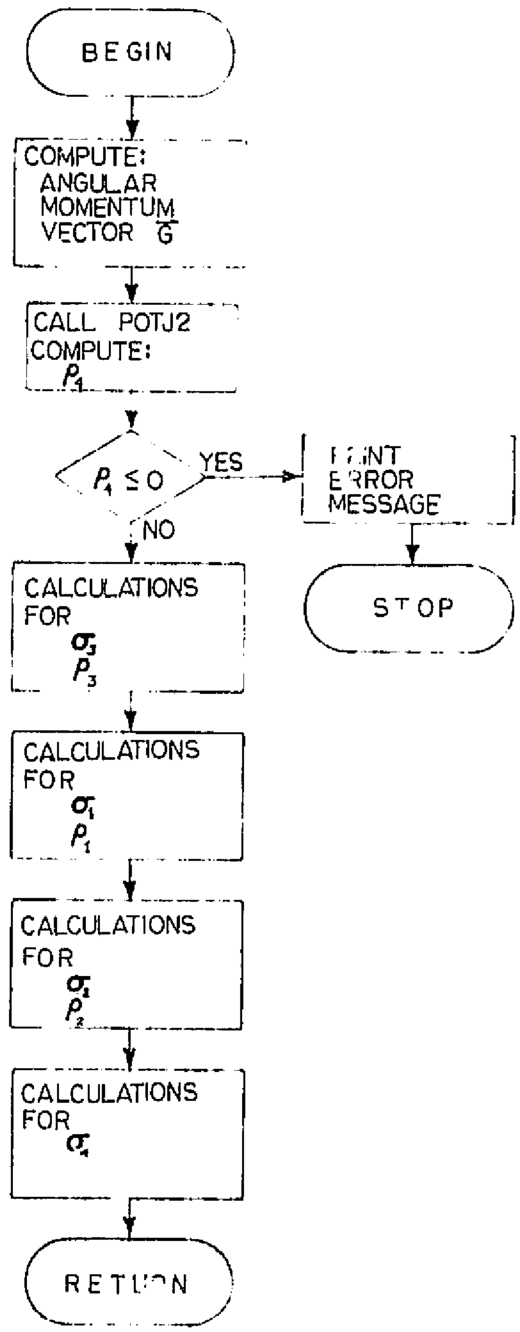


Figure 13.- XTOPS flow chart.

ORIGINAL PAGE IS
OF POOR QUALITY

3.3.12 XYZAEI (Subroutine)

Purpose: Transform the Cartesian coordinates $(\vec{X},\vec{V})$ into the Keplerian elements (a,e,i,ω,Ω,M)

Calling sequence: CALL XYZAEI

Called by: OUTPUT

Subroutines/functions used: None

Named COMMON:

/CARTC /	X1,X2,X3,V1,V2,V3,TIME,ENERGY,R
/CBASIC/	PI,TWOPI,DEG,RAD,DAY
/KEPLER/	A,E,XI,OMEGA,XNODE,XM
/KONST /	XMU,XMUI,SQTMU,SQTMUI,EPS
/RPOOL /	VSQ,RRDOT,ECOSE,ESINE,P,H,WX,WY,WZ,EA,RCOSF,RSINF, COSWF,SINWF,TEMP,RCOSL,RSINL,XXX(7)

Program data: Size ≈ 406_8 (262_{10}) words compiled
execution time = <1 milliseconds
Considers only elliptic motion

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
A	1	DP	O	see appendix A	Semimajor axis of the orbit
COSWF	1	DP	--	--	Cos $(\omega + f)$ ω = argument of perigee f = true anomaly
E	1	DP	O	--	Orbit eccentricity (e)
EA	1	DP	--	rad	Eccentric anomaly (E)
ECOSE	1	DP	--	rad	E cos e
ESINE	1	DP	--	rad	E sin e
H	1	DP	--	$dist^2/time$	Total angular momentum
OMEGA	1	DP	O	rad	Argument of pericenter (ω)
P	1	DP	--	--	$1 - e^2$
R	1	DP	O	see appendix A	Magnitude of the position vector

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
RCOSF	1	DP	--	same as R	r cos f f = true anomaly r = R
RCOSL	1	DP	--	same as R	r cos L L = mean anomaly r = R
RRDOT	1	DP	--	see appendix A	$\vec{X} \cdot \vec{V}$
RSINF	1	DP	--	same as R	r sin f r = R f = true anomaly
RSINL	1	DP	--	same as R	r sin L r = R L = mean anomaly
SINWF	1	DP	--	--	sin (ω + f) ω = argument of perigee f = true anomaly
V1	1	DP	I	see appendix A	Components of the velocity vector V1 = X component V2 = Y component V3 = Z component
V2	1	DP	I		
V3	1	DP	I		
VSQ	1	DP	--	see appendix A	Magnitude of the velocity vector, squared, (V^2)
WX	1	DP	--	$dist^2$/time	Components of the total angular momentum WX = X component WY = Y component WZ = Z component
WY	1	DP	--		
WZ	1	DP	--		

ORIGINAL PAGE IS
OF POOR QUALITY

FORTRAN variable	Dimension	Type	Input/ output	Units	Description
X1	1	DP	I	see appendix A	Components of the position vector X1 = X component X2 = Y component X3 = Z component
X2	1	DP	I		
X3	1	DP	I		
XM	1	DP	O	rad	Mean anomaly
XMU	1	DP	I	see appendix A	Central body gravitational constant (μ).
XNODE	1	DP	O	rad	Argument of the ascending node (Ω)

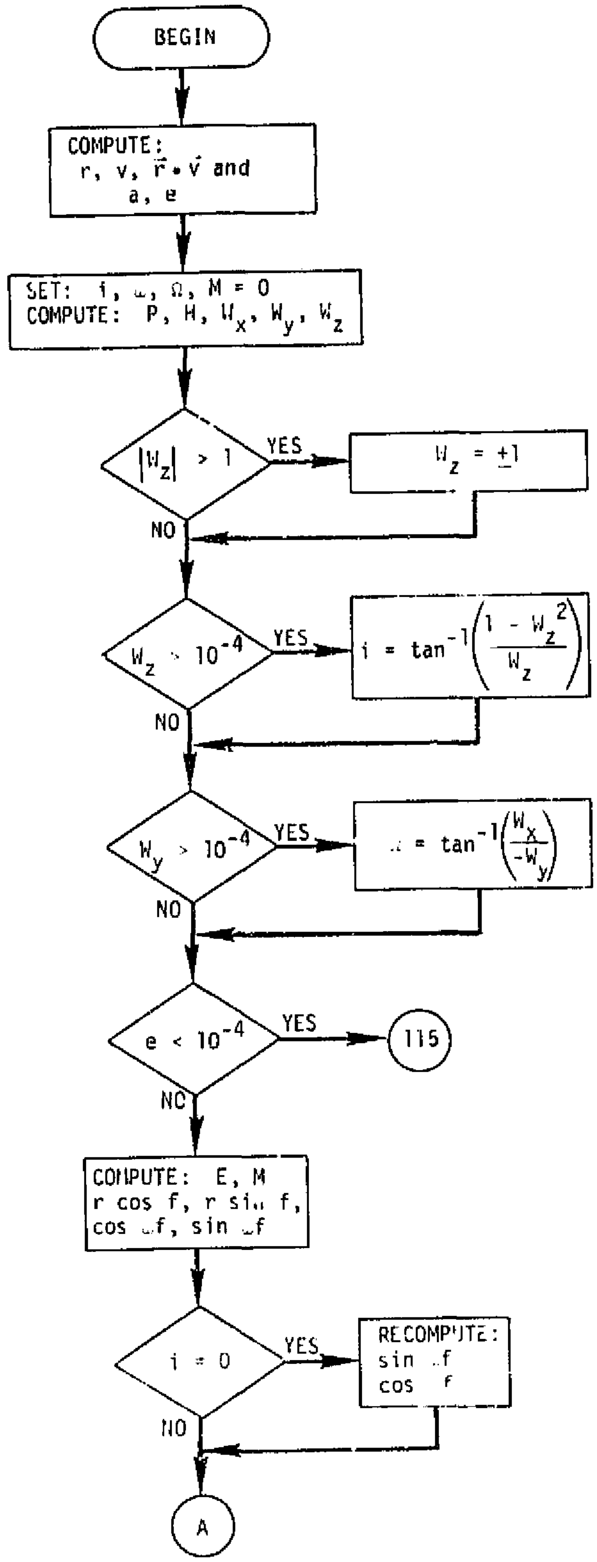


Figure 14.- XYZAEI flow chart.

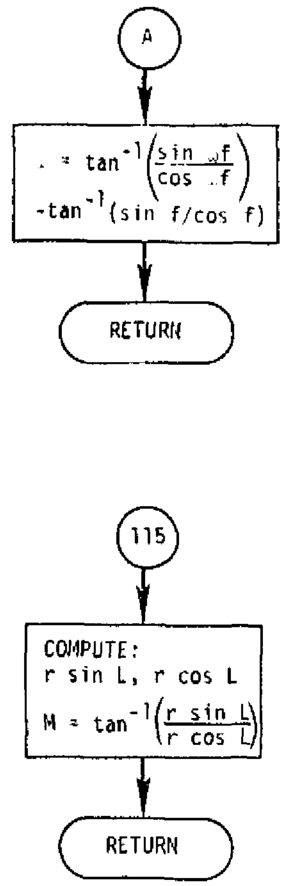


Figure 14.- Concluded.

3.4 LABELED COMMON:

Notation:
- R - Real variable } Type
- I - Integer variable }
- S - Single precision } Precision
- D - Double precision }

/CARTC/

Location	Name	Dimension	Type/ precision	Description
1-3	X	3	R/D	Cartesian coordinates for the position of the satellite; X,Y,Z
4-6	V	3	R/D	Velocity vector of the satellite; V_x, V_y, V_z
7	TIME	1	R/D	Elapsed time (hr, min, or sec)
8	ENERGY	1	R/D	Total orbital energy
9	R	1	R/D	Magnitude of the position vector

In subroutines: MAIN, AEIXYZ, ASOP, INPUT, OUTPUT, POTJ2, PSTOX, TIMEPS, XTOPS, XYZAEI

/CBASIC/

Location	Name	Dimension	Type/ precision	Description
1	PI	1	R/D	π
2	TWOPI	1	R/D	2π
3	DEG	1	R/D	$\pi/180$
4	RAD	1	R/D	$180/\pi$
5	DAY	1	R/D	Converts days into hours, minutes, or seconds

In subroutines: MAIN, ASOP, CONST, INPUT, OUTPUT, PSANS, TIMEPS, XYZAEI

/CPRINT/

Location	Name	Dimension	Type/ precision	Description
1	PRINT	1	R/D	Input parameters, see section 4.1.1
2	IPRINT	1	I	
3	IELPRT	1	I	
4	IUNITS	1	I	

In subroutines: MAIN, CONST, INPUT, OUTPUT

/DRAG/

Location	Name	Dimension	Type/ precision	Description
1	CD	1	R/D	Input parameters, see section 4.1.1
2	AREA	1	R/D	
3	XMASS	1	R/D	

In subroutines: INPUT

/END/

Location	Name	Dimension	Type/ precision	Description
1	STOP	1	R/D	Input parameters, see section 4.1.1
2	ISTOP	1	I	

In subroutines: MAIN, INPUT, OUTPUT

/EPOCH/

Location	Name	Dimension	Type/ precision	Description
1-5	DATE	5	R/D	Input parameter, see section 4.1.1
6	XJDATE	1	R/D	Julian date

In subroutines: INPUT

/GEO/

Location	Name	Dimension	Type/ precision	Description
1	RE	1	R/D	Equatorial radius of the central body
2	CJ2	1	R/D	J_2 geopotential coefficient of the central body

In subroutines: CONST

/IPOOL/

Location	Name	Dimension	Type/ precision	Description
1-10	--	--	I	Temporary integer variables; this COMMON block is used to help save data storage within the ASOP program.

In subroutines: PSANS

/KEPLER/

Location	Name	Dimension	Type/ precision	Description
1	A	1	R/D	Semimajor axis of the orbit (a)
2	E	1	R/D	Eccentricity (e)
3	XI	1	R/D	Orbital inclination to the Equator (i)
4	OMEGA	1	R/D	Argument of pericenter (ω)
5	XNODE	1	R/D	Argument of the ascending (Ω) node
6	XM	1	R/D	Mean anomaly (M)

In subroutines: AEIXYZ, INPUT, OUTPUT, XYZAEI

/KONST/

Location	Name	Dimension	Type/ precision	Description
1	XMU	1	R/D	Gravitational constant of the central body (μ)
2	XMUI	1	R/D	$1/\mu$
3	SQTMU	1	R/D	$\sqrt{\mu}$
4	SQTMUI	1	R/D	$1/\sqrt{\mu}$
5	EPS	1	R/D	$\varepsilon = 3/2\ \mu\ J_2\ R_e^2$ where J_2 = J_2 geopotential coefficient R_e = Equatorial radius of the central body

In subroutines: AE1XYZ, CONST, POTJ2, PSANS, PSTOX, TIMEPS, XTOPS, XYZAEI

/PERTRB/

Location	Name	Dimension	Type/ precision	Description
1	IDRAG	1	I	Input parameters, see section 4.1.1
2	ILONG	1	I	

In subroutines: INPUT

/PS/

Location	Name	Dimension	Type/ precision	Description
1-4	SIG	4	R/D	PS elements σ_1, σ_2, σ_3, σ_4
5-8	RHO	4	R/D	PS elements ρ_1, ρ_2, ρ_3, ρ_4
9	TAU	1	R/D	Independent variable of the PS elements.

In subroutines: MAIN, ASOP, OUTPUT, PSANS, PSTOX, TIMEPS, XTOPS

/PSTIME/

Location	Name	Dimension	Type/ precision	Description
1	CLC	1	R/D	Initial value of σ_{μ} (set when initializing, NEWX = 0)
2	FAKTPS	1	R/D	$\partial\sigma_{\mu}/\partial\tau$
3	Q	1	R/D	q

In subroutines: ASOP, PSANS, PSTOX, TIMEPS

/RPOOL/

Location	Name	Dimension	Type/ precision	Description
1-24	--	24	R/D	Temporary real variables; this COMMON block is used to help save storage within the ASOP program

In subroutines: AEIXYZ, POTJ2, PSANS, PSTOX, XTOPS, XYZAEI

/TESS/

Location	Name	Dimension	Type/ precision	Description
1	NMAX	1	I	Input parameters; see section 4.1.1
2	MMAX	1	I	

In subroutines: CONST, INPUT

4.0 USER'S GUIDE

This section is intended to give the user all the information necessary to operate the ASOP programs. Because the program is designed to operate in two modes (stand-alone and subroutine package), each mode of operation is described separately.

The first part of this section (section 4.1) will describe the general input parameters and options available when using the stand-alone ASOP program. Also described in this section are the standard default values and the typical control cards needed for executing the program in the demand or the batch mode. Finally some sample output is given to help the user if modifications are to be made to the system.

Section 4.2 will deal with the ASOP subroutine package. This section will describe the necessary modules that are used within the package, as well as any interface requirement that the user must be aware of if he is to include this package in his own software. The input to and output from the ASOP subroutine are also fully described in this section, as are the subroutines default values.

4.1 INDEPENDENT PROGRAM

The ASOP program was designed as an interactive program capable of giving the user fast, accurate answers to Shuttle-type orbit problems. The program may also be run in a batch environment if a large number of cases must be investigated.

There are two basic methods used to control the operation of the ASOP program; flags and direct-user interaction. The flags are used to indicate the type of data being entered and to select certain options within the program. Direct interaction allows the user to check his data to insure accuracy before continuing or to terminate the program.

Primary data input to the ASOP program is accomplished using the NAMELIST '$INPUT'. The necessary input variables and user responses to program questions are described in section 4.1.1, and the program default values are described in section 4.1.2. Section 4.1.3 explains the printed output generated by the ASOP program. Finally, sections 4.1.4 and 4.1.5 describe the instructions necessary to run the ASOP program and give an example of the resulting output.

4.1.1 Input Description

The NAMELIST is the primary method of getting data into the ASOP program. However, during normal operation, the user is expected to interact with the program by supplying additional information. After starting the ASOP program (see section 4.1.4), it will ask for the NAMELIST data with the statement

```
INPUT DATA USING NAMELIST '$INPUT'
```

At this point, the user has three options:

a. To enter the NAMELIST data directly from the keyboard.

b. To add a data file containing the NAMELIST information using the @ADD command (ref. 5).

c. To enter @EOF to terminate program execution.

If option C is selected, the program will respond with

NORMAL PROGRAM TERMINATION

and the program will stop. If option A or B is selected, then the program will print out all the NAMELIST variables and their associated values (including default), as well as the initial conditions of the problem. The message

ENTER: C = CONTINUE; S = STOP

should then appear. Here, the user should check the input data and enter the necessary letter. If a C is entered, the program will continue the execution as directed by the input. When the input stop condition is satisfied, the program will again ask for data input as described earlier. The series of instructions can be repeated as often as necessary.

If an S is entered in response to the message, then the program will respond with **NORMAL PROGRAM TERMINATION**. The user may then make the necessary corrections to the data and restart the program as described in section 4.1.4.

Table I describes the input variables that may be used in the NAMELIST '$INPUT'. Whether keying in the information or creating a data element, a b̸$INPUT must be entered first where the b̸ represents one or more spaces. Each variable entered must be preceded by one or more spaces, and if more than one variable is to appear on a line, they must be separated by a blank or a comma (,). To terminate the NAMELIST input a b̸$END or b̸$ must be the last item entered. (See ref. 7, pages 6 through 13, for a complete description of a NAMELIST statement).

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE I.- NAMELIST INPUT VARIABLES

Variable	Type	No. of inputs required	Description and available options
EL	DP	6	Must be supplied by the user. Can be given as Cartesian coordinates or Keplerian elements as determined by the flag IEL EL (1) = X or a EL (2) = Y or e EL (3) = Z or i EL (4) = $\dot{X}$ or ω EL (5) = $\dot{Y}$ or Ω EL (6) = $\dot{Z}$ or M All angular input is given in degrees. All other values must correspond to the option specified by the flag IUNITS.
IEL	I	1	Flag determining the type of initial conditions input. 1 = Keplerian elements[a] 2 = Cartesian coordinates
STOP	DP	1	Final condition that must be satisfied in order to stop program execution normally
ISTOP	I	1	Flag that specifies the type of STOP condition 1 = STOP in days 2 = STOP in revolutions[a]
PRINT	DP	1	Increment for the printed output. A value is not needed if IPRINT is set to 0. PRINT = 0.0 is a valid entry.

[a]Default value.

TABLE I.- Continued

Variable	Type	No. of inputs required	Description and available options
IPRINT	I	1	Flag that specifies the PRINT increment. 0 = No PRINT increment[a] 1 = PRINT is days 2 = PRINT is revolutions
DATE[b]	DP	6	Date of epoch given as a calendar date of the form Day, month, year, hours, minutes, seconds (to be implemented)
IDRAG[b]	I	1	Flag that specifies if the drag equations are to be included in the computation 0 = No[a] 1 = Yes (to be implemented)
AREA[b]	DP	1	Frontal surface area of the satellite. A value is not needed if IDRAG is set to 0 (to be implemented)
XMASS[b]	DP	1	Total mass of the satellite. A value is not needed if IDRAG is set to 0 (to be implemented)
ILONG	I	1	Flag that specifies the type of potential terms to be included in the computations 0 = None (two-body orbit) 1 = J_2, short period, and first order secular terms[a] 2 = J_2, long period, and higher order secular terms[b]

[a]Default value.
[b]To be implemented. Current values set to zero.

TABLE I.- Concluded

Variable	Type	No. of inputs required	Description and available options
NMAX	I	1	Total number of zonal terms to be included by the geopotential model; a value is needed only if ILONG is set to 2 (to be implemented).
MMAX	I	1	Total number of tesseral terms to be included by the geopotential model; a value is needed only if ILONG is set to 2 (to be implemented).
IELPRT	I	1	Flag to determine if the PS elements are to be included with all printout. 0 = No[a] 1 = Yes
IUNITS	I	1	Flag that specifi[illegible]he units of the input data and se[illegible] the appropriate physical const[illegible]. 1 = km, sec[a] 2 = nm, sec 3 = ft, sec 4 = m, sec 5 = km, hr 6 = nm, hr 7 = E.r., min

[a]Default value.

4.1.2 Default Values

To help shorten the number of data values that must be supplied by the user, the ASOP program assumes certain default values for those variables not explicitly mentioned on the input NAMELIST. These default values are listed in table II and a description of the variables can be found in section 4.1.1. Any variable not listed in table II must be specified by the user.

TABLE II.- DEFAULT NAMELIST VALUES

Variable	Default value
IEL	1
.TOP	100.0
ISTOP	2
PRINT	0.0
IPRINT	0
IDRAG[a]	0
AREA[a]	0.0
CD[a]	0.0
XMASS[a]	0.0
ILONG	1
NMAX[a]	2
MMAX[a]	0
IELPRT	0
IUNITS	1

[a]To be implemented.

4.1.3 Output Description

After the ASOP program has been started with the command

```
@XQT *NUMEG.ASOP-PROG
```

and the input data has been added, the program will print out all of the NAMELIST variables as shown in figure 15 and the initial conditions shown in figure 16.

```
$INPUT
 EL      =   .32726586100000000000D+004,   .5737872519999999
98D+004,
             .00000000000000000000D+000,  -.6774775910000000
00D+001,
             .39102830199999999990D+001,   .0000000000000000
00D+000
 IEL     =        +2
 STOP    =   .10000000000000000000D+003
 ISTOP   =        +2
 PRINT   =   .50000000000000000000D+002
 IPRINT  =        +2
 DATE    =   .00000000000000000000D+000,   .0000000000000000
00D+000,
             .00000000000000000000D+000,   .0000000000000000
00D+000,
             .00000000000000000000D+000
 IDRAG   =        +0
 AREA    =   .00000000000000000000D+000
 CD      =   .00000000000000000000D+000
 XMASS   =   .00000000000000000000D+000
 ILONG   =        +1
 NMAX    =        +2
 MMAX    =        +0
 IELPRT  =        +1
 IUNITS  =        +1
$END
```

Figure 15.- NAMELIST data output format.

```
----------------------------------------------------------------------------
                          INITIAL CONDITIONS
 DAYS: 0.0000000            ! REVS: 0.0000000      **!CHECK:-1.0000000+00
----------------------------+------------------------+----------------------
     A= 6.6993359+03 KM     !    E= 1.490[illegible]-02     ! I=      .00000   DEG
OMEGA= 339.57024     DEG    !NODE=      .00000  DEG  ! M=  19.83963     DEG
     X= 3.2726586+03 KM     !    Y= 5.73787.5+03 KM  ! Z= 0.0000000     KM
   VX=-6.7747759+00 KM/S    !   VY= 3.9102830+00 KM/S!VZ= 0.0000000     KM/S
PS ELEMENTS:
          .10524559+001 -.19730034+001   .00000000         .89433066+001
          .51679859+005  .24862148+001   .00000000         .29777769+002
----------------------------------------------------------------------------
ENTER: 1 = CONTINUE, 9 = STOP
>
```

Figure 16.- Initial condition format.

After the initial conditions have been displayed, the program will wait for the user to check the input parameters. Some checks that can be made are

a. CHECK value should be -1.0000000+00.

b. DAYS and REVS should be 0.0000000.

c. A double asterisk (**) will appear after the output condition that is being satisfied, i.e., after the DAYS or the REVS value. This will agree with input value of IPRINT or ISTOP.

d. Are the initial conditions (a, e, i, etc.) the desired values?

e. Are the units correct (km, km/s, etc.)?

Once the user is satisfied that the printed initial conditions are the ones needed, a C should be entered. When the C is entered, the program will continue execution and print out information as specified by the input parameters, i.e., STOP, ISTOP, PRINT AND IPRINT (figure 17).

```
 ENTER:  C = CONTINUE; S = STOP
>C
 DAYS: 3.1493985+00        ! REVS: 5.0000000+01  **!CHECK: 4.9900358-07
---------------------------+---------------------------+---------------------------
     A= 6.6993637+03 KM    !    E= 1.5013305-02        ! I=      .00000      DEG
 OMEGA= 359.53545     DEG  !NODE=      .00000    DEG   ! M=      .4507.      DEG
     X= 2.8173764+03 KM    !    Y= 5.9670977+03 KM     ! Z= 0.0000000        KM
    VX=-7.0802476+00 KM/S!  VY= 3.3439907+00 KM/S!VZ= 0.0000000              KM/S
 PS ELEMENTS:
          .31528894+003 -.28657742+001  .00000000       .27210824+006
          .51674873+005  .13836588+001  .00000000       .2977976 9+00?
-------------------------------------------------------------------------------------
                            FINAL CONDITIONS
 DAYS: 6.2987974+00        ! REVS: 1.0000000+02  **!CHECK: 4.9689020-07
---------------------------+---------------------------+---------------------------
     A= 6.6993344+03 KM    !    E= 1.491..56-02        ! I=      .00000      DEG
 OMEGA=  88.64542     DEG  !NODE=      .00000    DEG   ! M= 341.06898        DEG
     X= 2.3508691+03 KM    !    Y= 6.1724382+03 KM     ! Z= 0.0000000        KM
    VX=-7.3242597+00 KM/S!  VY= 2.7484611+00 KM/S!VZ= 0.0000000              KM/S
 PS ELEMENTS:
          .62952542+003 -.31747392+001  .00000000       .54420755+006
          .51674849+005 -.95472769-003  .00000000       .29779769+002
-------------------------------------------------------------------------------------
 INPUT DATA USING NAMELIST '$INPUT'
>_
```

Figure 17.- Intermediate and final output format.

In general, all output is clearly labeled, but some terms should be explained further.

DAYS: Total number of days elapsed since the starting epoch.

REVS: Total number of revolutions completed.

CHECK: Value indicating the accuracy of the analytical theory; although this value is necessary as a check on the theory, it is not a sufficient check.

ORIGINAL PAGE IS
OF POOR QUALITY

PS ELEMENTS: The Poincaré-Similar elements listed as

$\sigma_1 \quad \sigma_2 \quad \sigma_3 \quad \sigma_4$

$\rho_1 \quad \rho_2 \quad \rho_3 \quad \rho_4$

Double asterisk (**): Indicates the stopping condition being satisfied; this flag will move between the DAYS and REVS value as needed.

4.1.4 Run Setup (Control Cards)

The ASOP program is written in standard FORTRAN V and designed to run on the NASA/JSC UNIVAC 1110 computer using the EXEC-8 operating system. All the relocatable and executable elements are on the file FM6-N08569*NUMEG. ASOP may be executed by entering the following for demand operation.

a. @QUAL FM6-N08569

b. @ASG,A *NUMEG.

c. @XQT *NUMEG.ASOP-PROG

d. Add input data

e. Enter the letter C or S (see section 4.1.3)

f. @EOF or go to step d.

g. @FIN

If run in a batch mode, the following input cards are needed.

a. @QUAL FM6-N08569

b. @ASG,A *NUMEG.

c. @XQT *NUMEG.ASOP-PROG

d. Add data file or data cards

e. 'C'

f. Repeat instructions d and e as often as necessary.

g. @EOF

h. @FIN

4.1.5 Sample Computer Run

In this section, a sample computer run is reproduced for a typical Shuttle-type orbit. The orbit has been predicted for 100 revolutions (≈6.3 days) with the output given every three days.

This example is intended to familiarize the user with the format of the ASOP output and to illustrate the use of the various input options discussed in section 4.1.1. A full description of the output is given in section 4.1.3.

Initial parameters

Semimajor axis (a)	6699.3532 km (1.05 ER)
Eccentricity (e)	.001
Inclination (i)	30 degrees
Argument of perigee (ω)	18 degrees
Argument of the ascending node (Ω)	20 degrees
Mean anomaly (M)	22 degrees

Input parameters under the NAMELIST $INPUT

```
$INPUT
EL(1)    = 3542.07055
EL(2)    = 5256.17858
EL(3)    = 2152.20751
EL(4)    = -6.41478310
EL(5)    = 3.11545050
EL(6)    = 2.95692882
IEL      = 2
STOP     = 100.0
ISTOP    = 2
PRINT    = 3.0
IPRINT   = 1
ILONG    = 1
NMAX     = 2
MMAX     = 0
IELPRT   = 1
IUNITS   = 1
$END
```

Sample computer run.

```
@XQT *NUMF6.ASOP-PROG
 INPUT DATA USING NAMELIST '$INPUT'
>@ADD S.DATA-ASOP3
> STOP    = 100.000
> ISTOP   = 2
> PRINT   = 3.000
> IPRINT  = 1
> $END
 $INPUT
 EL       =              .35420705499999999D+004,          .5256178579999999
99D+004,
                         .21522075100000000D+004,         -.6414783099999999
99D+001,
                         .31154505000000000D+001,          .2956928820000000
00D+001
 IEL      =                  +2
 STOP     =              .10000000000000000000D+003
 ISTOP    =                  +2
 PRINT    =              .30000000000000000000D+001
 IPRINT   =                  +1
 DATE     =              .00000000000000000000D+000,       .000000000000000000
00D+000,
                         .00000000000000000000D+000,       .000000000000000000
00D+000,
                         .00000000000000000000D+000
 IDRAG    =                  +0
 AREA     =              .00000000000000000000D+000
 CD       =              .00000000000000000000D+000
 XMASS    =              .00000000000000000000D+000
 ILONG    =                  +1
 NMAX     =                  +2
 MMAX     =                  +0
 IELPRT   =                  +1
 IUNITS   =                  +1
 $END
-----------------------------------------------------------------------------
                            INITIAL CONDITIONS
 DAYS: 0.0000000       ** ! REVS: 0.0000000         !CHECK:-1.0000000+00
--------------------------+---------------------------+----------------------
     A= 6.6993532+03 KM   !    E= 9.0995685-04        ! I=  30.00000     DEG
 OMEGA=  17.92236     DEG !NODE=  20.00000      DEG   ! M=  22.05838     DEG
     X= 3.5425105+03 KM   !    Y= 5.2561786+03 KM     ! Z= 2.1522075+03 KM
    VX=-6.4147831+00 KM/S!   VY= 3.1154505+00 KM/S!VZ= 2.9569280+00 KM/S
PS ELEMENTS:
          .10575457+001   .63790429-002 -.40245815+002   .55372075+000
          .51675619+002   .86109799-001  .11057497+003   .29769928+002
-----------------------------------------------------------------------------
ENTER:  C = CONTINUE, S = STOP
```

Sample computer run.- Concluded

```
ENTER:  C = CONTINUE; S = STOP
>C
 DAYS: 3.0000000+00  ** ! REVS: 4.7582597+01    !CHECK: 1.0093107-06
-----------------------+-----------------------------+-------------------------
   A= 6.6964E01+03 KM  !   E= 9.6051810-04          ! I=  29.97058    DEG
OMEGA= 296.00154   DEG !NODE= 358.10580    DEG ! M= 335.38775     DEG
   X=-3.4730360+01 KM  !   Y=-5.7959173+03 KM  ! Z=-3.3422[illegible]+03 KM
  VX= 7.7207886+00 KM/S!  VY=-9.5760931-02 KM/S!VZ= 9.2009[illegible]-02 KM/S
PS ELEMENTS:
        .30001249+003  .14328777+000  .38867651+001  .2591[illegible]31+006
        .51664463+005  .89695373-001  .117[illegible]1105+003  .29[illegible]9428+002
-------------------------------------------------------------------------------
 DAYS: 6.0000000+00  ** ! REVS: 9.5165702+01    !CHECK: 5.3573[illegible]
-----------------------+-----------------------------+-------------------------
   A= 6.6995870+03 KM  !   E= 7.7194707-04          ! I=  30.00177    DEG
OMEGA= 108.73237   DEG !NODE= 336.21553    DEG ! M=  34.06419     DEG
   X=-3.4707616+03 KM  !   Y= 5.3565510+03 KM  ! Z= 2.0219[illegible]0+03 KM
  VX=-6.4158754+00 KM/S!  VY=-2.9922585+00 KM/S!VZ=-3.0749717+00 KM/S
PS ELEMENTS:
        .59898062+003 -.33073053-001  .47459491+002  .51840075+006
        .51676523+005  .94153688-001  .10768413+003  .29769428+002
-------------------------------------------------------------------------------
                          FINAL CONDITIONS
 DAYS: 6.3047894+00      ! REVS: 1.0000000+02  **!CHECK: 1.0044341-06
-----------------------+-----------------------------+-------------------------
   A= 6.6964889+03 KM  !   E= 8.4715012-04          ! I=  29.97880    DEG
OMEGA=  53.13712   DEG !NODE= 334.03604    DEG ! M=  32.16245     DEG
   X= 3.0169511+03 KM  !   Y= 4.9567[illegible]5+03 KM  ! Z= 3.3327003+03 KM
  VX=-6.6797490+00 KM/S!  VY= 3.8589430+00 KM/S!VZ= 3.1440[illegible]-01 KM/S
PS ELEMENTS:
        .62935504+004 -.30904845-001  .51475927+002  .54470459+006
        .51664576+005  .1422[illegible]+000  .10570999+003  .2976[illegible]8+002
-------------------------------------------------------------------------------
INPUT DATA USING NAMELIST '$INPUT'
>EOF
 ** NORMAL PROGRAM TERMINATION **
```

ORIGINAL PAGE IS
OF POOR QUALITY

In the interactive mode, the program solicits input data with the messages:

a. INPUT DATA USING NAMELIST '$INPUT'

b. ENTER: C = CONTINUE; S = STOP

In batch mode, the data input must be followed by one card containing the letter C[a] as shown in the following examples:

Example (1)	Example (2)
ƀ$INPUT	@ADD filename.data element
.	.
.	.
.	.
necessary data cards (see section 4.1.1)	changes to data element (if any)
.	.
.	.
.	.
ƀ$END	ƀ$END
C	C
ƀ$INPUT	@ADD filename.data element
.	.
.	.
.	.
etc.	etc.
.	.
.	.
.	.
@EOF	@EOF
@FIN	@FIN

(ƀ is a blank that must be included)

4.2 SUBROUTINE PACKAGE

Along with the ASOP stand-alone program, there is a subroutine package that may be included in the user's software. This package is in the form of a relocatable element and is located in FM6-N08569*NUMEG.ASOP-SUB. Its operation is identical with any user-written subroutine.

This section will describe the information needed by the user to insure proper insertion and operation of the ASOP package within the user's software.

---

[a]The card containing the letter C may be replaced by just a blank card.

4.2.1 Required Subroutines

The ASOP subroutine package consists of seven subroutines. These are a driver subroutine (ASOP) that controls the basic logic of the package and six general subroutines that perform the functions necessary to the analytical theory. These subroutines are

ASOP	Driver subroutine
CONST	Planetary and mathematical constants
POTJ2	Determines J_2 potential
PSANS	PS analytical theory
PSTOX	Transformation subroutine:PS elements to Cartesian coordinates
TIMEPS	Time iteration stopping procedure
XTOPS	Transformation subroutine:Cartesian coordinates to PS elements

The subroutines listed above are fully described in section 3.3, and a diagram of the data flow between these subroutines can be found in appendix G.

To help the user add these subroutines to his own software, a relocatable element has been formed that includes all the above subroutines. Therefore, the user needs only to include the element

```
FM6-N08569*NUMEG.ASOP-SUB
```

when forming an executable element.

4.2.2 Interface Requirement

To access the ASOP subroutine package, the programer must use the FORTRAN statement

```
CALL ASOP (X,STOP,ISTOP,NEWX)
```

A full description of the argument list variables can be found in sections 4.2.3 and 3.3.3. Also, the user must initialize certain COMMON block variables before entering the ASOP subroutine.

Table III gives a list of the variables that must be initialized prior to calling the ASOP subroutine and the COMMON block in which the variables are located. A complete description of the variables and their allowed values can be found in section 4.1.1.

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE III.- COMMON BLOCK INITIALIZATION

Variable	COMMON block	Default value(s)	Page
IUNITS	CPRINT	1	61
CD[a]	DRAG	0.0	61
AREA[a]	DRAG	0.0	61
XMASS[a]	DRAG	0.0	61
ILONG	PERTRB	1	63
DATE[a]	EPOCH	5*0.0	61
XJDATE[a]	EPOCH	0.0	61
NMAX[a]	TESS	2	64
MMAX[a]	TESS	0	64

[a]To be implemented.

4.2.3 Input/Output Description

The argument list to the ASOP subroutine consists of four arguments given in the following order.

CALL ASOP (X,STOP,ISTOP,NEWX)

On input the variables are

X An array of eight elements corresponding to the initial state vector in the following order:

X(1): X position
X(2): Y position
X(3): Z position
X(4): X velocity component
X(5): Y velocity component
X(6): Z velocity component
X(7): Physical time (set to zero)
X(8): Total energy (set to zero)

STOP Stop value desired; it may be given in days or revolutions.

ISTOP Flag determining whether the value given to STOP is in days (ISTOP = 1) or revolutions (ISTOP = 2).

NEWX Flag determining if the ASOP subroutine is to be initialized

NO = 0, YES = 1

The initialization process must be done whenever new initial conditions are entered.

> WARNING: A number must not be used in this position of the argument list. Assign the desired value to a variable, and use the variable in the argument list. If a number is used in this position instead of a variable, unpredictable results may occur.

Input to the ASOP program is also done by means of COMMON blocks. These COMMON block input variables control the internal operation of the ASOP subroutine package and should not be changed once the subroutine has been initialized.

Table III (section 4.2.2) gives a complete list of the variables that must be initialized and gives their default values.

On output the variables are

X An array of eight elements corresponding to the final state vector at the given value of STOP. The order is the same as for input. If the value of STOP was given in days (ISTOP = 1), then the value of X(7) will be an approximate value of STOP given in the time units specified by IUNITS.

STOP, ISTOP } Unchanged from the input values.

NEWX Set to zero (0) upon return from the ASOP subroutine; therefore, a variable name should always occupy this position in the argument list. Otherwise, unpredictable results may occur.

REFERENCES

1. Mueller, A. C.: The Development of the Poincaré-Similar Elements With the True Anomaly as the Independent Variable. JSC IN 76-FM-60, August 1976.

2. Bond, V. R.: Analytical, Singularity-Free Solution to the J_2 Perturbation Problem. JSC IN 77-FM-52, October 1977.

3. Schiefele, G.; Mueller, A.; and Starke, S.: A Singularity Free Analytical Solution of Artificial Satellite Motion With Drag. ACM Technical Report, ACM-TR-103, March 1977.

4. Mueller, A.: Recursive Analytical Solution Describing Artificial Satellite Motion Perturbed by an Arbitrary Number of Zonal Terms. Presented at the 1977 AAS/AIAA Astrouynamics Conference, September, 1977.

5. UNIVAC 1100 Series Operating System Programmer Reference. UP-4144, Rev. 3, 1973.

6. Mueller, A. C.: An Analytical State Transition Matrix for Orbits Perturbed by an Oblate Spheroid. ACM Technical Report, ACM-TR-104, May 1977.

7. Sperry UNIVAC 1100 Series FORTRAN V Programmer Reference. UP-4060, Rev. 2, 1974.

APPENDIX A

AVAILABLE UNITS AND PHYSICAL CONSTANTS

Within the ASOP program, there are a number of options for the input units. The compatibility of the input values with the selected physical constants is the responsibility of the user and is controlled by the input flag IUNITS. The following constants are listed in order according to the value assigned to IUNITS. If no value for IUNITS is explicitly given, then 1 is assumed.

R_e = Earth radius (equatorial)

= 6378.140 km	(IUNITS = 1,5)
= 3443.920 nm	(IUNITS = 2,6)
= 2.092566×10^7 ft	(IUNITS = 3)
= 6.378140×10^6 m	(IUNITS = 4)
= 1.0 E.r.	(IUNITS = 7)

μ = Gravitational constant of the Earth

= 3.986013×10^5 km^3/sec^2	(IUNITS = 1)
= 6.275029×10^4 nm^3/sec^2	(IUNITS = 2)
= 1.407647×10^{16} ft^3/sec^2	(IUNITS = 3)
= 3.986013×10^{14} m^3/sec^2	(IUNITS = 4)
= 5.165873×10^{12} km^3/hr^2	(IUNITS = 5)
= 8.132438×10^{11} nm^3/hr^2	(IUNITS = 6)
= 5.530432×10^{-3} E.r.^{3}/min^2	(IUNITS = 7)

DAY = Time conversion

= 8.64×10^4 sec/day	(IUNITS = 1,2,3,4)
= 1.44×10^3 min/day	(IUNITS = 7)
= 2.40×10^1 hr/day	(IUNITS = 5,6)

ORIGINAL PAGE IS
OF POOR QUALITY

$J_2 = J_2$ coefficient of the Earth's geopotential

$\varepsilon = 1.082637 \times 10^{-3}$

ε = EPS = $3/2\ (\mu\ J_2\ R_e^{\ 2})$

π = PI = 3.14159 26535 898

If Keplerian elements are input (IEL = 1), then the semimajor axis must be input in a distance compatible with the selected value of IUNITS, i.e., if IUNITS = 2, then EL(1) must be given in nautical miles; if IUNITS = 4, then EL(1) must be in meters, etc. All angles must be given in degrees.

For Cartesian coordinates, the input unit must be

IUNITS	EL(1) → EL(3)	EL(4) → EL(6)
1	km	km/sec
2	nm	nm/sec
3	ft	ft/sec
4	m	m/sec
5	km	km/hr
6	nm	nm/hr
7	E.r	E.r/min

All computations within ASOP are done using the input units.

APPENDIX B

REQUIRED CONTROL CARDS

1. >@QUAL FM6-N08569

2. >@ASG,A *NUMEG.

3. >@XQT *NUMEG.ASOP-PROG

4. INPUT DATA USING NAMELIST '$INPUT'

5.

>b̸$INPUT Necessary input parameters; see section 4.1.1. All parameters must be preceded by at least one space.	or	>@ADD filename.element name > Input the necessary correction if needed.
>b̸$END		>b̸$END

6. Initial output

7. ENTER: C = CONTINUE; S = STOP

   \>

   If an S is entered in response to the prompt '>', then the message **NORMAL PROGRAM TERMINATION** should appear.

   If a C is entered, the program will printout the desired information, and the sequence will begin again at line 4.

8. >@EOF

   NORMAL PROGRAM TERMINATION

   \> = system prompt;
   b̸ = necessary space

APPENDIX C

ASOP DEFAULT VALUES

IEL = 1 (Keplerian elements)

EL = None; must be input by user

STOP = 100.0D0

ISTOP = 2 (STOP value is in revolutions)

PRINT = 0.0D0

IPRINT = 0 (No printout; PRINT value ignored)

IDRAG[a] = 0 (No drag terms desired)

AREA[a] = 0.0D0

CD[a] = 0.0D0

XMASS[a] = 0.0D0

ILONG = 1 (J_2 and sh period secular terms)

NMAX[a] = 2 (Include $J_{2,0}$ zonal term)

MMAX[a] = 0 (No tesseral terms)

IELPRT = 0 (Do not print PS elements)

IUNITS = 1 (Input and output values are given as

distance = km
velocity = km/sec
time = day
angles = deg)

---

[a]To be implemented.

APPENDIX D

SUBROUTINE CYCLE TIMES AND STORAGE REQUIREMENTS

All times refer to the execution of a FORTRAN V program on a UNIVAC 1110-EXEC 8 system.

FORTRAN compiler: SOE3

MAP processor : 27.1 RL71-3

Subroutine name	Cycle times, ms	Storage requirements, words	
		Octal	Decimal
MAIN	[a] 34 to 42	145	101
AEIXYZ	<1	356	238
ASOP[b]	2 to 11	134	92
CONST[b]	<1	207	135
INPUT	--	272	186
OUTPUT	28 to 33	503	326
POTJ2[b]	<1	41	33
PSANS[b]	1	1773	1019
PSTOX[b]	<1	360	240
TIMEPS[b]	4 to 9	202	130
XTOPS[b]	<1	540	352
XYZAEI	<1	406	262
Totals			
ASOP program		6047	3111
ASOP subroutine package		3721	2001

[a] Does not include the time required for data input.
[b] ASOP subroutine package programs.

The cycle times shown are the total times needed by the module to return an answer after it is called. Therefore, if a module must call a subroutine, then the time shown for that module is the execution time needed by the module and the subroutine called.

Example: The time of 2 to 11 milliseconds given for the ASOP subroutine is the time required by the module to

a. Accept the input data

b. Transform the input data into the PS elements (call XTOPS)

c. Initialize the analytical theory and set the necessary constants (call PSANS and CONST)

d. Compute the new PS elements (call PSANS)

e. Transform the new PS elements back into the Cartesian coordinates (call PSTOX)

f. Return to the calling routine

Values given for the storage requirements of the individual subroutines are values returned by the FORTRAN compiler when forming a relocatable element. The final, executable program will require more space because of the system library modules that must also be included.

APPENDIX E

GENERAL VARIABLE ABBREVIATIONS AND DEFINITIONS

PS elements:

Coordinates	$\sigma_1, \sigma_2, \sigma_3, \sigma_4$
	(Note: $\sigma_5 = \rho_1, \sigma_6 = \rho_2, \sigma_7 = \rho_3, \sigma_8 = \rho_4$)
Momenta	$\rho_1, \rho_2, \rho_3, \rho_4$
Independent variable	τ (true anomaly)

PS Hamiltonian:

$$F = \rho_1 - \frac{\mu}{\sqrt{2\rho_4}} + \frac{r^2}{q} V$$

where $q = G - 1/2\ \Phi + \frac{\mu}{2\ \ 2L}$, and V is the perturbing potential.

DS elements:

ϕ = true anomaly

g = argument of perigee

h = argument of the ascending mode

ℓ = time element

Φ = conjugate to ϕ, related to the two-body energy

G = total angular momentum

H = Z component of the angular momentum

L = total energy (two body plus perturbing potential)

(Note: For a complete description of the relationship between the DS and the PS elements, see reference 2.)

Cartesian coordinates:

$\vec{X} = (X_x, X_y, X_z)$ = position vector

$\vec{V} = (V_x, V_y, V_z)$ = velocity vector

r = magnitude of the position vector

t = physical time

Keplerian elements:

a = semimajor axis

e = eccentricity

i = inclination to the equator

ω = argument of pericenter

Ω = argument of the ascending node

M = mean anomaly

Planetary variables: (see appendix A for the numerical values used)

R_e = equatorial radius

μ = gravitational constant

General:

km = kilometers

nm = nautical miles

ft = feet

m = meters

E.r. = Earth radius

sec = seconds

hr = hours

min = minutes

rad = radians

deg = degrees

t = time

$\rightarrow$ = denotes a vector as $\vec{X}$

APPENDIX F

EQUATIONS OF THE ANALYTICAL THEORY

A complete first order solution for the motion of a satellite perturbed by oblateness has been developed (ref. 2). A brief outline was given in reference 6 and is reproduced in this appendix.

The Hamiltonian for the J_2 perturbed case can be written as

$$F = \rho_1 - \frac{\mu}{\sqrt{2\rho_4}} + \varepsilon F_1$$

where

$$F_1 = 1/r \left[\left(\frac{X_3}{r} \right)^2 - \frac{1}{3} \right]$$

and

$$\varepsilon = 3/2 \; J_2 \; \mu \; R_e^2$$

R_e is the mean equatorial radius of the central body; μ is the gravitational constant of the central body, and J_2 is the J_2 oblateness coefficient.

The differential equations are solved by a method of Von-Zeipel. The elements undergo a canonical transformation through a determining function S_1 so that the short periodic terms are eliminated from the Hamiltonian. The equations of motion in the transformed system $\vec{\sigma}'$ may then be solved with an accuracy of order ε.

The solution algorithm can be divided into three steps:

a. Canonical transformation to eliminate the short periodic terms:

$$\sigma'_{k,0} = \sigma_{k,0} + \varepsilon \frac{\partial S_1}{\partial \rho_{k,0}} (\sigma_0, \rho_0)$$

$$\rho'_{k,0} = \rho_{k,0} - \varepsilon \frac{\partial S_1}{\partial \sigma_{k,0}} (\sigma_0, \rho_0)$$

$$k = 1, 2, 3, 4$$

ORIGINAL PAGE IS
OF POOR QUALITY

b. The analytical integration of the transformed equations of motion:

$$\sigma_1' = \sigma_{1,0}' + A_1\tau$$
$$\sigma_2' = \sigma_{2,0}' \cos(A_2\tau) - \rho_{2,0}' \sin(A_2\tau)$$
$$\sigma_3' = \sigma_{3,0}' \cos(A_3\tau) - \rho_{3,0}' \sin(A_3\tau)$$
$$\sigma_4' = \sigma_{4,0}' + A_4\tau$$
$$\rho_1' = \rho_{1,0}'$$
$$\rho_2' = \rho_{2,0}' \cos(A_2\tau) + \sigma_{2,0}' \sin(A_2\tau)$$
$$\rho_3' = \rho_{3,0}' \cos(A_3\tau) + \sigma_{3,0}' \sin(A_3\tau)$$
$$\rho_4' = \rho_{4,0}'$$

c. The back transformation:

$$\sigma_k = \sigma_k' - \varepsilon \frac{\partial S_1}{\partial \rho_{k'}}(\sigma',\rho')$$

$$\rho_k = \rho_k' + \varepsilon \frac{\partial S_1}{\partial \sigma_{k'}}(\sigma',\rho')$$

$$k = 1,2,3,4$$

If one defines

$$S_{1k} = \frac{\partial S_1}{\partial \sigma_k}$$

then

$$S_{1k} = \frac{1}{G}\ w_k\, y + w\, y_k - \frac{2\, w\, y\, G_k}{G}$$

where

$$w = \frac{Q}{2pq}$$

$$w_k = \frac{1}{2p^2q^2} \left(pq\, Q_k - Q\,(p_k q + q p_k)\right)$$

$$y = \sum_{\ell=1}^{3} (\delta_\ell\, \eta_\ell + \gamma_\ell\, \xi_\ell)$$

$$y_1 = \sum_{\ell=1}^{3} \left(\ell\,(\delta_\ell\, \xi_\ell - \gamma_\ell\, \eta_\ell) + \delta_{\ell 1}\eta_\ell + \gamma_\ell \xi_\ell\right)$$

$$y_k = \sum_{\ell=1}^{3} (\delta_{\ell k}\eta_\ell + \gamma_{\ell k}\xi_\ell) \quad k = 2,3,\ldots,8$$

$$G = \sigma_5 - \frac{1}{2}(\sigma_2^{\,2} + \sigma_6^{\,2})$$

$$G_k = 0 \quad \text{for } k = 1,3,4,7,8$$

$$G_2 = -\sigma_2$$

$$G_5 = 1$$

$$G_6 = -\sigma_6$$

Here p, p_k, q, q_k, Q, Q_k, δ_ℓ, η_ℓ, γ_ℓ, ξ_ℓ and $\delta_{\ell k}$, $\gamma_{\ell k}$ are displayed

$$p = \frac{1}{\mu}\left[-\frac{1}{2}\left(\sigma_2^{\,2} + \sigma_6^{\,2}\right) + \frac{\mu}{\sqrt{2\sigma_8}}\right]^2$$

$$p_2 = -2\frac{\sqrt{\mu p}}{\mu}\,\sigma_2$$

$$p_6 = -2\frac{\sqrt{\mu p}}{\mu}\,\sigma_6$$

$$p_8 = -2\frac{\sqrt{\mu p}}{(2\sigma_8)^{3/2}}$$

$$p_k = 0 \quad \text{for } k = 1,3,4,5,7$$

ORIGINAL PAGE IS OF POOR QUALITY

$$q = -\frac{1}{2}(\sigma_6^{\,2} + \sigma_2^{\,2} - \sigma_5) + \frac{\mu}{2\sqrt{2\sigma_8}}$$

$$q_2 = -\sigma_2$$

$$q_5 = \frac{1}{2}$$

$$q_8 = -\frac{\mu}{2}\frac{1}{(2\sigma_8)^{3/2}}$$

$q_k = 0$ for $k = 1,3,4,6,7$

$$Q = \left\{\frac{\sigma_8}{\mu^2}\left[\frac{2\mu}{\sqrt{2\sigma_8}} - \frac{1}{2}\left(\sigma_2{}^2 + \sigma_6{}^2\right)\right]\right\}^{1/2}$$

$$Q_2 = -\frac{\sigma_8\sigma_2}{2Q\mu^2}$$

$$Q_6 = -\frac{\sigma_8\sigma_6}{2Q\mu^2}$$

$$Q_8 = \frac{\sqrt{\mu p}}{2Q\mu^2}$$

$Q_k = 0$ for $k = 1,3,4,5,7$

$$\delta_1 = \frac{B}{3}\sigma_6 - \frac{1}{2}\left(\sigma_6 c - \sigma_2 s\right)$$

$$\delta_{12} = \frac{s}{2} + \frac{\sigma_6}{3} B_2 - \frac{1}{2}\left(\sigma_6 c_2 - \sigma_2 s_2\right)$$

$$\delta_{16} = \frac{B}{3} - \frac{c}{2} + \frac{\sigma_6}{3} B_6 - \frac{1}{2}\left(\sigma_6 c_6 - \sigma_2 s_6\right)$$

$\delta_{1k} = \dfrac{\sigma_6}{3} B_k - \dfrac{1}{2}\left(\sigma_6 c_k - \sigma_2 s_k\right)$ for $k = 1,3,4,5,7,8$

$$\gamma_1 = \frac{B}{3}\sigma_2 + \frac{1}{2}\left(\sigma_6 s + \sigma_2 c\right)$$

$$\gamma_{12} = \left(\frac{B}{3} + \frac{C}{2}\right) + \frac{\sigma_2}{3} B_2 + \frac{1}{2} (\sigma_6 s_2 + \sigma_2 c_2)$$

$$\gamma_{16} = \frac{s}{2} + \frac{\sigma_2}{3} B_6 + \frac{1}{2} (\sigma_6 s_6 + \sigma_2 c_6)$$

$$\gamma_{1k} = \frac{\sigma_2}{3} B_k + \frac{1}{2} (\sigma_6 s_k + \sigma_2 c_k) \quad \text{for} \quad k = 1,3,4,5,7,8$$

$$\delta_2 = -\frac{c}{2Q}$$

$$\delta_{2k} = \frac{1}{2Q}\left(\frac{c}{Q} Q_k - c_k\right) \text{ for } \quad k = 1,2,3,\ldots,8$$

$$\gamma_2 = \frac{s}{2Q}$$

$$\gamma_{2k} = -\frac{1}{2Q}\left(\frac{s}{Q} Q_k - s_k\right) \text{ for } \quad k = 1,2,3,\ldots,8$$

$$\delta_3 = -\frac{1}{6} (\sigma_2 s + \sigma_6 c)$$

$$\delta_{32} = -\frac{1}{6} (\sigma_2 s_2 + \sigma_6 c_2 + s)$$

$$\delta_{36} = -\frac{1}{6} (\sigma_2 s_6 + \sigma_6 c_6 + c)$$

$$\delta_{3k} = -\frac{1}{6} (\sigma_2 s_k + \sigma_6 c_k) \quad \text{for} \quad k = 1,3,4,5,7,8$$

$$\gamma_3 = \frac{1}{6} (\sigma_6 s - \sigma_2 c)$$

ORIGINAL PAGE IS
OF POOR QUALITY

$$\gamma_{32} = \frac{1}{6} (\sigma_6 s_2 - \sigma_2 c_2 - c)$$

$$\gamma_{36} = \frac{1}{6}\,(\sigma_6 s_6 - \sigma_2 c_6 + s)$$

$$\gamma_{3k} = \frac{1}{6}\,(\sigma_6 s_k - \sigma_2 c_k) \quad \text{for} \quad k = 1,3,4,5,7,8$$

$$\eta_\ell = \sin \ell\sigma_1$$

$$\xi_\ell = \cos \ell\sigma_1 \quad \text{for} \quad \ell = 1,2,3$$

Here c, s, c_k, s_k, B, B_k, H and H_k are displayed

$$c = (G + H)\left(\frac{\sigma_7{}^2 - \sigma_3{}^2}{2}\right)$$

$$c_3 = \frac{H_3 c}{(G + H)} - (G + H)\,\sigma_3$$

$$c_7 = \frac{H_7 c}{(G + H)} + (G + H)\,\sigma_7$$

$$c_k = \frac{G_k + H_k}{(G + H)}\,c \quad \text{for } k = 1,2,4,5,6,8$$

$$s = -(G + H)\,\sigma_3\,\sigma_7$$

$$s_3 = \frac{H_3 s}{G + H} - (G + H)\,\sigma_7$$

$$s_7 = \frac{H_7 s}{G + H} - (G + H)\,\sigma_3$$

$$s_k = \frac{(G_k + H_k)}{(G + H)}\,s \quad \text{for } k = 1,2,4,5,6,8$$

$$B = G^2 - 3H^2$$

$$B_k = 2(GG_k - 3HH_k) \quad \text{for} \quad k = 1,2,3,\ldots,8$$

$$H = G - \frac{1}{2}\left(\sigma_3^{\,2} + c_7^{\,2}\right)$$

$$H_3 = -\sigma_3$$

$$H_7 = -\sigma_7$$

$$H_k = G_k \quad \text{for } k = 1,2,4,5,6,8$$

Abbreviations used in the integration of the primed system

$$A_4 = \frac{\varepsilon}{2} f_4\left(b - \frac{2}{3}\right) + \frac{\mu}{(2\sigma_8)^{3/2}}$$

$$A_3 = \frac{\varepsilon}{2} f b_3$$

$$A_2 = \frac{\varepsilon}{2}\left[f_2\left(b - \frac{2}{3}\right) + f b_2\right] + A_3$$

$$A_1 = 1 + \frac{\varepsilon}{2} f_1\left(b - \frac{2}{3}\right) + A_2$$

$$f = \frac{1}{pq}$$

$$f_1 = \frac{f^2}{\mu}\left(\frac{1}{2}\mu p + 2q\sqrt{\mu p}\right)$$

$$f_2 = -\frac{f^2}{\mu}\left(\mu p + 2q\sqrt{\mu p}\right)$$

$$f_4 = \frac{f^2}{(2\sigma_4)^{3/2}}\left(\frac{1}{2}\mu p + 2q\sqrt{\mu p}\right)$$

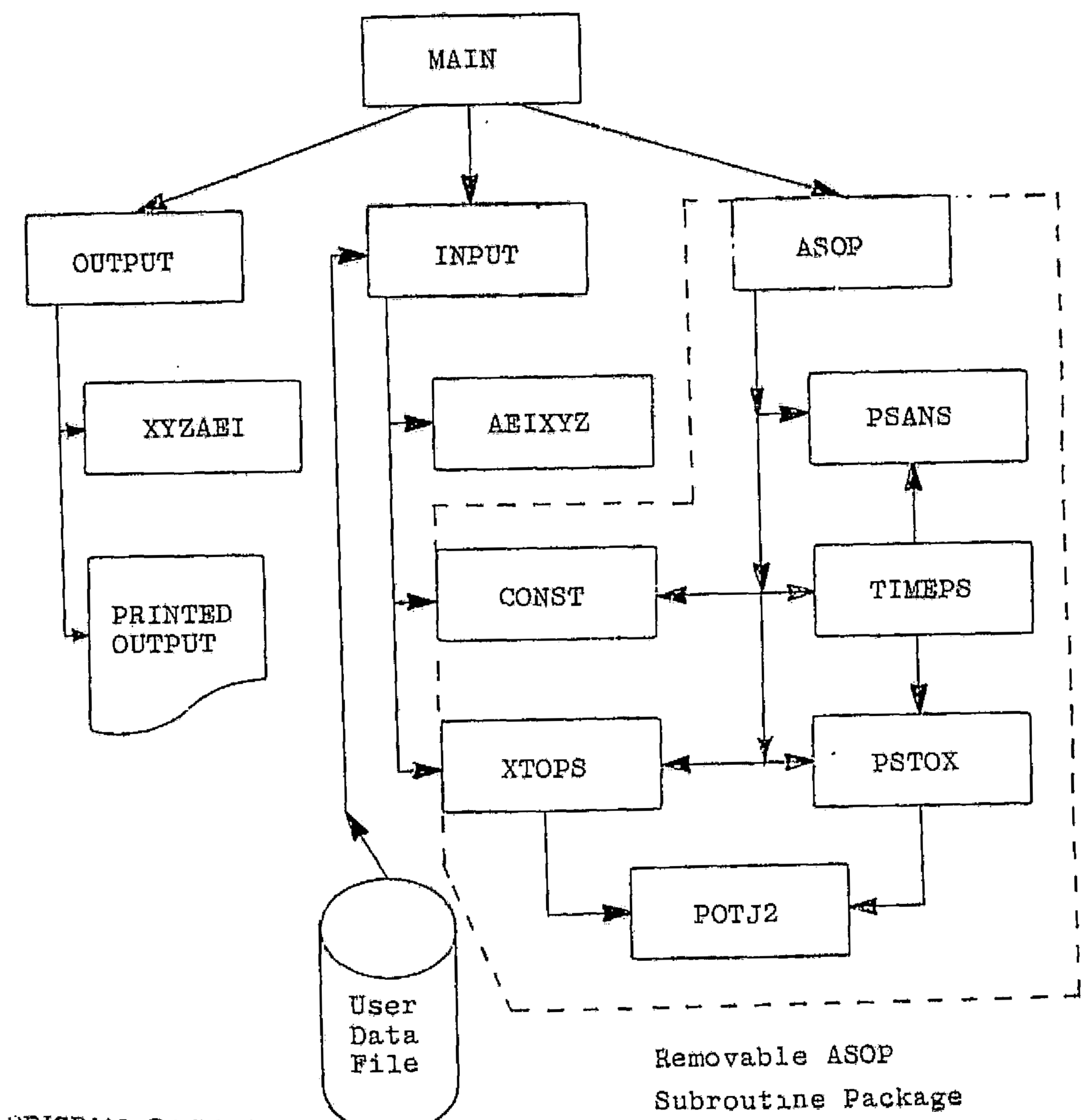
$$b = 1 - \frac{H^2}{G^2}$$

$$b_2 = \frac{2}{G}\left(\frac{H}{G}\right)^2$$

$$b_3 = -\frac{2}{G}\left(\frac{H}{G}\right)$$

APPENDIX B

DATA FLOW IN ASOP



ORIGINAL PAGE IS
OF POOR QUALITY

NASA-JSC