

MPAD REPORT CONTROL
COPY
DO NOT REMOVE

MSC INTERNAL NOTE NO. 64-FM-57

PROJECT GEMINI

LOGIC FOR REAL TIME COMPUTATION OF RELATIVE VEHICULAR QUANTITIES
USED BY THE TERMINAL PHASE SUBPROCESSOR AND
GEMINI COMPUTER CHECKLIST PROCESSOR

Prepared by:

Jerome A. Bell
Jerome A. Bell

Approved:

Edgar C. Lineberry
Edgar C. Lineberry
Chief, Rendezvous Analysis Branch

Approved:

John P. Mayer
John P. Mayer
Chief, Mission Planning and Analysis Division

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

MANNED SPACECRAFT CENTER

HOUSTON, TEXAS

November 26, 1964

TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
1. Introduction	1
2. Discussion of Subroutine COORD	1
2.1 Input for COORD	1
2.2 Output	2
3. Subroutine ROTATE	2
3.1 Input	3
3.2 Output	3
4. Subroutine RELPOS	3
4.1 Input	4
4.2 Output	5
5. Subroutine RANRAT	5
5.1 Input	5
5.2 Output	6
6. Subroutine EMPULS	6
6.1 Input	7
6.2 Output	8
7. Subroutine ROUND	8
7.1 Input	8
7.2 Output	8
8. Subroutine APPLY	9
8.1 Input	9
8.2 Output	10

9. Detailed Flow Charts	11
9.1 Subroutine COORD	11
9.2 Subroutine ROTATE	12
9.3 Subroutine RELPOS	13
9.4 Subroutine RANRAT	14
9.5 Subroutine EMPULS	15
9.6 Subroutine ROUND	16
9.7 Subroutine APPLY	17
10. References	18

1. Introduction

The subroutines included in this paper were developed for use in the Terminal Phase Subprocessor (TPS) and Gemini Computer Checklist Processor (GCCP). It was found that a given block of equations or a combination of blocks were repeated throughout the logic flow of the TPS and GCCP. For this express reason, the seven following subroutines were written. They are: COORD, ROTATE, RELPOS, RANRAT, EMPULS, ROUND, and APPLY. In most instances, output from one subroutine is fed directly as input into another. The AEG, the discussion of which will be found in Ref. 1, is utilized by the TPS and GCCP as an orbital ephemeris generator; the orbital elements obtained from the AEG will be used to compute the various display parameters.

2. Discussion of Subroutine COORD

This subroutine's only function is to convert orbital elements into earth centered inertial coordinates and velocity components. This will enable the TPS and the GCCP to do the following:

- o Compute the relative position of one vehicle in the other vehicle's coordinate system (The coordinate systems will be discussed in 3.0) by calling subroutine ROTATE.
- o Compute range rate by calling subroutine RANRAT.
- o Compute orbital elements after an impulsive maneuver by calling subroutine APPLY.

It should be stated that inertial velocity components are not always required each time this subroutine is called; therefore by setting a control number external to this subroutine, the inertial velocity computations may be omitted. It should also be pointed out that the output from this subroutine is not dimensioned; therefore the output corresponds to the vehicle whose elements were input. The proper vehicle is obtained by carrying its identification number from the main program into the subroutine.

2.1 Input for COORD

2.1.1 The constant input for COORD is μ (earth's gravitational constant).

2.1.2 Variable input

a	semimajor axis
e	eccentricity
I	inclination
g	argument of perigee
h	longitude of ascending node
U	argument of latitude
R	radius from center of earth
L	vehicle for which the above elements are inputs
K ₁₀₁	control number for computing inertial velocity components

2.2 Output

X _{IN} , Y _{IN} , Z _{IN}	inertial coordinates for the input vehicle
\dot{X}_{IN} , \dot{Y}_{IN} , \dot{Z}_{IN}	inertial velocity components for the input vehicle

3. Discussion of Subroutine ROTATE

Prior to the discussion of this subroutine, it is well to define the target centered and spacecraft centered orthogonal coordinate system. The target centered system is composed of a positive Y axis pointing radially away from the center of the earth, the positive X axis opposite the direction of motion and the positive Z forming a right handed coordinate system. For the spacecraft system, positive X and Y are opposite that of the target system and again the positive Z forms a right handed system.

Simply stated, this subroutine computes the relative position of one vehicle with respect to the other vehicle's coordinate system. This is accomplished in the following manner: Assume that the relative position of the spacecraft with respect to the target's coordinate system is desired. First, the inertial coordinates (X_{IN} , Y_{IN} , Z_{IN}) of the spacecraft are determined by using Subroutine COORD. Then this information along with the orbital elements

(I, h, U, R) of the target and the control number (L) for the coordinate system that is to be used (L = 1 for the target coordinate system and L = 2 for the spacecraft coordinate system) is inserted into ROTATE. ROTATE will then compute the relative position of the spacecraft with respect to the target coordinate system. This procedure can be reversed therefore, obtaining the relative position of the target with respect to the spacecraft coordinate system.

The main program will use the output from ROTATE in the following manner:

- o To determine the phase angle between the spacecraft and target, measured in the target orbital plane.
- o To determine positions to be used in the guidance equations by calling RELPOS.
- o To determine range rate by calling RANRAT.

3.1 Input for ROTATE

3.1.1 There are no constant variables to be input into ROTATE

3.1.2 Variable inputs

I	inclination
h	longitude of ascending node
U	argument of latitude
R	radius from center of earth
L	vehicle coordinate system
X_{IN} , Y_{IN} , Z_{IN}	inertial coordinates of vehicle to be rotated

3.2 Output

X_{RL} , Y_{RL} , Z_{RL}	relative position of rotated vehicle in the other vehicle's coordinate system
--------------------------------	---

4. Discussion of Subroutine RELPOS

Prior to the discussion of this subroutine, the technique by which the on-board computer obtains proper relative positions between the spacecraft and

target will be outlined. The spacecraft radar measures the range and the sine of the relative azimuth and elevation angles between the two vehicles. This data is then converted into relative positions of the target measured in the spacecraft coordinate system. The guidance equations, however, utilize the target coordinate system. Therefore, the assumption is made that the proper relative positions may be obtained by the following expression:

$$\begin{bmatrix} X_T \\ Y_T \\ Z_T \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} X_S \\ Y_S \\ Z_S \end{bmatrix}$$

Note: The subscript T denotes the target coordinate system and subscript S denotes spacecraft coordinate system.

RELPOS is used to generate the radar data in the following manner:

The relative position of the target with respect to the spacecraft system is determined in ROTATE. The output from ROTATE is then inserted into RELPOS. The azimuth, elevation, and range are then computed in RELPOS. Utilizing this calculated data and the above transformation, the information necessary for the guidance equations is established.

The guidance equations utilizes a series of positions. Therefore, for a given maneuver, this subroutine must be called several times. Since it is not the intention to elaborate on the guidance equations (see 5.) it will suffice to say that the output from this subroutine must be stored according to an identification number. This number is set in the main program and carried into this subroutine.

4.1 Input for RELPOS

4.1.1 The constant input for RELPOS is R_T (Target radius from the center of the earth to be used in the guidance equations). This input may be changed during the flight.

4.1.2 Variable Input

X_{RL}, Y_{RL}, Z_{RL}

relative position of the target in the space-craft coordinate system

K_{103}

identification number for the output from this subroutine

4.2 Output

$X_{C_K}_{103}$ $Y_{C_K}_{103}$ $Z_{C_K}_{103}$ simulated radar measurements of the relative position of the target in the spacecraft system

5. Discussion of Subroutine RANRAT

At various stages throughout the logic of the main program, the need occurs for the computation of the rate of change of the range between the two vehicles. It is needed not only for display, but also for obtaining solutions for other displays.

The basis of this subroutine is derived from the expression for range rate in terms of relative position as found on page 5 of Ref. 2. As shown on this page, range rate is a function of both relative position and velocity.

It has been shown from paragraph 3 (ROTATE) that the relative position may be found at any instant. The inertial velocities may also be found from paragraph 2 (COORD). Therefore, RANRAT calls COORD and ROTATE in order to establish the inertial velocities of both vehicles and the relative position of the spacecraft in the target coordinate system. The inertial velocities are then rotated from the inertial frame to the target frame. It should be noted that ROTATE may only be used for displacement; therefore, another matrix must be used for the rotation of velocities. The inertial relative velocity components, defined as V_u , V_v , V_w on page 4 of Ref. 2, can then be determined by taking the difference between the rotated inertial velocity components. Using the above information, range rate is then computed.

5.1 Input for RANRAT

5.1.1 The constant input for RANRAT is μ (earth's gravitation constant)

5.1.2 Variable Input

a	semimajor axis
e	eccentricity
I	inclination
g	argument of perigee
h	longitude of ascending node

R_L	radius from center of earth
U	argument of latitude
K_{101}	control number for computing inertial velocity components
L	vehicle coordinate system

5.2 Output

r range
 \dot{r} range rate
 X_{IN}, Y_{IN}, Z_{IN} spacecraft inertial coordinates

5.3 Constant and Variable Input Required for

5.3.1 COORD

$a, e, I, g, h, U, R_L, K_{101}, \mu, K_{107}$

5.3.2 ROTATE

I, h, U, R_L, L

5.3.3 For complete symbol definition for 5.3.1 and 5.3.2 see sections 2. and 3.

6. Discussion of Subroutine EMPULS

This subroutine functions as one of the most important parts of the TPS and GCCP logic. Its duty is to compute the terminal phase maneuvers exactly the same as the onboard computer. The equations and technique may be found in Ref. 3; however, a brief discussion of the operational procedure will be included here.

For the initial impulse, 11 data points (the output from RELPOS) are collected at 100 second intervals. Using the first and last data point and the time between the two points, the actual relative velocity at the last point is computed. This is repeated using the second and last, third and last, etc. until seven relative velocities have been computed. The average of the seven is assumed to be the actual relative velocity at the last data point. Having the position and velocity, the position at a point 250 seconds ahead in time is predicted. Considering this new point as the last data point, the procedure previously described is repeated to obtain the relative velocity at this new point. From this point, the

velocity needed to rendezvous within a given transfer angle is computed. The difference between the actual relative velocity and that which is needed is the impulse to be applied by the astronaut.

For intermediate corrections, seven measurements are made at 20 second intervals starting 200 seconds after the midpoint of the initial burn. After some predetermined target angular travel, a final data point is collected. The actual relative velocity and the relative velocity needed to rendezvous within a resultant transfer angle are then computed similar to procedure used for the initial impulse. The only exception is that there is no predicting ahead for 250 seconds. This is only done for the initial impulse.

The next intermediate correction is computed starting 200 seconds from the time that the last data point was collected. The same procedure is followed in computing all of the intermediate corrections. At the time of the last intermediate correction which is determined by some preset minimum value for the target angular travel between this correction and the time of rendezvous, the magnitude of the braking is computed.

The proper number of data points are entered into EMPULS from the main program along with a control number. This control number determines whether the maneuver is an intermediate correction or the initial impulse. The logic is set to function as was described for the operational procedure. Output is used for displays and for input into the Subroutine APPLY which will be described later.

6.1 Input for EMPULS

6.1.1 The constant inputs for EMPULS are:

- o μ (earth's gravitational constant)
- o The following constants may be changed in flight.

ω_t target central travel angle between terminal phase initiation and rendezvous

$\Delta \omega_t$ target central travel angle between intermediate corrections

R_T target radius for use in the guidance equations
target orbital period

$\omega_{t_{LLM}}$ smallest transfer angle for which an intermediate correction will be made

6.1.2 Variable input

X_C, Y_C, Z_C	simulated radar measurements of relative position of the spacecraft in the target coordinate system
K_{200}	control number which determines whether the maneuver is an initial or intermediate impulse
K_{105}	identification number for the last data point

6.2 Output

$\Delta \dot{X}, \Delta \dot{Y}, \Delta \dot{Z}$	thrust components to be applied by the spacecraft
$\Delta \dot{X}_I, \Delta \dot{Y}_I, \Delta \dot{Z}_I$	initial thrust components
ΔV_I	initial thrust vector
$\Delta \dot{X}_F, \Delta \dot{Y}_F, \Delta \dot{Z}_F$	final thrust components
ΔV_F	final thrust vector

7. Discussion of Subroutine ROUND

For display purposes, the attitude angles and impulses, initial and final, are to be rounded off the nearest fps and degree. This is due to the fact that the astronaut would not be able to hold his attitude closer than one degree nor apply a thrust closer than one fps. Subroutine ROUND accomplishes this. If angles are used, they must be in degrees; if only two variables are required to be rounded, a dummy variable must be inserted as this subroutine is designed to round off three variables each time it is called.

7.1 Variable inputs for Subroutine ROUND are Q_1 , Q_2 , and Q_3 which are three variables to be rounded off.

7.2 Output

The outputs are Q_1 , Q_2 , and Q_3 which are the inputs that were rounded off.

8. Discussion of Subroutine APPLY

The function of this subroutine is to impart an impulsive thrust (computed in EMPULS) to the spacecraft in order to obtain the new set of orbital elements. This is accomplished in the following way: The inertial coordinates and velocity components are obtained at the time of the impulse. The thrust components to be applied by the spacecraft are then rotated to the inertial reference frame. The new thrust components are added to the present inertial velocity components. The program now has the inertial coordinates and velocity after an impulsive maneuver. These may then be converted to the orbital elements compatible with the AEG by the equations presented in Appendix D of Ref. 4.

8.1 Input for APPLY

8.1.1 The constant input for APPLY is μ (earth's gravitational constant)

8.1.2 Variable Input

$\Delta\dot{X}$, $\Delta\dot{Y}$, $\Delta\dot{Z}$	impulse to be applied to spacecraft measured in spacecraft coordinate system
X_{IN} , Y_{IN} , Z_{IN}	spacecraft inertial coordinates
\dot{X}_{IN} , \dot{Y}_{IN} , \dot{Z}_{IN}	spacecraft inertial velocity components prior to the impulse
L	spacecraft vehicle number

The following refer to spacecraft orbital elements prior to the impulse:

a	semimajor axis
e	eccentricity
I	inclination
g	argument of perigee
h	longitude of ascending node
l	mean anomaly
U	argument of latitude

8.2 Output

The following refer to the spacecraft elements after the impulse and are defined above:

a
e
I
g
h
l
U

9. Detailed Flow Charts

9.1 Subroutine COORD

SUBROUTINE COORD INPUT/OUTPUT
 $a, e, I, g, h, U, R_L, X_{IN}, Y_{IN}, Z_{IN}, L, \dot{X}_{IN}, \dot{Y}_{IN}, \dot{Z}_{IN}, K_{101}, \mu$

$$X_{IN} = R_L [\cos U_L \cosh h_L - \cos I_L \sin U_L \sin h_L]$$

$$Y_{IN} = R_L [\cos U_L \sin h_L + \cos I_L \sin U_L \cosh h_L]$$

$$Z_{IN} = R_L \sin U_L \sin I_L$$

$K_{101} : 0 \leq$

$$f = U_L - g_L$$

$$\dot{X}_{IN} = \sqrt{\frac{\mu}{a_L(1-e_L^2)}} [(e_L + \cos f)(-\sin g_L \cosh h_L - \cos I_L \sin h_L \cos g_L) - \sin f (\cos g_L \cosh h_L - \cos I_L \sin h_L \sin g_L)]$$

$$\dot{Y}_{IN} = \sqrt{\frac{\mu}{a_L(1-e_L^2)}} [(e_L + \cos f)(-\sin g_L \sin h_L + \cos I_L \cosh h_L \cos g_L) - \sin f (\cos g_L \sin h_L + \cos I_L \cosh h_L \sin g_L)]$$

$$\dot{Z}_{IN} = \sqrt{\frac{\mu}{a_L(1-e_L^2)}} [(e_L + \cos f) \sin I_L \cos g_L - \sin f \sin I_L \sin g_L]$$



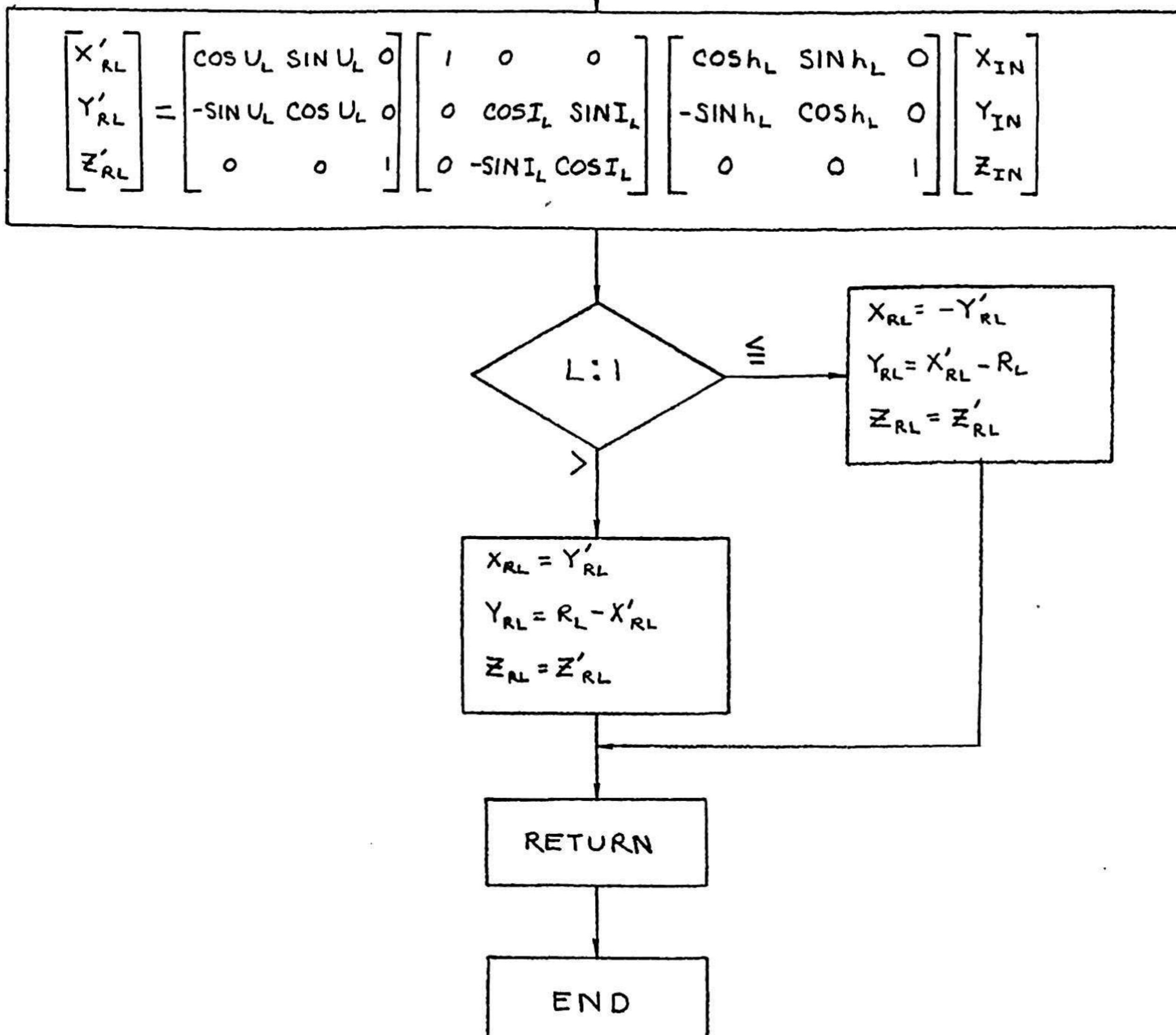
RETURN

END

9.2 Subroutine ROTATE

- 12 -

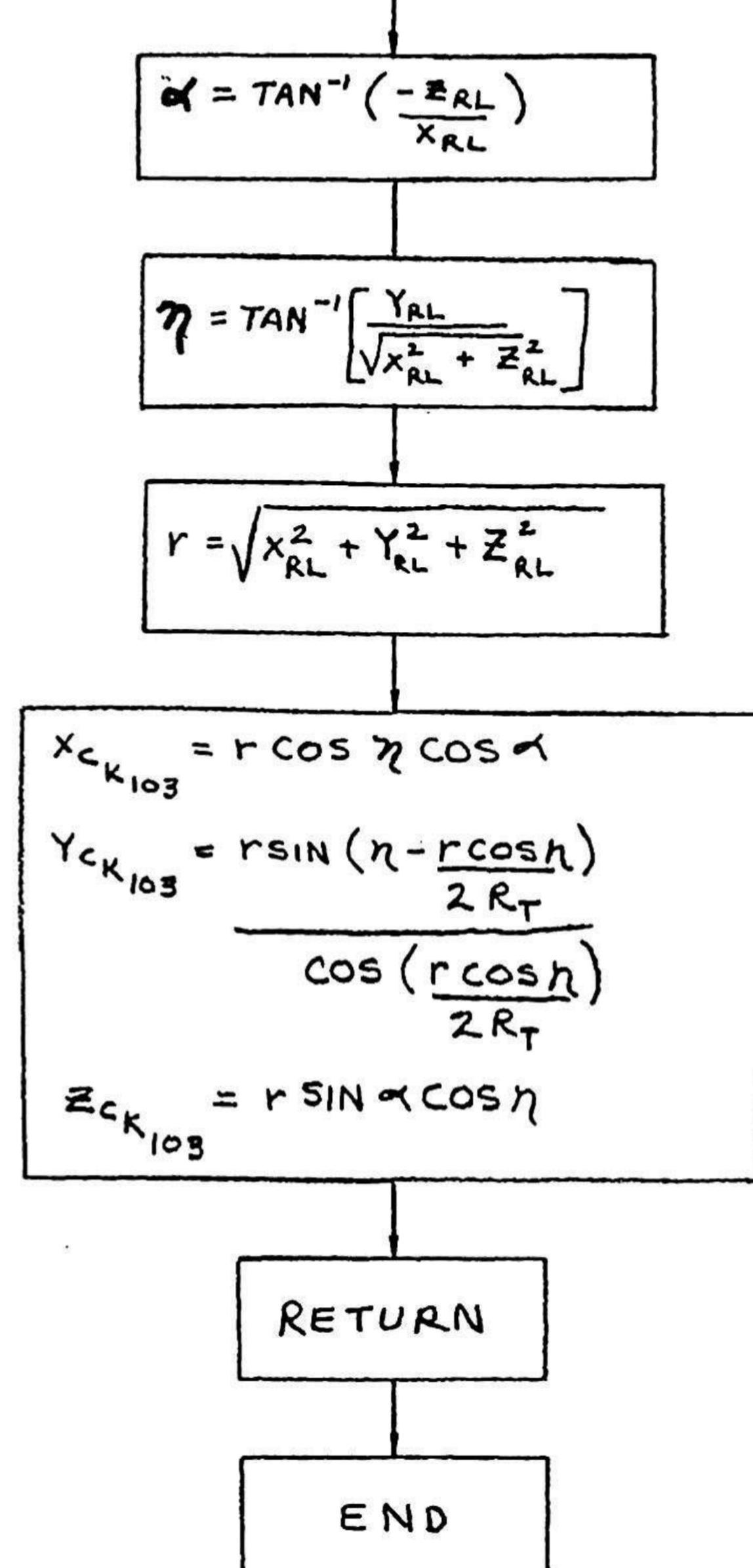
SUBROUTINE ROTATE INPUT/OUTPUT
 $I, h, U, R_L, X_{IN}, Y_{IN}, Z_{IN}, L, X_{RL}, Y_{RL}, Z_{RL}$



9.3 Subroutine RELPOS

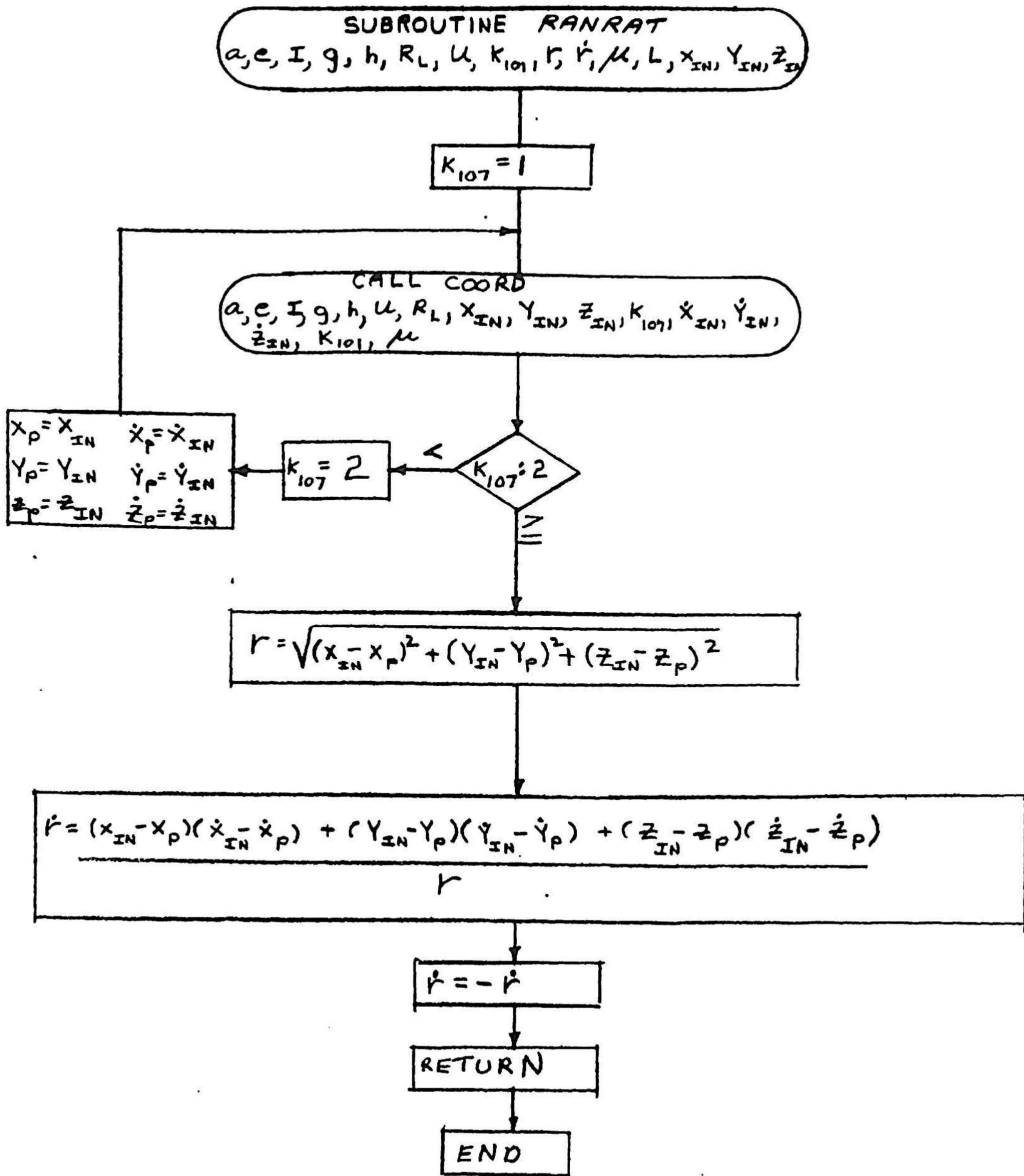
- 13 -

SUBROUTINE RELPOS INPUT/OUTPUT
 $X_{RL}, Y_{RL}, Z_{RL}, R_T, X_C, Y_C, Z_C, K_{103}$



9.4 Subroutine RANRAT

-14-

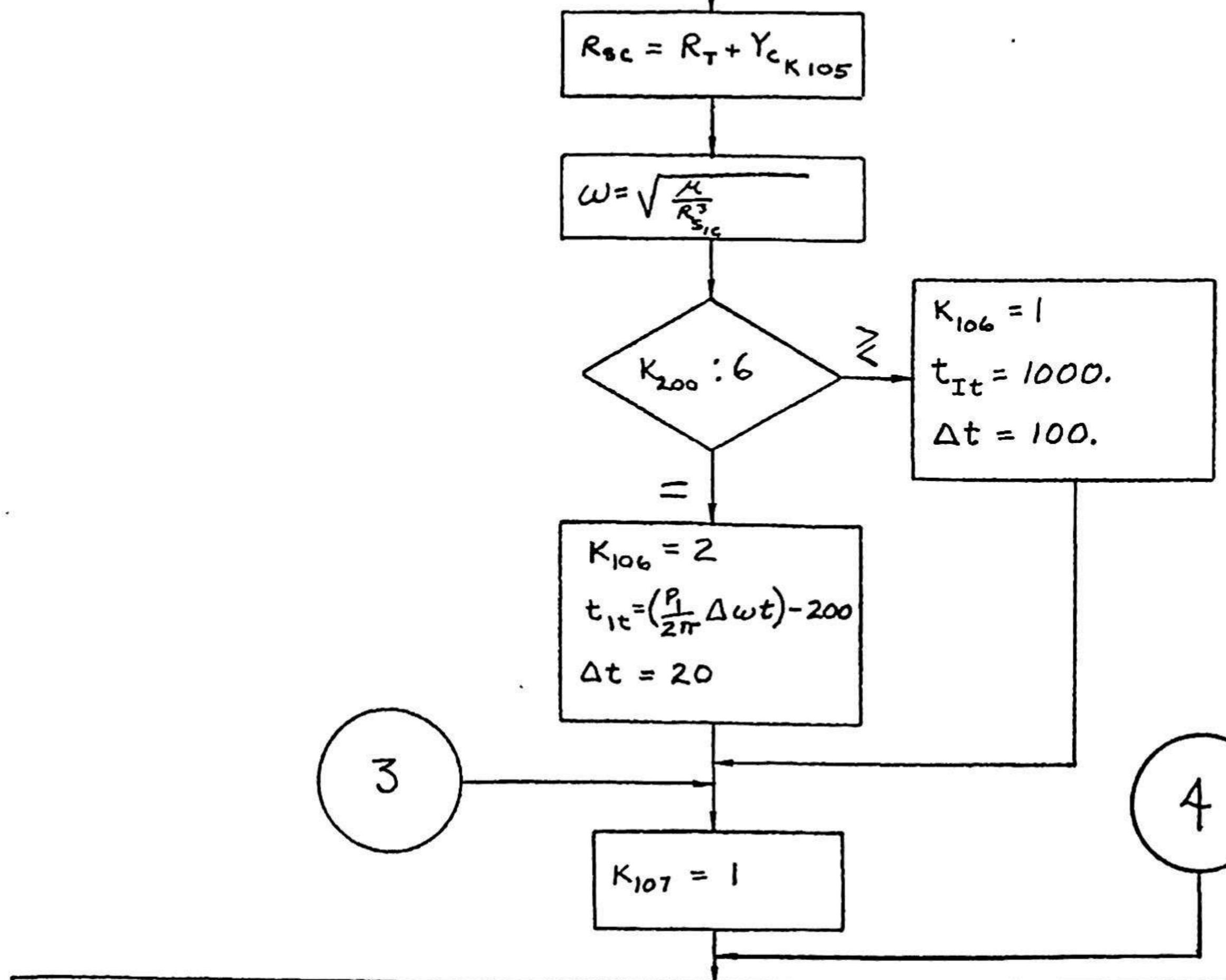


PAGE	OF	PAGES
1		1

9.5 Subroutine EMPULS

- 15 -

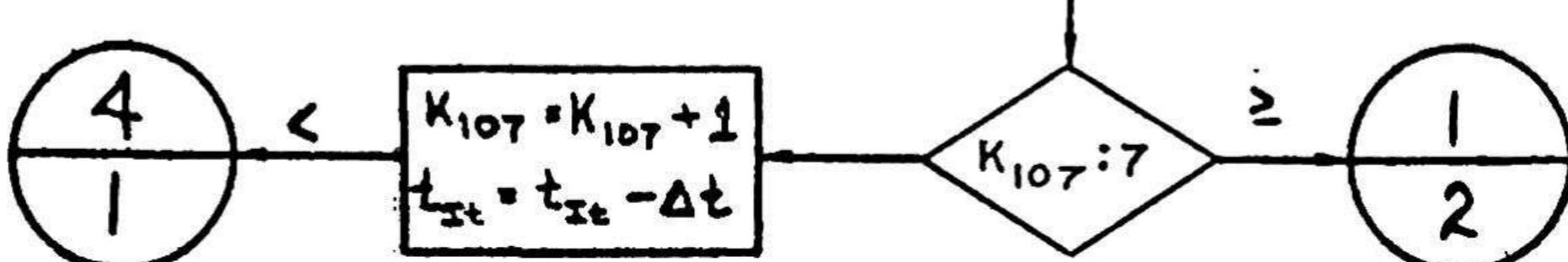
SUBROUTINE EMPULS INPUT/OUTPUT
 $x_c, Y_c, Z_c, \omega t, K_{105}, \Delta \omega t, K_{200}, R_T, \Delta \dot{x}, \Delta \dot{y}, \Delta \dot{z}, \Delta v_I,$
 $\Delta \dot{x}_I, \Delta \dot{y}_I, \Delta \dot{z}_I, \Delta v_F, \Delta \dot{x}_F, \Delta \dot{y}_F, \Delta \dot{z}_F, P_1, M, \omega c_{lim}$

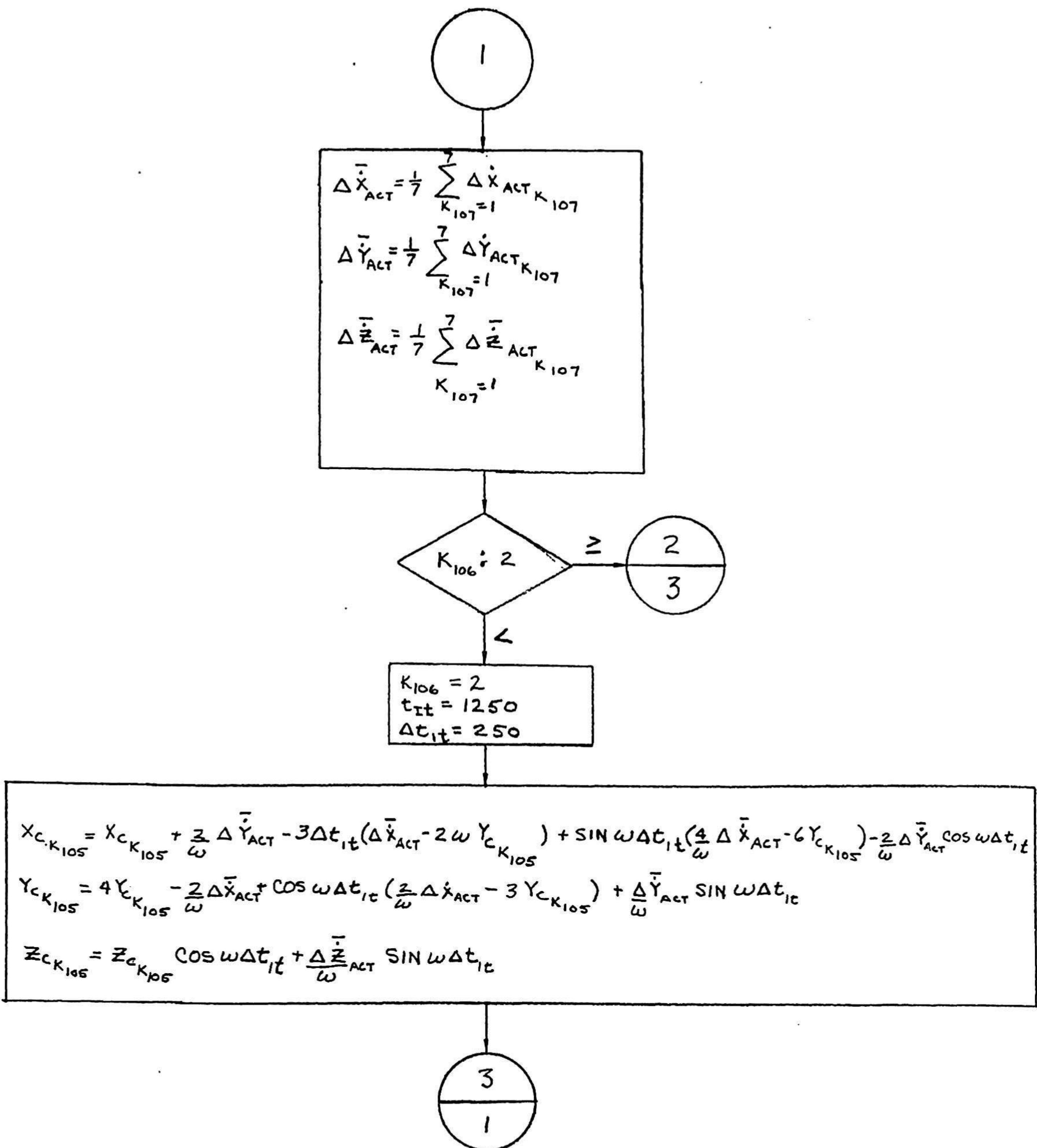


$$\Delta \dot{x}_{ACT_{K_{107}}} = \omega \left[\frac{(x_{c_{K_{107}}} - x_{c_{K_{105}}}) \sin \omega t_{It} + 2 Y_{c_{K_{107}}} (1 - \cos \omega t_{It}) + Y_{c_{K_{105}}} [6 \omega t_{It} \sin \omega t_{It} - 14(1 - \cos \omega t_{It})]}{3 \omega t_{It} \sin \omega t_{It} - 8(1 - \cos \omega t_{It})} \right]$$

$$\Delta \dot{y}_{ACT_{K_{107}}} = \omega \left[\frac{2(x_{c_{K_{105}}} - x_{c_{K_{107}}}) (1 - \cos \omega t_{It}) + Y_{c_{K_{107}}} (4 \sin \omega t_{It} - 3 \omega t_{It}) + Y_{c_{K_{105}}} (3 \omega t_{It} \cos \omega t_{It} - 4 \sin \omega t_{It})}{3 \omega t_{It} \sin \omega t_{It} - 8(1 - \cos \omega t_{It})} \right]$$

$$\Delta \dot{z}_{ACT_{K_{107}}} = \omega \left[\frac{z_{c_{K_{105}}} \cos \omega t_{It} - z_{c_{K_{107}}}}{\sin \omega t_{It}} \right]$$





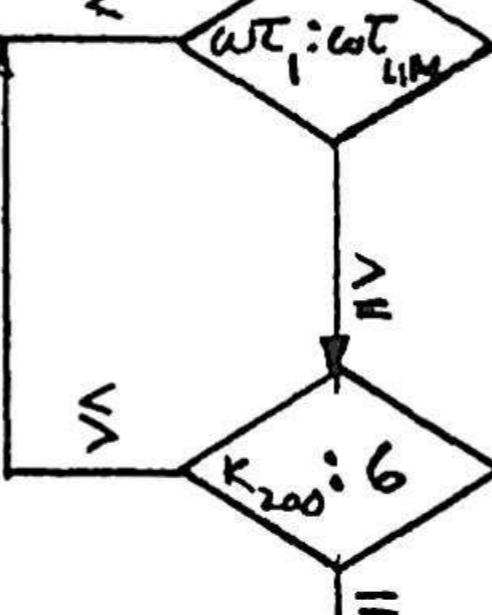
2

$$\Delta \dot{x}_{RQ} = \omega \left\{ \frac{x_c \sin \omega t + Y_c k_{105} [6 \omega t \sin \omega t - 14(1 - \cos \omega t)]}{3 \omega t \sin \omega t - 8(1 - \cos \omega t)} \right\}$$

$$\Delta \dot{y}_{RQ} = \omega \left\{ \frac{2x_c (1 - \cos \omega t) + Y_c k_{105} [4 \sin \omega t - 3 \omega t \cos \omega t]}{3 \omega t \sin \omega t - 8(1 - \cos \omega t)} \right\}$$

$$\Delta \dot{z}_{RQ} = \omega \left\{ \frac{-Z_c k_{105} \cos \omega t}{\sin \omega t} \right\}$$

$\omega t_1 = \omega t - \Delta \omega t$



... THE FINAL IMPULSE IS TO BE COMPUTED AT THE INITIAL MANEUVER AND LAST INTERMEDIATE CORRECTION

$$\Delta \dot{x}_F = -\omega \left[\frac{x_c \sin \omega t + Y_c (2 - 2 \cos \omega t)}{3 \omega t \sin \omega t - 8(1 - \cos \omega t)} \right]$$

$$\Delta \dot{y}_F = -\omega \left[\frac{2(\cos \omega t - 1)x_c + Y_c (4 \sin \omega t - 3 \omega t)}{3 \omega t \sin \omega t - 8(1 - \cos \omega t)} \right]$$

$$\Delta \dot{z}_F = -\omega \frac{Z_c k_{105}}{\sin \omega t}$$

$$\Delta V_F = \sqrt{\Delta \dot{x}_F^2 + \Delta \dot{y}_F^2 + \Delta \dot{z}_F^2}$$

THE MINUS SIGN IS
TO BE ADDED TO
 $\Delta \dot{z}_F$

$$\Delta \dot{x} = \Delta \dot{x}_{ACT} - \Delta \dot{x}_{RQ}$$

$$\Delta \dot{y} = \Delta \dot{y}_{ACT} - \Delta \dot{y}_{RQ}$$

$$\Delta \dot{z} = \Delta \dot{z}_{RQ} - \Delta \dot{z}_{ACT}$$



$$\begin{aligned} \Delta \dot{x}_I &= \Delta \dot{x} \\ \Delta \dot{y}_I &= \Delta \dot{y} \\ \Delta \dot{z}_I &= \Delta \dot{z} \end{aligned}$$

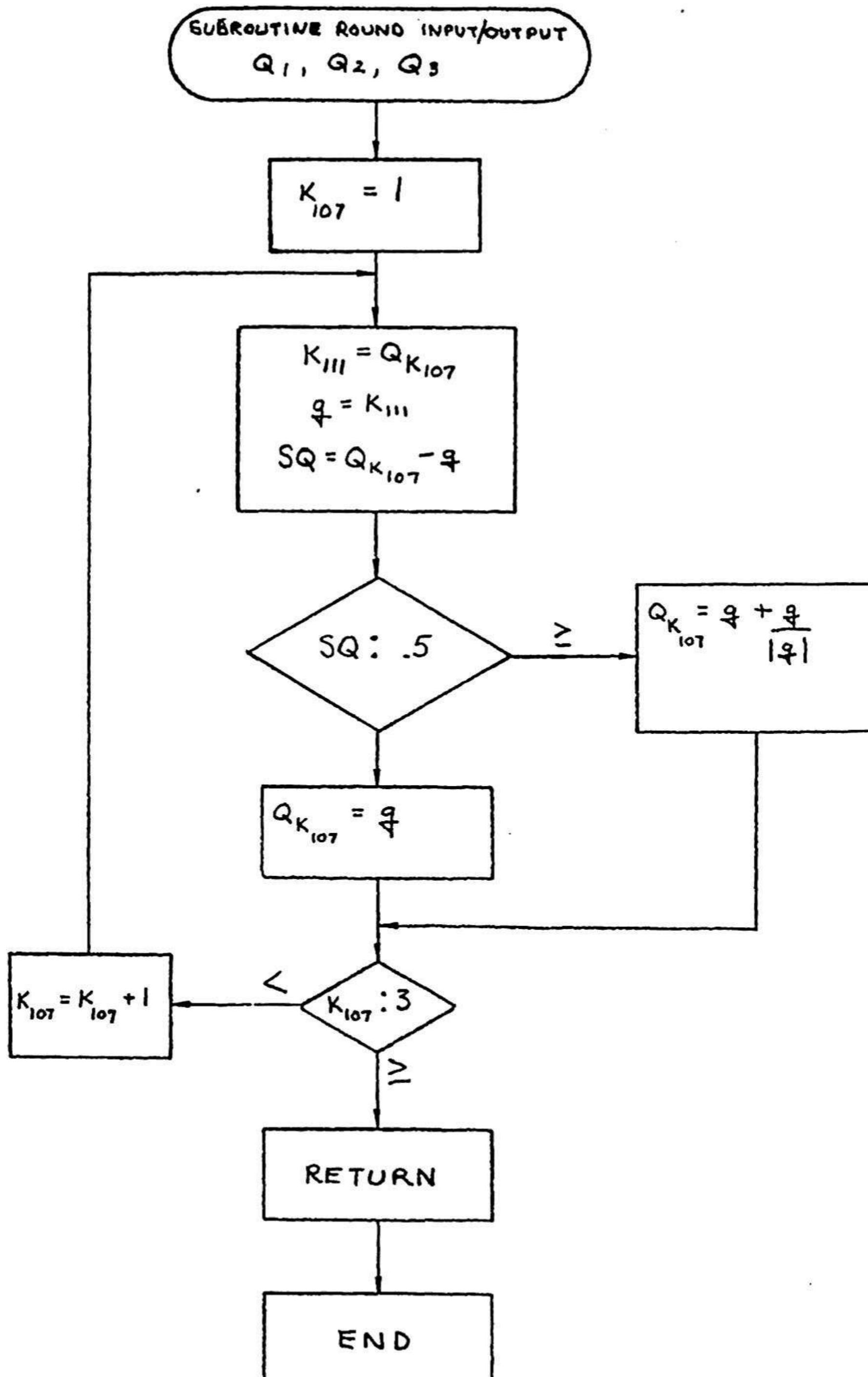
$$\Delta V_I = \sqrt{\Delta \dot{x}_I^2 + \Delta \dot{y}_I^2 + \Delta \dot{z}_I^2}$$

END

RETURN

PAGE	OF	PAGES
3		3

9.6 Subroutine ROUND



9.7 Subroutine APPLY

- 17 -

SUBROUTINE APPLY INPUT/OUTPUT
 $X_{IN}, Y_{IN}, Z_{IN}, \dot{X}_{IN}, \dot{Y}_{IN}, \dot{Z}_{IN}, a, e, I, g, h, l, U, L, \Delta\dot{x}, \Delta\dot{y}, \Delta\dot{z}, \mu$

$$\begin{bmatrix} \Delta\dot{x}_{IN} \\ \Delta\dot{y}_{IN} \\ \Delta\dot{z}_{IN} \end{bmatrix} = \begin{bmatrix} \cosh h_L - \sinh h_L & 0 \\ \sinh h_L \cosh h_L & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos I_L - \sin I_L & 0 \\ 0 & \sin I_L \cos I_L & 1 \end{bmatrix} \begin{bmatrix} \cos U_L - \sin U_L & 0 \\ \sin U_L \cos U_L & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \Delta\dot{x} \\ \Delta\dot{y} \\ \Delta\dot{z} \end{bmatrix}$$

$$\begin{aligned} \dot{x}_{IN} &= \dot{x}_{IN} + \Delta\dot{x}_{IN} \\ \dot{y}_{IN} &= \dot{y}_{IN} + \Delta\dot{y}_{IN} \\ \dot{z}_{IN} &= \dot{z}_{IN} + \Delta\dot{z}_{IN} \end{aligned}$$

$$V_{IN}^2 = \dot{x}_{IN}^2 + \dot{y}_{IN}^2 + \dot{z}_{IN}^2$$

$$R_L = \sqrt{\dot{x}_{IN}^2 + \dot{y}_{IN}^2 + \dot{z}_{IN}^2}$$

$$a_L = \frac{I}{\left(\frac{2}{R_L} - \frac{V_{IN}^2}{M} \right)}$$

*

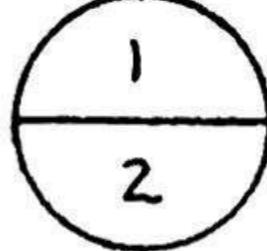
$$\begin{aligned} Q_{10} &= Y_{IN} \dot{z}_{IN} - \dot{z}_{IN} \dot{y}_{IN} \\ Q_{11} &= Z_{IN} \dot{x}_{IN} - X_{IN} \dot{z}_{IN} \\ Q_{12} &= X_{IN} \dot{y}_{IN} - Y_{IN} \dot{x}_{IN} \end{aligned}$$

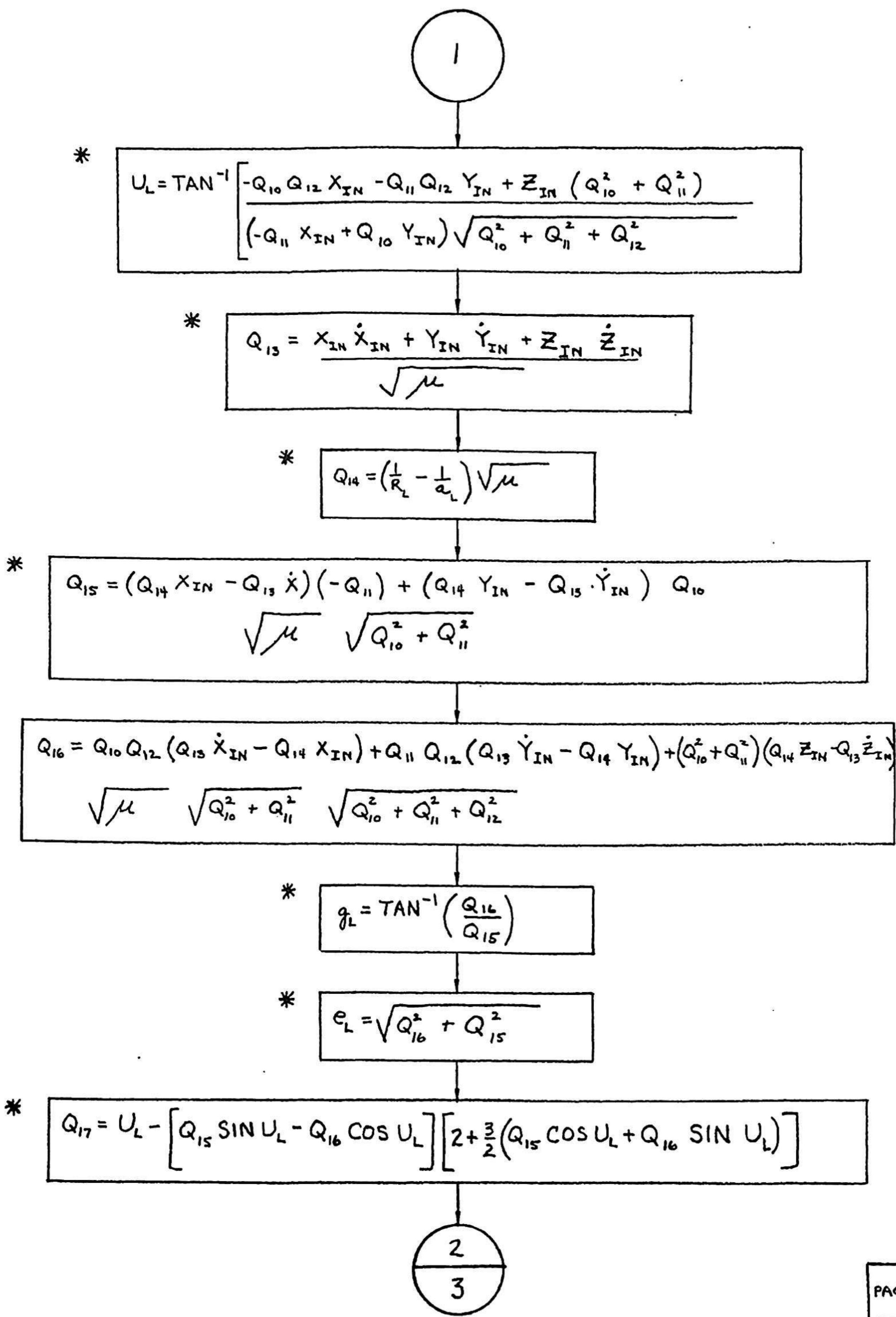
*

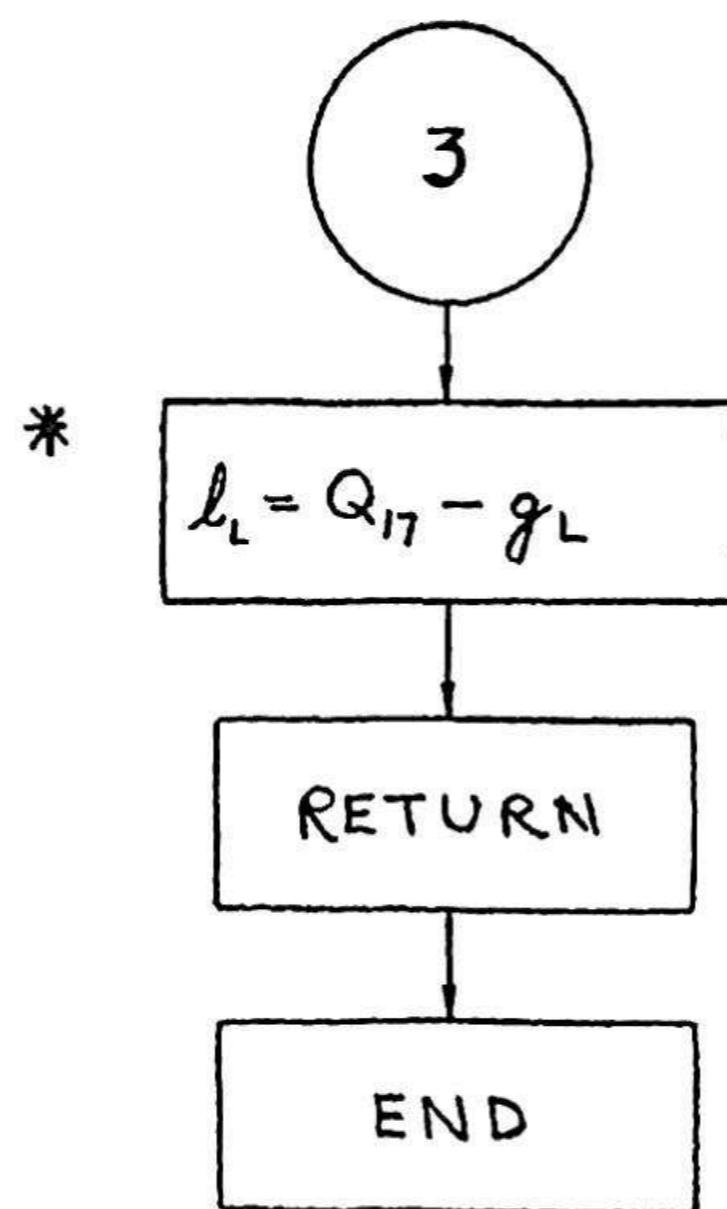
$$I_L = \tan^{-1} \left\{ \frac{\sqrt{1 - \frac{Q_{12}^2}{Q_{10}^2 + Q_{11}^2 + Q_{12}^2}}}{\frac{Q_{12}}{\sqrt{Q_{10}^2 + Q_{11}^2 + Q_{12}^2}}} \right\}$$

*

$$h_L = \tan^{-1} \left[\frac{Q_{10}}{-Q_{11}} \right]$$







* NOTE: DOUBLE PRECISION SHOULD BE EMPLOYED FOR GOOD RESULTS.

REFERENCES

1. Moore, A. J. and Reini, W. A.: Logic and Equations for Analytic Ephemeris Generation, MSC Internal Note No. 64-FM-50, Nov. 5, 1964.
2. Relative Velocity, IBM/RTCC Report 12-009, Houston, Texas, Feb. 25, 1964, pp. 4-5.
3. Gemini Catchup and Rendezvous Analysis and Simulation Report No. 2, IBM Space Guidance Center, Owego, New York, May 1, 1964. Appendix II.
4. Analytic Ephemeris Generation for Low Eccentricity, Drag-Perturbed Satellite Motion About an Oblate Planet, Space Technology Laboratories, Inc., Redondo Beach, California, Nov. 26, 1963, Appendix C.
5. Orbital Flight Manual, Orbital Mechanics, Martin Aircraft Company, July 1961, Chapter II, p. 15.