

5.4 TARGETING ROUTINES

5.4.1 GENERAL COMMENTS

The objectives of the targeting routines presented in this section are to provide a CMC capability to determine the required input target parameters and set control modes for the various powered flight guidance routines of Section 5.3. It should be noted that the CMC does not have a targeting capability for every phase of the lunar landing mission, and therefore must rely on RTCC targeting for many of the nominal and abort phases of the lunar mission. The restricted targeting capability provided by the CMC is presented in the following subsections:

5.4.2 Rendezvous Retrieval Targeting Routines

These include the Concentric Flight Plan, TPI Search, Stable Orbit and Midcourse Routines.

5.4.3 Return-to-Earth Targeting Routines

These routines are restricted to operation outside the lunar sphere of influence for abort returns to earth.

All CMC targeting programs use the Basic Reference Coordinate System defined in Section 5.1.4.1. The Rendezvous Retrieval Targeting Routines of Section 5.4.2 can be used in either earth or lunar orbits. The basic input parameters required by the targeting routines of this section are the vehicle state vector estimates determined by the navigation programs of Section 5.2 and 5.3.2.

5.4.2 Rendezvous Targeting

5.4.2.1 General

In addition to containing the nominal rendezvous approach in the Concentric Flight Plan program, the CMC contains the TPI Search program for fuel critical situations and the Stable Orbit Rendezvous program for primarily time critical situations.

The Concentric Flight Plan rendezvous scheme is illustrated in Figure 4.2-1. The following active vehicle impulsive maneuvers are used to establish a rendezvous intercept trajectory with the passive vehicle in an appropriately coplanar orbit:

1. Coelliptic Sequence Initiation (CSI)
2. Constant Differential Altitude (CDH)
3. Transfer Phase Initiation (TPI)

These maneuvers are selected by the program to satisfy the following constraints: specified times for the CSI and TPI maneuvers, a CSI ΔV which results in "coelliptic" orbits, and a specified geometry at the TPI point. The "coelliptic" or "concentric" orbits have approximately constant differential altitudes.

The TPI Search Program determines a two maneuver transfer trajectory which minimizes the total fuel used while providing a safe pericenter. The time of the first maneuver must be specified.

The Stable Orbit Rendezvous program provides a direct transfer capability between the active and passive vehicle orbits. The time of the first maneuver and a measure of the transfer time must be specified.

Both the Pre-TPI and Stable Orbit programs have Midcourse Maneuver programs to ensure that the active vehicle will rendezvous with the passive vehicle.

For the routines in this section there are two associated program numbers based on whether the CSM or the LM is the active vehicle. If the astronaut elects to make the LM active, the only change is the program number selected. All equations and program operations in Section 5.4.2 are identical for these two modes of operation. The function $SGN(X)$ used in Figures 4.2-4 and -5 is set equal to +1 if X is zero or a positive number; otherwise, it is set equal to -1.

The pre-thrust programs P-32 and P-33 use External ΔV guidance (Section

5.4-2

Revised COLOSSUS

Added GSOP #R-577 PCR # 681 Rev. 5 Date 1-7-69

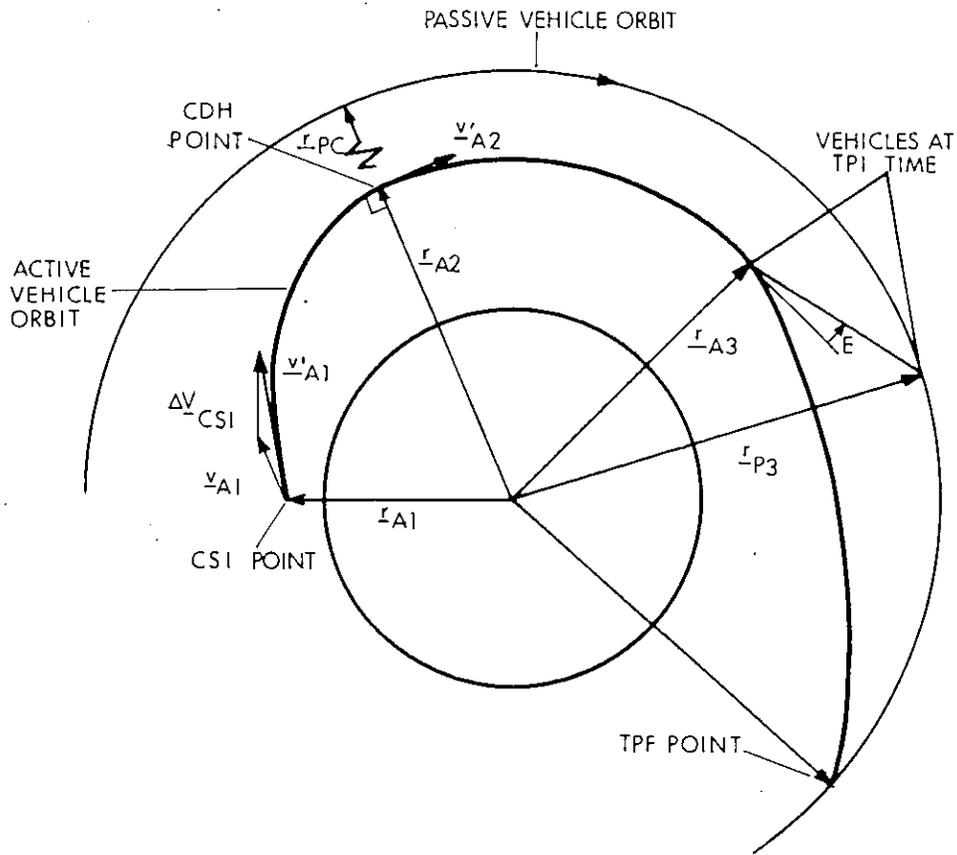


Figure 4.2-1 Concentric Rendezvous Profile

5.4-3

Revised COLOSSUS

Added GSOP # R-577 PCR # 681

Rev. 5

Date 1-7-69

5.3.3.3.1), and P-34, P-35, P-38, and P-39 use the Lambert targeting concept employing two precision offsets (Section 5.3.3.3.2) with the steering constant ϵ set equal to unity. All six of these programs use the Cross-Product Steering of Section 5.3.3.4.

5.4.2.1.1 Pre-CSI Maneuver. -- This program, corresponding to P-32 (CSM active) or P-72 (LM active) of Section 4, computes the parameters associated with the CSI and CDH maneuvers. The astronaut inputs are as follows:

1. Choice of active vehicle (P-32 CSM, P-72 LM).
2. Time t_1 of the CSI maneuver.
3. Number N of the apsidal crossing.
(If $N = 1$, the CDH maneuver occurs when the active vehicle reaches its first apsidal point following the CSI maneuver, etc.)
4. Desired line of sight (LOS) angle E at the time of the TPI maneuver.
(See Figure 4.2-2 and Figure 4.2-3.)
5. Time t_3 of the TPI maneuver.
6. Selection of flag CDHASW to ensure that CDH will occur at a time, measured from the CSI time, equal to the post-CSI maneuver orbital period multiplied by $N/2$. If this option is desired, CDHASW is set equal to 1.

The active vehicle state vector $\underline{r}_A, \underline{v}_A$ and passive vehicle state vector $\underline{r}_P, \underline{v}_P$ are available in the guidance computer.

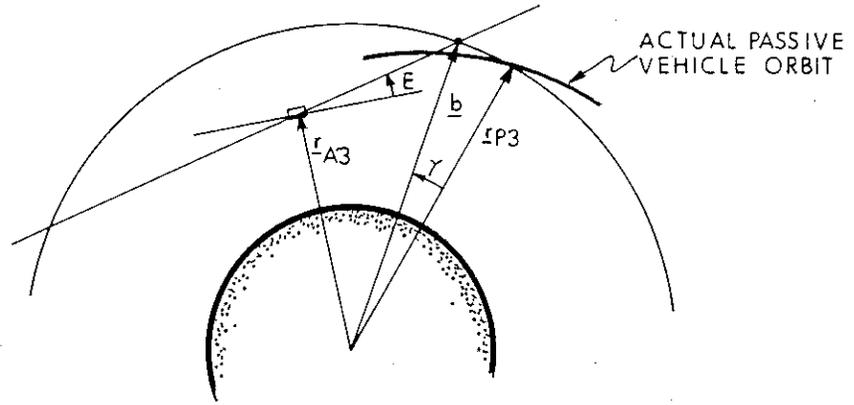
The following constraints must be satisfied (see Figure 4.2-1 and Figure 4.2-2).

1. The CSI ΔV is applied in the horizontal plane of the active vehicle at the CSI time.
2. The CDH maneuver occurs when the active vehicle is at a specified apsidal crossing.
3. The semimajor axis and radial component of velocity of the active vehicle are such that the active and passive vehicles are in coelliptic orbits following the CDH maneuver.
4. The line of sight between the active vehicle and the passive vehicle at the TPI time forms the angle E with the horizontal plane of the active vehicle.

5.4-4

Revised COLOSSUS

Added GSOP #R-577 PCR # 254.1, 681 Rev. 5 Date 10-4-68



TPI geometry, active vehicle below

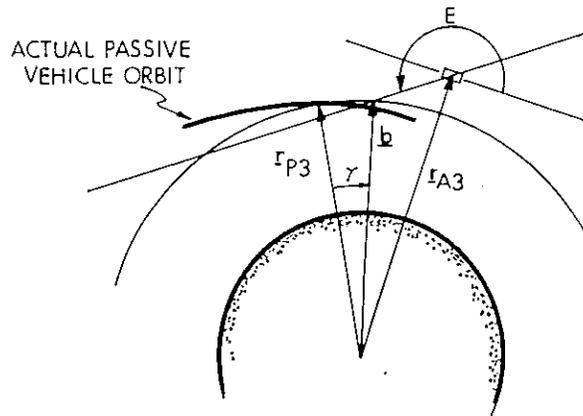
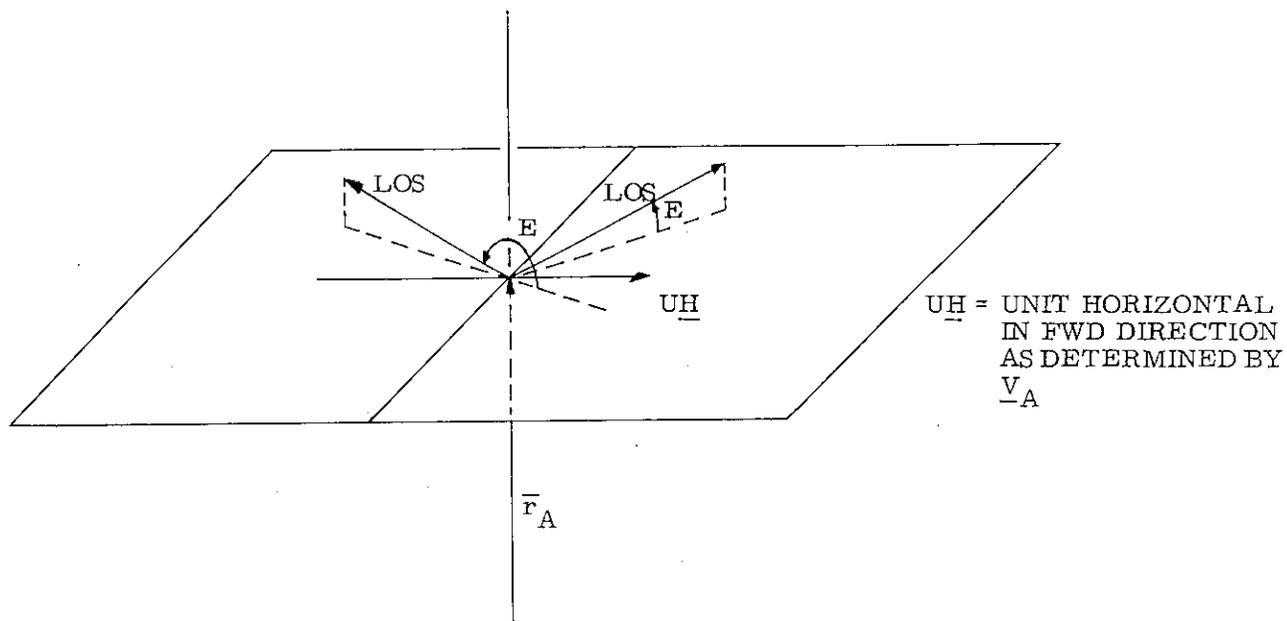


Figure 4.2-2 TPI geometry, active vehicle above



1) IF THE LOS PROJECTION ON \underline{UH} IS POSITIVE:

- a) WHEN THE LOS IS ABOVE THE HORIZONTAL PLANE
 $0 < E < \pi/2$
- b) WHEN THE LOS IS BELOW THE HORIZONTAL PLANE
 $3\pi/2 < E < 2\pi$

2) IF THE LOS PROJECTION ON \underline{UH} IS NEGATIVE

- a) WHEN THE LOS IS ABOVE THE HORIZONTAL PLANE
 $\pi/2 < E < \pi$
- b) WHEN THE LOS IS BELOW THE HORIZONTAL PLANE
 $\pi < E < 3\pi/2$

Figure 4.2-3 Definition of Elevation Angle, E

5. The time intervals between the CSI-CDH and CDH-TPI maneuvers are 10 minutes or greater.
6. After both the CSI and CDH maneuvers, the perigee altitude of the active vehicle orbit is greater than 35,000 ft for lunar orbits and 85 nm for earth orbits.

The program solution, as illustrated in the logic flow diagram in Figure 4.2-4, is based on conic trajectories and contains an iteration loop to select the CSI maneuver magnitude v_1 .

After updating of the active and passive vehicle state vectors to the CSI time, the active vehicle state vector is rotated into the passive vehicle orbital plane, resulting in coplanar orbits. The passive vehicle is then updated through the time $t_3 - t_1$ to the TPI point obtaining \underline{r}_{P3} , \underline{v}_{P3} . The initial guess of v_1 , when added to the velocity of the active vehicle, results in the active vehicle attaining a radius equal to r_{P3} 180 degrees from the CSI point.

If CDHSW equals one, the eccentricity of the active vehicle orbit is less than .0001, or the vertical velocity at the CSI point is less than 7 ft/sec, the time t_2 of the CDH maneuver is set equal to the CSI time plus N times half the period t_P of the active vehicle. Otherwise, the angle ψ to the nearest perigee is computed and used in the Time-Theta Subroutine to find the corresponding time of flight Δt . This is then used to calculate t_2 based on the value of N. After verifying that t_3 is greater than t_2 , both vehicles are updated to the CDH point.

The angle ϕ between the active and passive vehicles at the CDH point is calculated and used in the Time-Theta Routine to update the passive vehicle to a point radially above or below the active vehicle, obtaining \underline{r}_{PC} , \underline{v}_{PC} .

The semimajor axis a_A and the radial component of velocity v_{AV} , along with a radius, uniquely define the orbit of the active vehicle immediately following the CDH maneuver. The equations for a_A (based on the semimajor axis a_P of the passive vehicle) and v_{AV} , contained in Figure 4.2-4, therefore, specify the "coelliptic" CDH maneuver Δv_2 .

The active vehicle state vector is next updated to the TPI time. The unit vector which passes through the active vehicle position and is coincident with the desired TPI line of sight is given by \underline{u}_L . The position vector of the two points of intersection between the line of sight and a circle (with origin at the center of the attracting body) as shown in Figure 4.2-2 is defined by:

$$\underline{b} = \underline{r}_{A3} + k\underline{u}_L \quad (4.2.1)$$

5.4-7

Revised COLOSSUS

Added GSOP # R-577 PCR # 681 Rev. 5 Date 1-7-69

where k normally assumes two values (see Figure 4.2-4). Equating the magnitude of \underline{b} to r_{P3} results in a quadratic expression for k :

$$k^2 + 2kr_{A3} \cdot \frac{u_L}{r_{A3}} + r_{A3}^2 - r_{P3}^2 = 0 \quad (4.2.2)$$

If there are no real solutions to the above equation ($c_2 < 0$ in Figure 4.2-4), the line of sight does not intersect the circle. If there is a real solution to the above equation, the k of minimum absolute value is chosen as the desirable TPI solution. The error γ is defined as the central angle between the position vector of the passive vehicle at the TPI point and the position vector \underline{b} representing where the passive vehicle should be based on the active vehicle location. This error is driven to zero using a Newton-Raphson iteration loop with v_1 as the independent variable. The two initial values of v_1 are 10 ft/sec apart with the succeeding step size Δv_1 restricted to 200 ft/sec.

A second computational attempt is made if the first iteration attempt fails. v_1 is incremented in steps of 50 ft/sec in the direction opposite to that initially taken in the first attempt until the error γ has undergone a sign change. The second attempt then starts with the latest value of v_1 and an adjacent point in the direction of the 50 ft/sec steps.

Included in the Pre-CSI program are the following seven program checks and the corresponding Alarm Codes AC (see Figure 4.2-4).

1. If on the first iteration there is no solution to the TPI geometry, the iteration cannot be established. The alarm code (AC) equals A in this case.
2. The iteration counter n exceeds 15. (AC = C)
3. Two succeeding iterations resulting in v_1 greater than 1000 ft/sec. (AC = B)
4. The altitude of perigee h_{P1} following the CSI maneuver is less than a minimum acceptable amount. (AC = D)
5. The altitude of perigee h_{P2} following the CDH maneuver is less than a minimum acceptable amount. (AC = E)
6. The time ΔT_2 between the CDH and CSI maneuver is less than 10 minutes. (AC = F)
7. The time ΔT_3 between the TPI and CDH maneuvers is less than 10 minutes. (AC = G)

5.4-8

Revised COLOSSUS

Added GSOP # R-577 PCR # 681 Rev. 5 Date 1-7-69

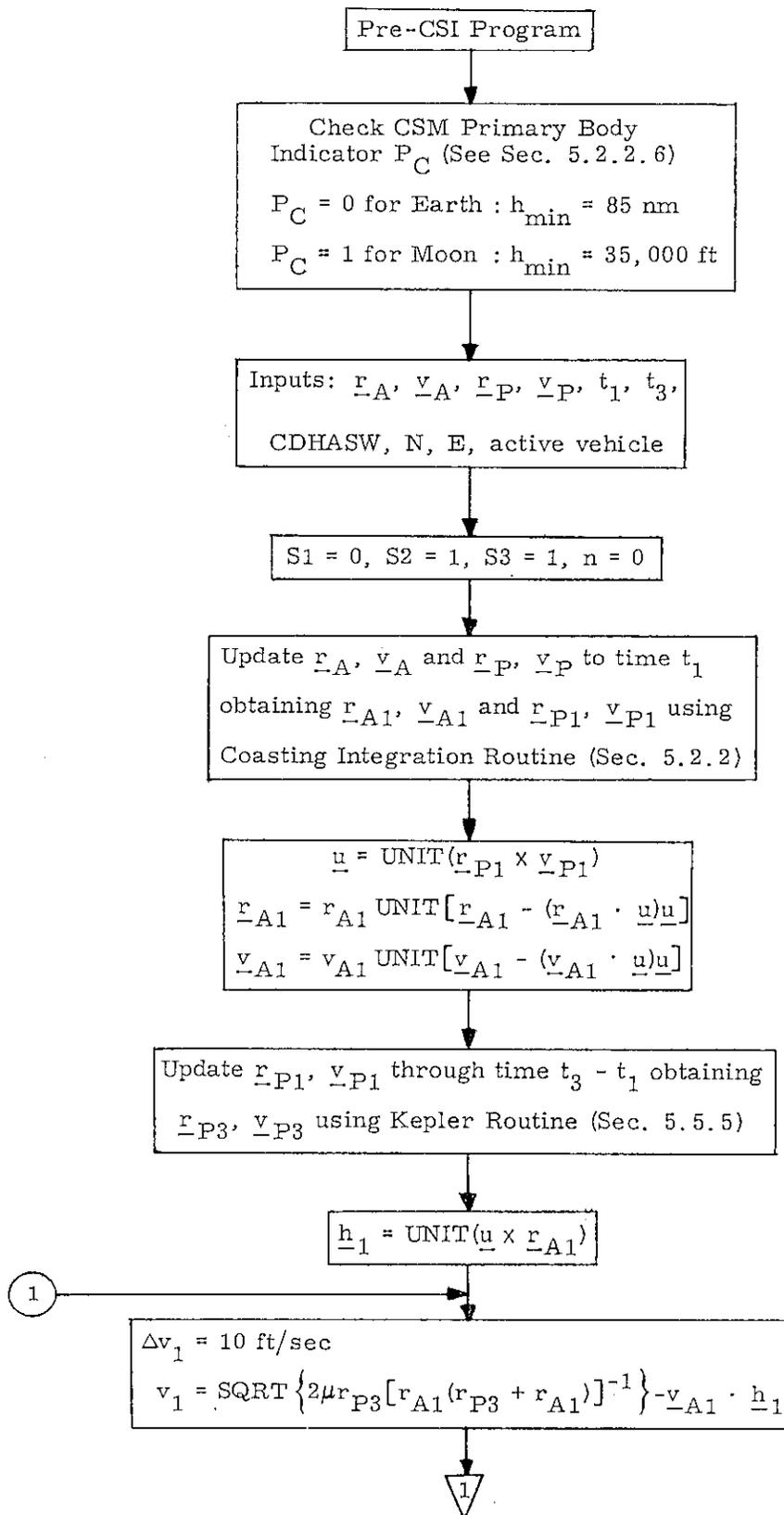


Figure 4.2-4 Pre-CSI-CDH Maneuver Program
(Page 1 of 7)

5.4-9

Revised COLOSSUS

Added GSOP #R-577 PCR # 681

Rev. 5

Date 1-7-69

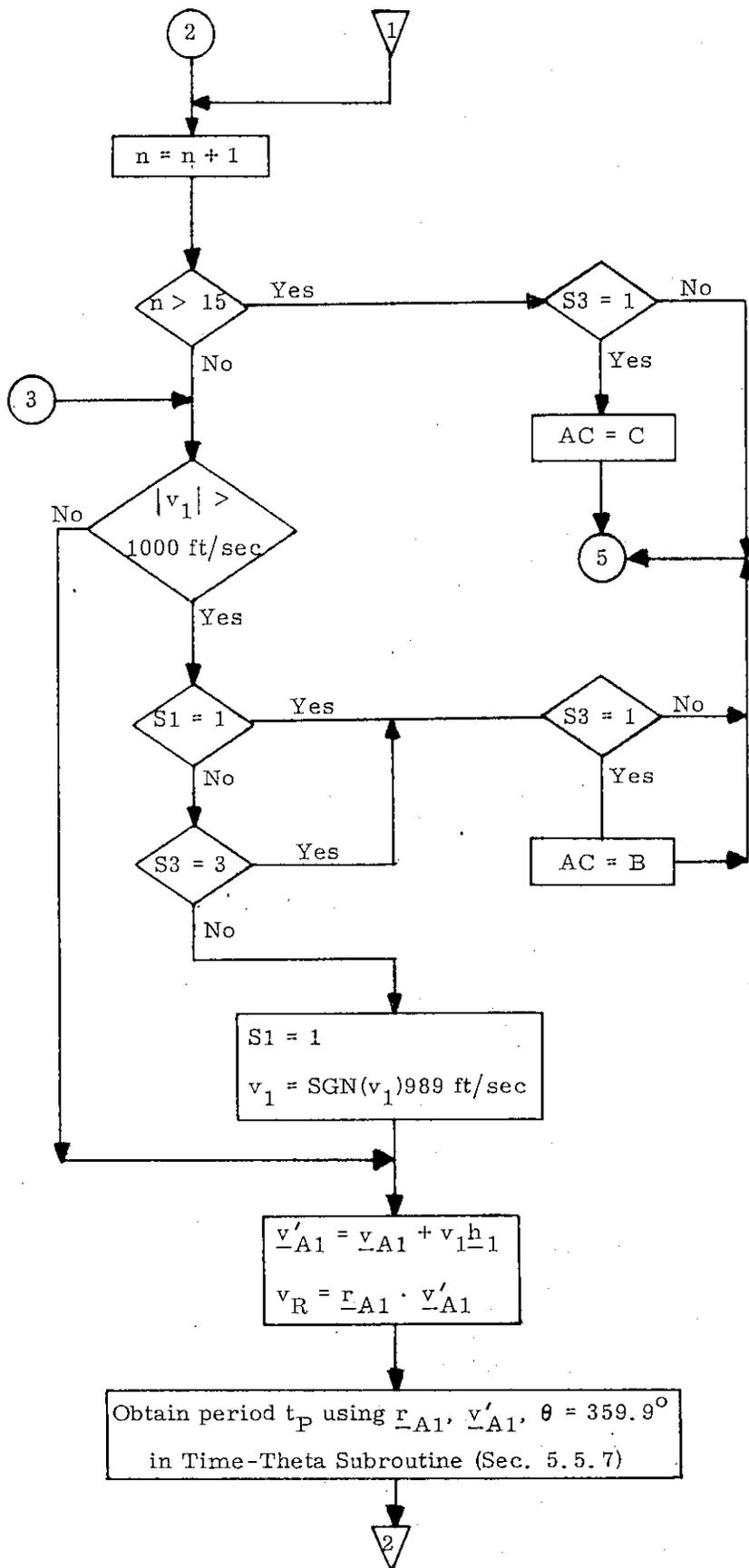


Figure 4.2-4 Pre-CSI-CDH Maneuver Program (Page 2 of 7)

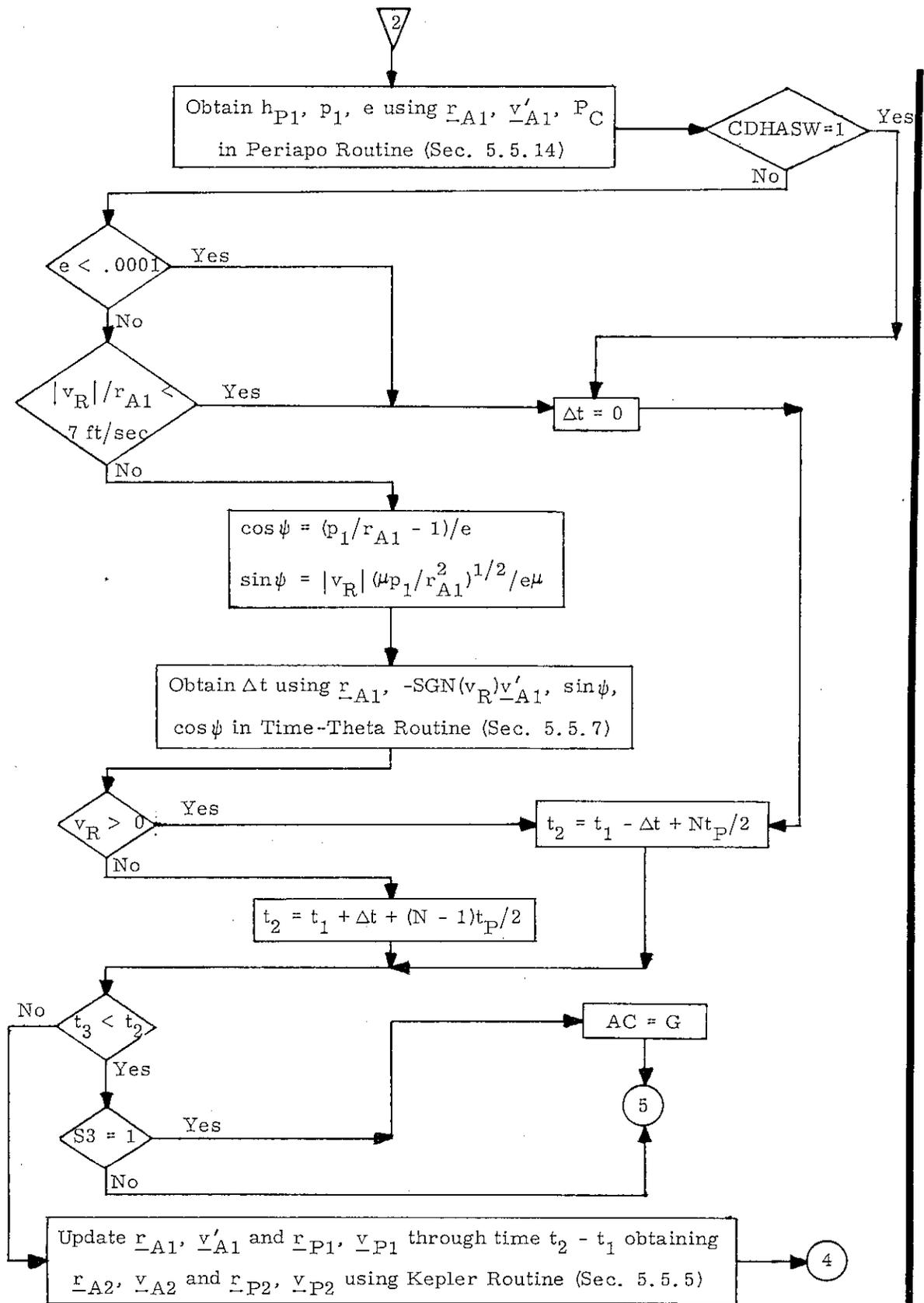


Figure 4.2-4 Pre-CSI-CDH Maneuver Program
(Page 3 of 7)

5.4-11

Revised COLOSSUS

Added GSOP #R-577 PCR # 254.1

Rev. 5

Date 10-4-68

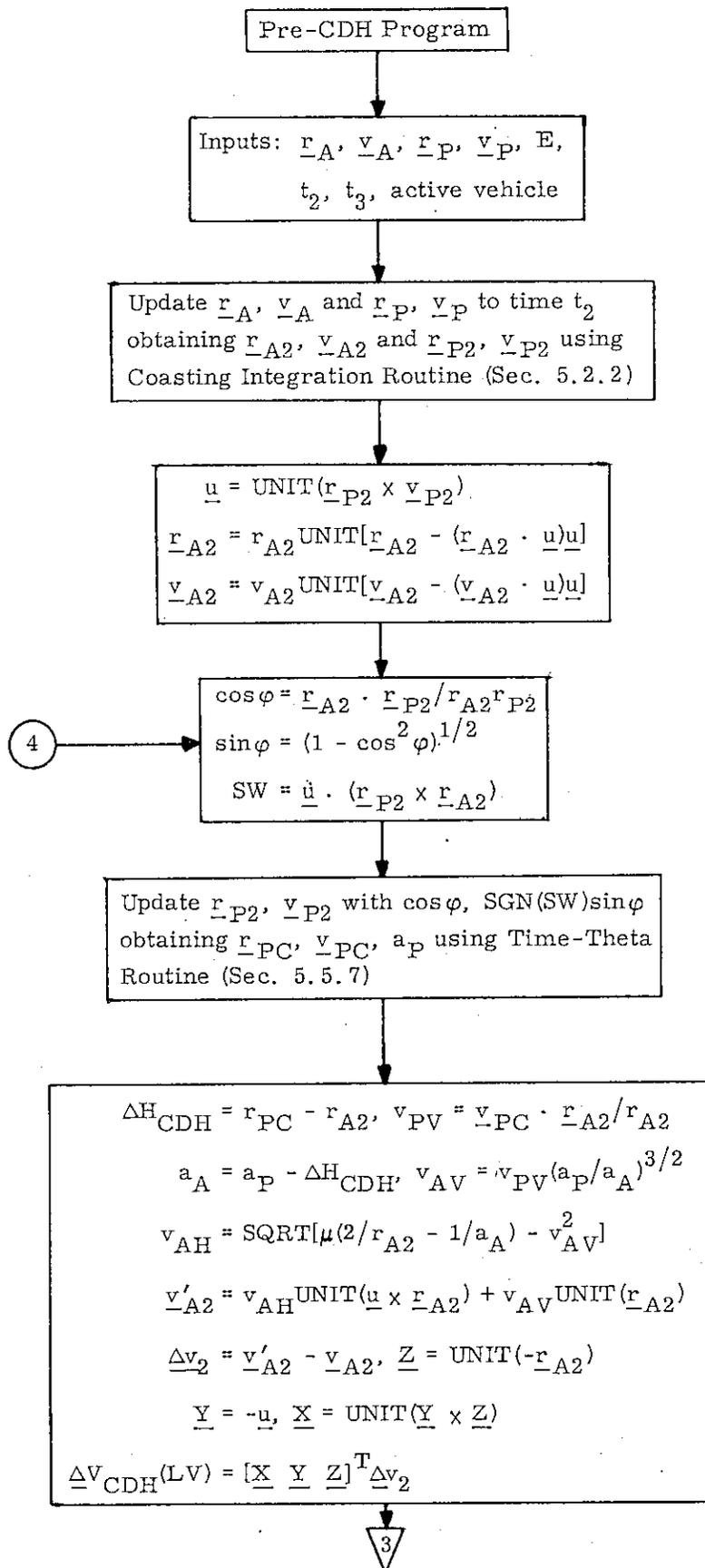


Figure 4.2-4 Pre-CSI-CDH Maneuver Program (Page 4 of 7)

Revised COLOSSUS

5.4-12

Added

GSOP #R-577

PCR # 681

Rev. 5

Date 1-7-69

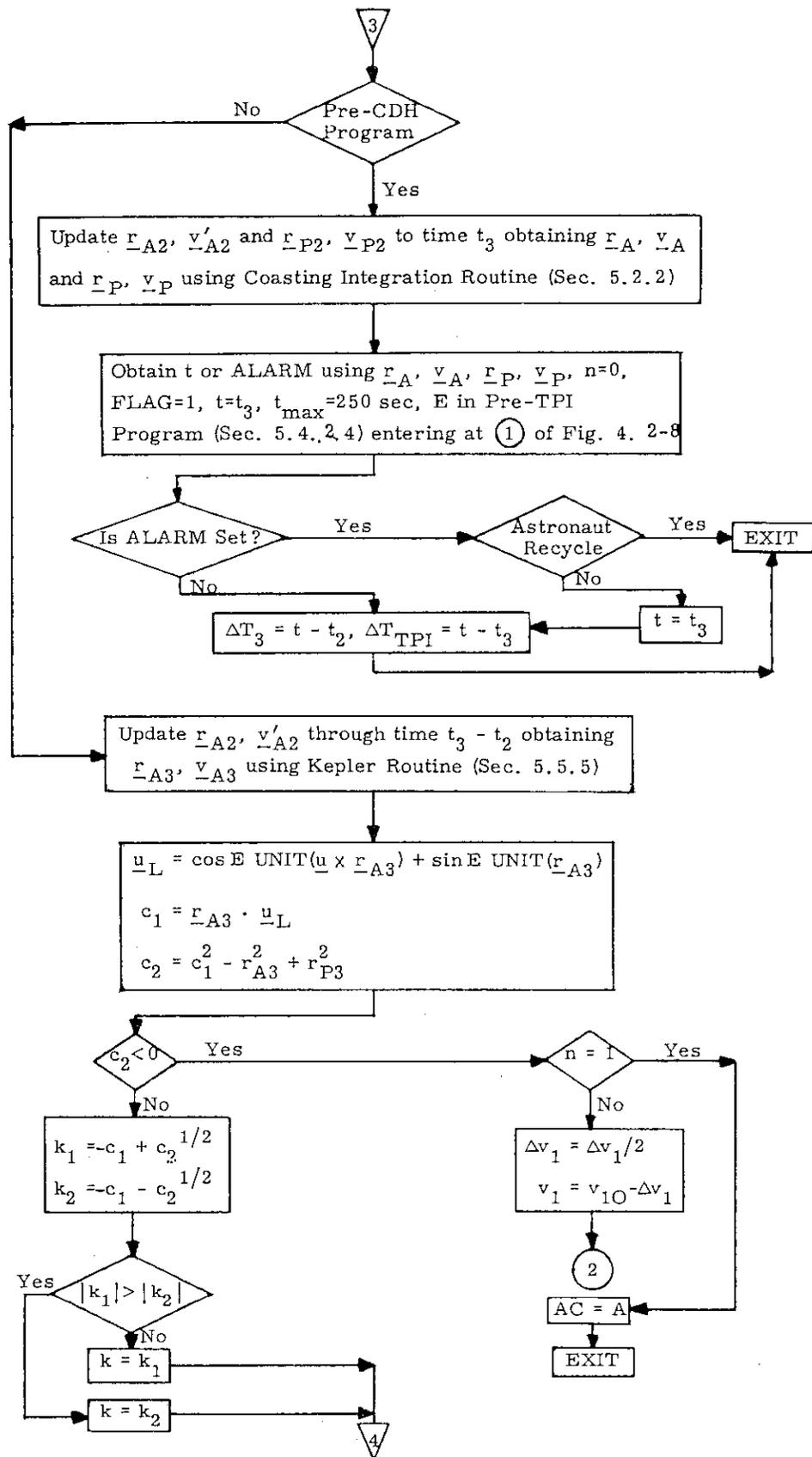


Figure 4.2-4 Pre-CSI-CDH Maneuver Program
(Page 5 of 7)

5.4-13

Revised COLOSSUS

Added GSOP #R-577 PCR # 681 Rev. 5 Date 1-7-69

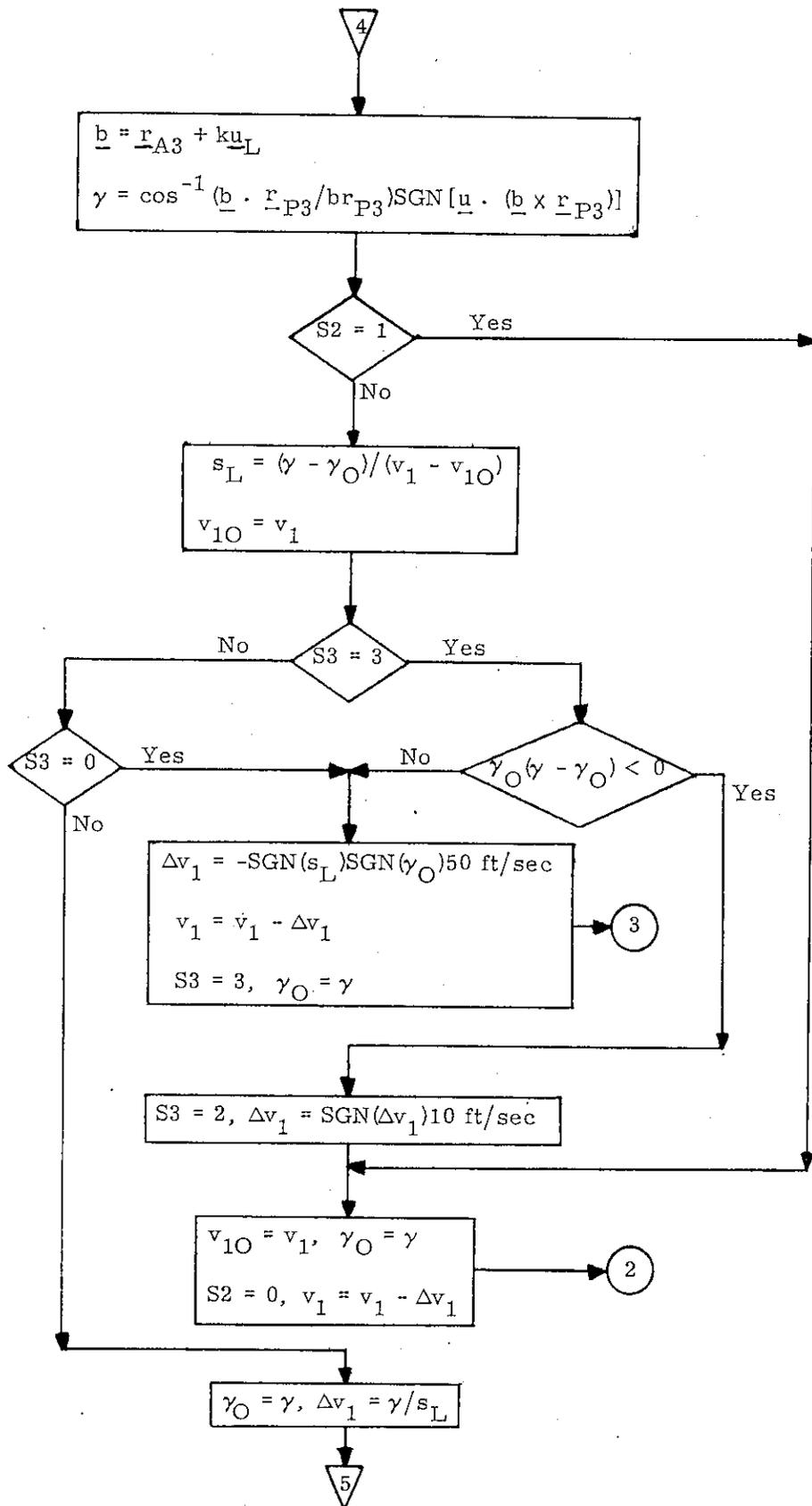


Figure 4.2-4 Pre-CSI-CDH Maneuver Program
(Page 6 of 7)

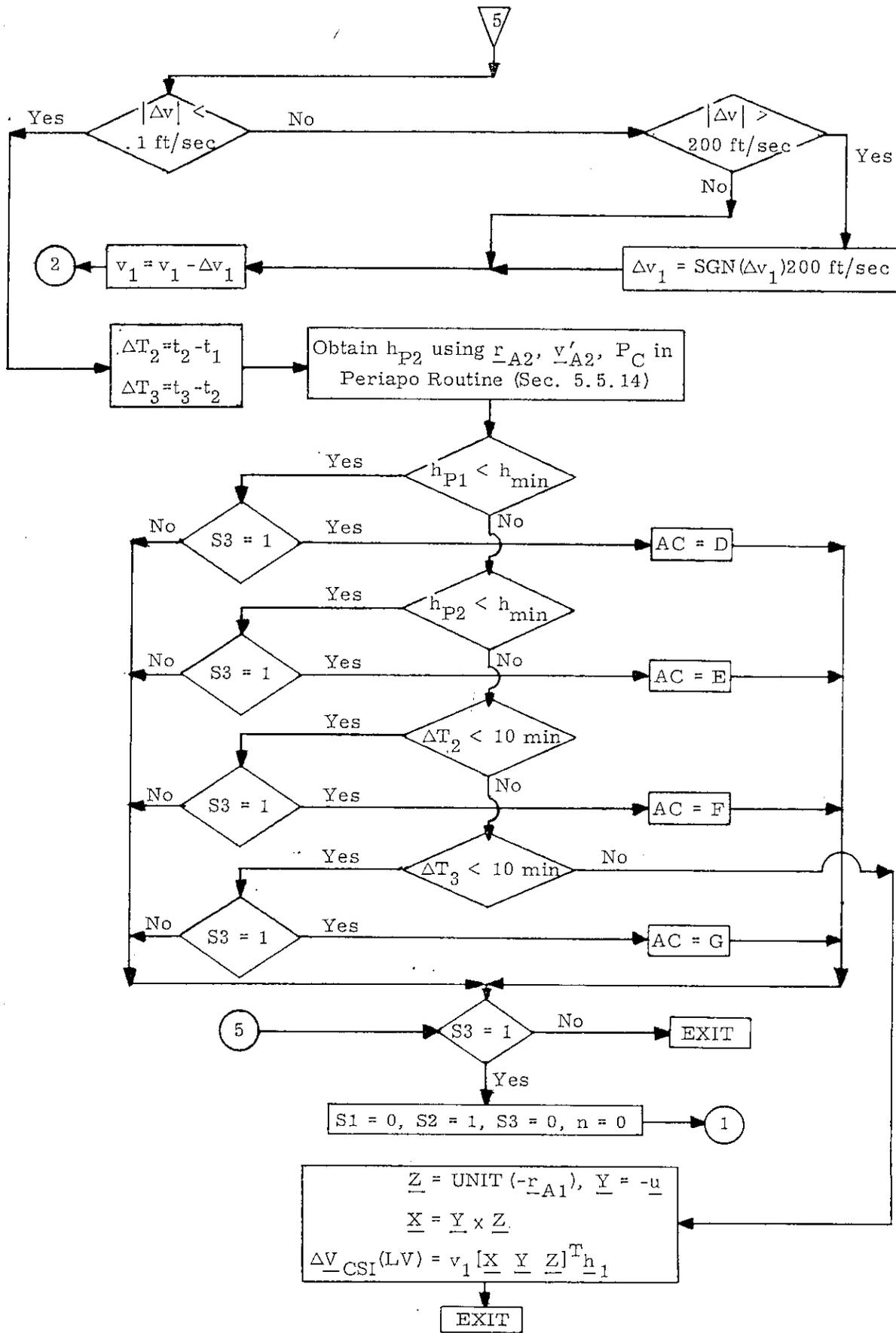


Figure 4.2-4 Pre-CSI-CDH Maneuver Program (Page 7 of 7)

5.4-15

Revised COLOSSUS

Added GSOP #R-577 PCR # 681 Rev. 5 Date 1-7-69

For the first check above, a check failure results in an immediate program exit. The last four checks are made only if the iteration has succeeded. The alarm codes are set (excluding the first check) only if the second iteration attempt has been made.

The displays for the Pre-CSI maneuver program are:

1. Differential altitude ΔH_{CDH} at the CDH point. ΔH_{CDH} is positive when the active vehicle's altitude is less than the passive vehicle's altitude when measured at the CDH point.
2. Time ΔT_2 between the CDH and CSI maneuvers displayed in minutes and seconds with hours deleted.
3. Time ΔT_3 between the TPI and CDH maneuvers displayed in minutes and seconds with hours deleted.
4. $\Delta V_{CSI}(LV)$ in local vertical coordinates at CSI time.
5. $\Delta V_{CDH}(LV)$ in local vertical coordinates at CDH time.
6. The alarm codes.

In the Pre-CSI program P-32, the astronaut can change the CSI computed maneuver by overwriting the displayed maneuver velocity vector in local vertical coordinates, $\Delta V_{CSI}(LV)$. In such a case the astronaut input then becomes the defined velocity-to-be-gained in the External ΔV Maneuver Guidance (Section 5.3.3.3.1) used to control the CSI maneuver. The overwriting procedure is normally used to correct or minimize out-of-plane conditions which are not directly controlled by the CSI or CDH maneuvers computations such that minimum out of plane conditions exist at the TPI maneuver.

In addition, M, the number of navigation measurements processed since the last maneuver, MGA, the middle gimbal angle at the start of the maneuver assuming plus X acceleration, and TFI, the time between the CSI maneuver and the present time, are displayed.

The middle gimbal angle is computed with the following equation:

$$MGA = \sin^{-1} \left[\underline{Y}_{REF} \cdot \text{UNIT}(\Delta V) \right]$$

If $-90^\circ \leq MGA < 0^\circ$, MGA is redefined such that $270^\circ \leq MGA < 360^\circ$. ΔV is the maneuver ΔV in reference coordinates and \underline{Y}_{REF} is the second row of the REFSMMAT matrix defined in Section 5.6.3.3.

5.4-16

Revised COLOSSUS

Added GSOP #R-577 PCR # 681 Rev. 5 Date 1-7-69

5.4.2.1.2 Pre-CDH Maneuver. -- This program, corresponding to Program P-33 (CSM active) or P-73 (LM active) of Section 4, computes the parameters associated with the CDH maneuver.

The astronaut inputs to this program are:

1. Choice of active vehicle (P-33 CSM, P-73 LM).
2. Time t_2 of CDH maneuver.

The active and passive vehicle's state vectors \underline{r}_A , \underline{v}_A , \underline{r}_P , \underline{v}_P are available in the guidance computer. The TPI elevation angle E and the TPI maneuver time t_3 are also available from the previous Pre-CSI program computations.

Figure 4.2-4, page 4 of 7, illustrates the program logic. The state vectors of both the active and passive vehicles are advanced to the CDH time using the Coasting Integration Routine. After rotating the active vehicle state vector into the plane of the passive vehicle, the CDH maneuver is calculated as in the Pre-CSI program. Following a precision update of the state vectors to time t_3 , the Pre-TPI program is used to calculate the time t at which the specified elevation angle is attained. If the iteration was not successful, an alarm code is displayed. At this point, the astronaut can elect to recycle the program or to proceed with the calculation of the displays, which are:

1. $\Delta \underline{V}_{CDH}(LV)$.
2. Differential altitude ΔH_{CDH} at the CDH point.
3. Time ΔT_3 between the calculated TPI time and the CDH time displayed in minutes and seconds with hours deleted.
4. Time ΔT_{TPI} between the calculated TPI time and the specified TPI time. (This number is positive if the new TPI time is later than that previously used.)

If the iteration did not converge, the last two displays reflect an unchanged TPI time. In addition, M, MGA, and TFI are displayed as in the Pre-CSI program displays.

The astronaut can modify the CDH maneuver to minimize out-of-plane conditions by overwriting the displayed maneuver velocity vector, $\Delta \underline{V}_{CDH}(LV)$, as in the Pre-CSI program. The out-of-plane information used in this process is determined by the astronaut with the aid of the Out of Plane Rendezvous Display Routine, R-36 of Section 5.6.7.4.

5.4.2.2 Pre-TPI Maneuver

This program corresponds to Program P-34 (CSM active) or P-74 (LM active) of Section 4. Its objective is to establish the terminal phase initiation (TPI) maneuver. The position of the TPI maneuver is determined by specifying either the TPI time or the elevation angle which specified the relative geometry of the vehicles at the TPI point. The astronaut inputs are:

1. Choice of active vehicle (P-34 CSM, P-74 LM)
2. Time t of the TPI maneuver
3. Elevation angle E (set equal to zero if t is specified), defined in Fig. 4.2-2.
4. Central angle ωt of the passive vehicle between the TPF and TPI points.

The active \underline{r}_{AI} , \underline{v}_{AI} and passive vehicle \underline{r}_{PI} , \underline{v}_{PI} state vectors are available in the guidance computer. The program starts with a precision update of these vectors to the TPI time.

If the elevation angle is not specified, it is computed after the precision updating of both vehicles to the TPI maneuver time. If E is specified, an iteration procedure is initiated to find the TPI time at which E is attained. This procedure uses the input TPI time as the initial guess and is based on conic trajectories (except for the original precision update). The time correction δt is based on (1) the angular distance between \underline{r}_P and the desired position of the passive vehicle, obtained by assuming the passive vehicle is in a circular orbit, and (2) assuming the vehicles are moving at a constant angular rate. The flow diagram for this procedure is shown in Fig. 4.2-5.

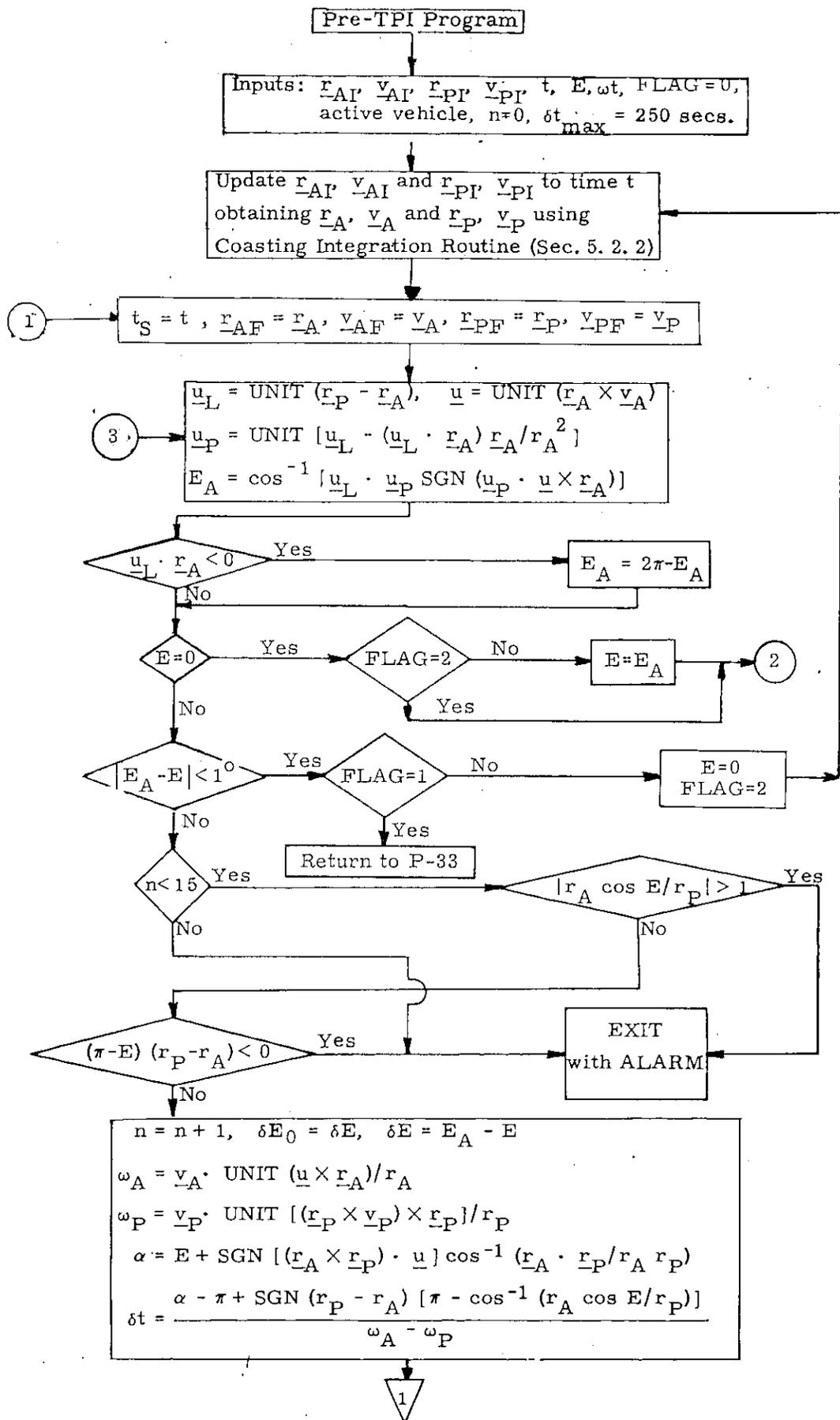


Figure 4.2-5 Pre-TPI Maneuver Program (page 1 of 3)

5.4-19

Revised COLOSSUS

Added GSOP #R-577 PCR # 681

Rev. 5

Date 1-7-63

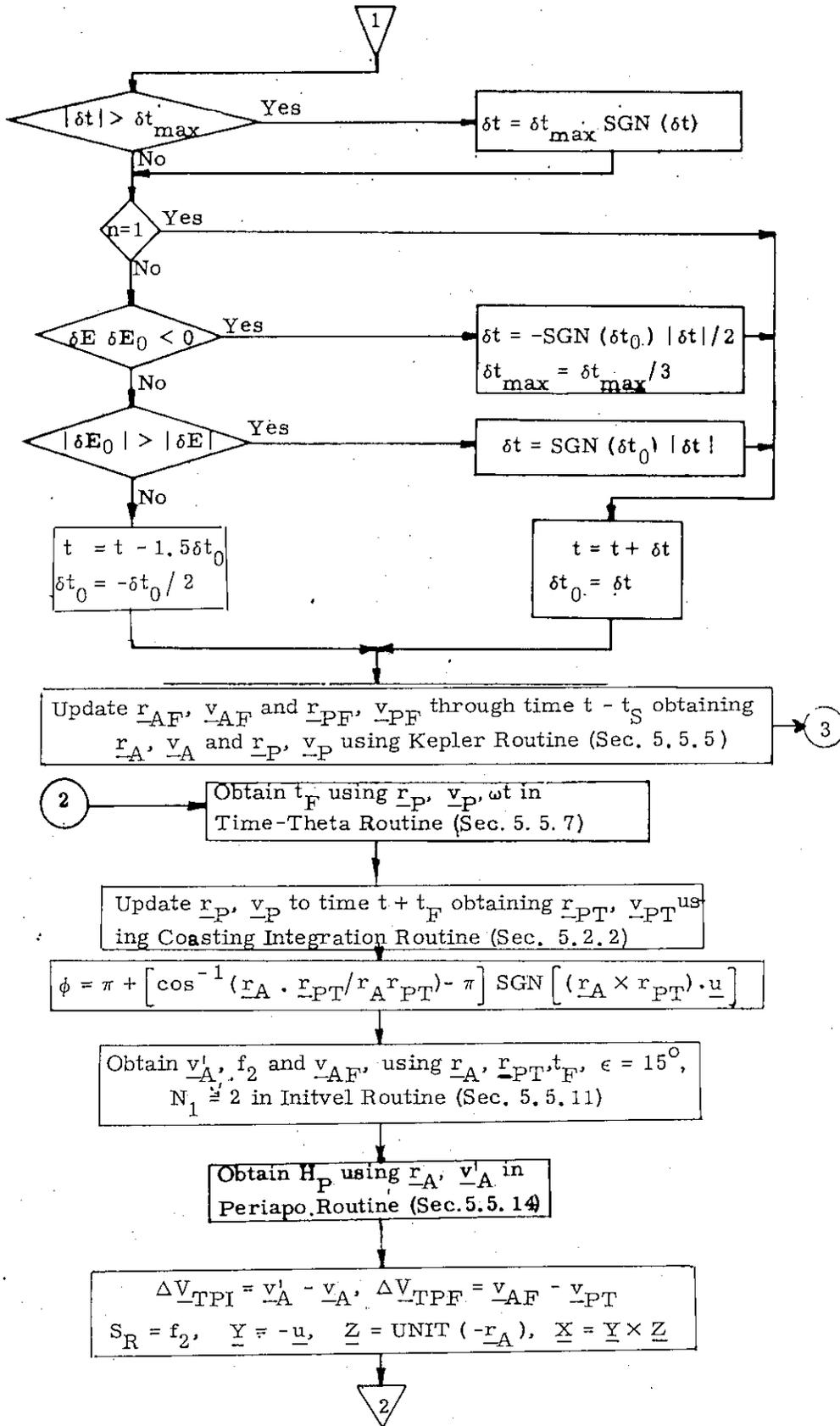


Figure 4.2-5 Pre-TPI Maneuver Program (page 2 of 3)

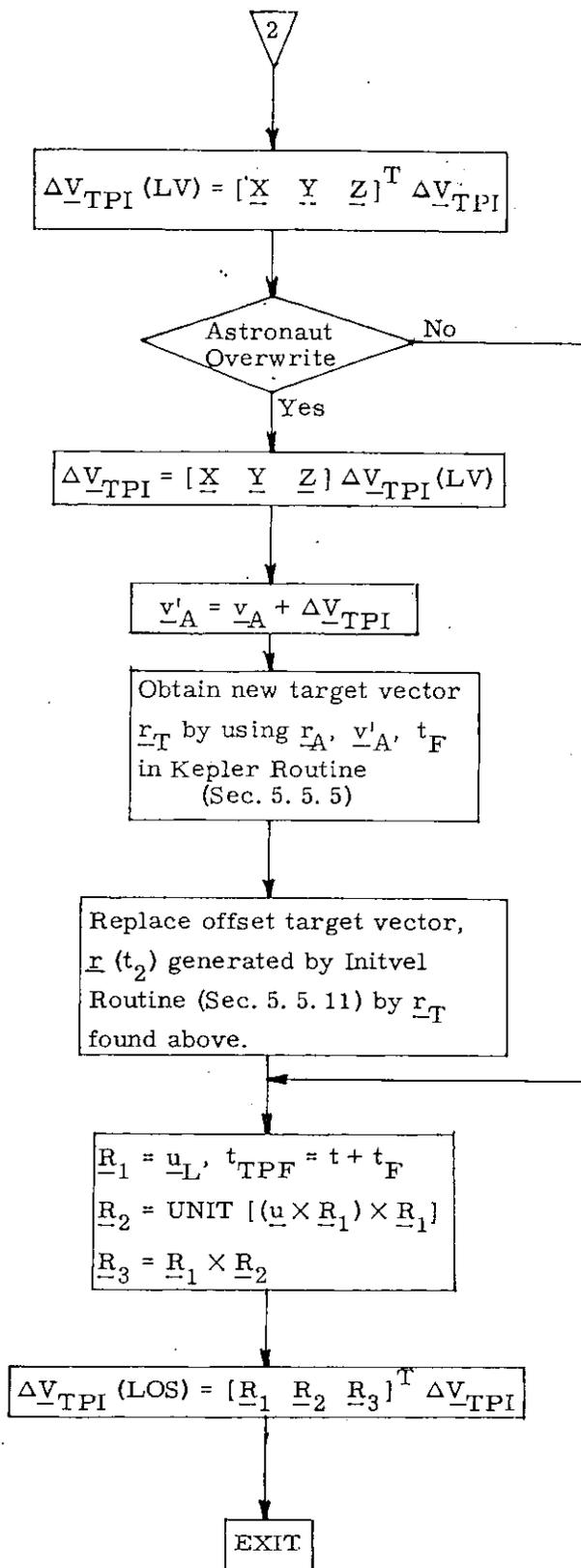


Figure 4.2-5 Pre-TPI Maneuver Program (page 3 of 3)

The iteration is successful when the computed elevation angle E_A is sufficiently close to E .

To help insure convergence, the following steps are taken:

- a. The step size δt is restricted to 250 secs.
- b. If the solution has been passed ($\delta E \delta E_0 < 0$), the step size is halved and forced in the opposite direction of the last step.
- c. If the iteration is converging ($|\delta E_0| - |\delta E| > 0$), the sign of δt is maintained.
- d. If the iteration is proceeding in the wrong direction, the step direction is reversed.

The iteration is terminated for any of the following reasons, and a single alarm as described in Section 4 is set:

1. The iteration counter has exceeded its maximum value of 15.
2. The line of sight emanating from the active vehicle does not intersect the circular orbit with radius equal to that of the passive vehicle.
3. The elevation angle is inconsistent with the relative altitudes of the two vehicles (e. g., if the elevation angle is less than 180° when the active vehicle is above the passive vehicle).

Upon convergence, the state vectors are precision updated to the TPI time.

The TPI-TPF phase of the program starts with the use of the angle ωt in the Time-Theta Subroutine (Section 5.5.7) to determine the corresponding transfer time t_F . After precision updating of the passive vehicle through t_F , the Initvel Routine (Section 5.5.11) is then called to compute a transfer trajectory between the TPI and TPF points with the switch N_1 set equal to two and the cone angle ϵ set equal to 15° . This results in the precision offsetting procedure to be used twice in succession. The rotation projection switch f_2 is obtained from the Initvel Routine for use in the powered flight steering program as S_R .

The central angle ϕ traversed by the active vehicle from the TPI time to intercept is computed as shown in Fig. 4.2-5 for display if requested by the astronaut. This display is used to avoid 180° transfer angle problems.

The displays for the Pre-TPI program are:

1. ΔV_{TPI}
2. ΔV_{TPF}
3. $\Delta V_{TPI}(\text{LOS})$: Line-of-sight coordinates
4. $\Delta V_{TPI}(\text{LV})$: Local vertical coordinates
5. Perigee altitude (H_P) following the TPI maneuver

The display of $\Delta V_{TPI}(\text{LV})$ may be overwritten by the astronaut. If so, a new target vector is generated as shown in Fig. 4.2-5. This new aimpoint is used for powered flight steering.

In addition, M, MGA, and TFI are displayed as in the Pre-CSI program displays.

5.4.2.3 Rendezvous Midcourse Maneuver

This program, corresponding to Program P-35 (CSM active) or P-75 (LM active) of Section 4, computes a midcourse correction maneuver. This maneuver insures that the active vehicle will intercept the passive vehicle at the time established in the previous Pre-TPI program. The astronaut may call this program any time after the TPI maneuver, but in general no later than 10 minutes before the intercept time. The flow diagram for this program is shown in Fig. 4.2-6.

There is one astronaut input: choice of the active vehicle (P-35 CSM, P-75 LM). The active \underline{r}_A , \underline{v}_A and passive \underline{r}_P , \underline{v}_P vehicle state vectors, the intercept time t_{TPI} (from the Pre-TPI program) and a time delay are available in the guidance computer. The time delay ($\delta\tau_3$ for P-35 and $\delta\tau_7$ for P-75) is the time required to prepare for the thrust maneuver and is stored in either of two erasable locations.

When the program is initiated the number of navigation measurements since the last maneuver and the time since TPI are displayed. Based on this information and additional displays discussed in Section 5.6.7, the astronaut may elect to proceed with the midcourse maneuver at some time. When he does so, the program updates the state vectors to the present time plus the time delay using the Coasting Integration Routine. The Initvel Routine is then called to obtain the velocity vector on the transfer ellipse. The cone angle ϵ is set equal to 15° and the N_1 switch is set equal to two, resulting in a double use of the offsetting procedure. The rotation projection switch f_2 is obtained from the Initvel Routine for use in the powered flight steering program as S_R .

The central angle ϕ traversed by the active vehicle from the maneuver time to intercept is computed as shown in Fig. 4.2-6 for display if requested by the astronaut. This display is used to avoid 180° transfer angle problems.

After obtaining the maneuver $\Delta\underline{V}$, it is rotated into a local vertical (LV) and a line of sight (LOS) coordinate system obtaining $\Delta\underline{V}(LV)$ and $\Delta\underline{V}(LOS)$, and both are displayed.

The display of $\Delta\underline{V}(LV)$ may be overwritten by the astronaut. If so, a new target vector is generated as shown in Fig. 4.2-6. This new aimpoint is used for powered flight steering.

In addition, M, MGA and TFI are displayed as in the Pre-TPI program displays.

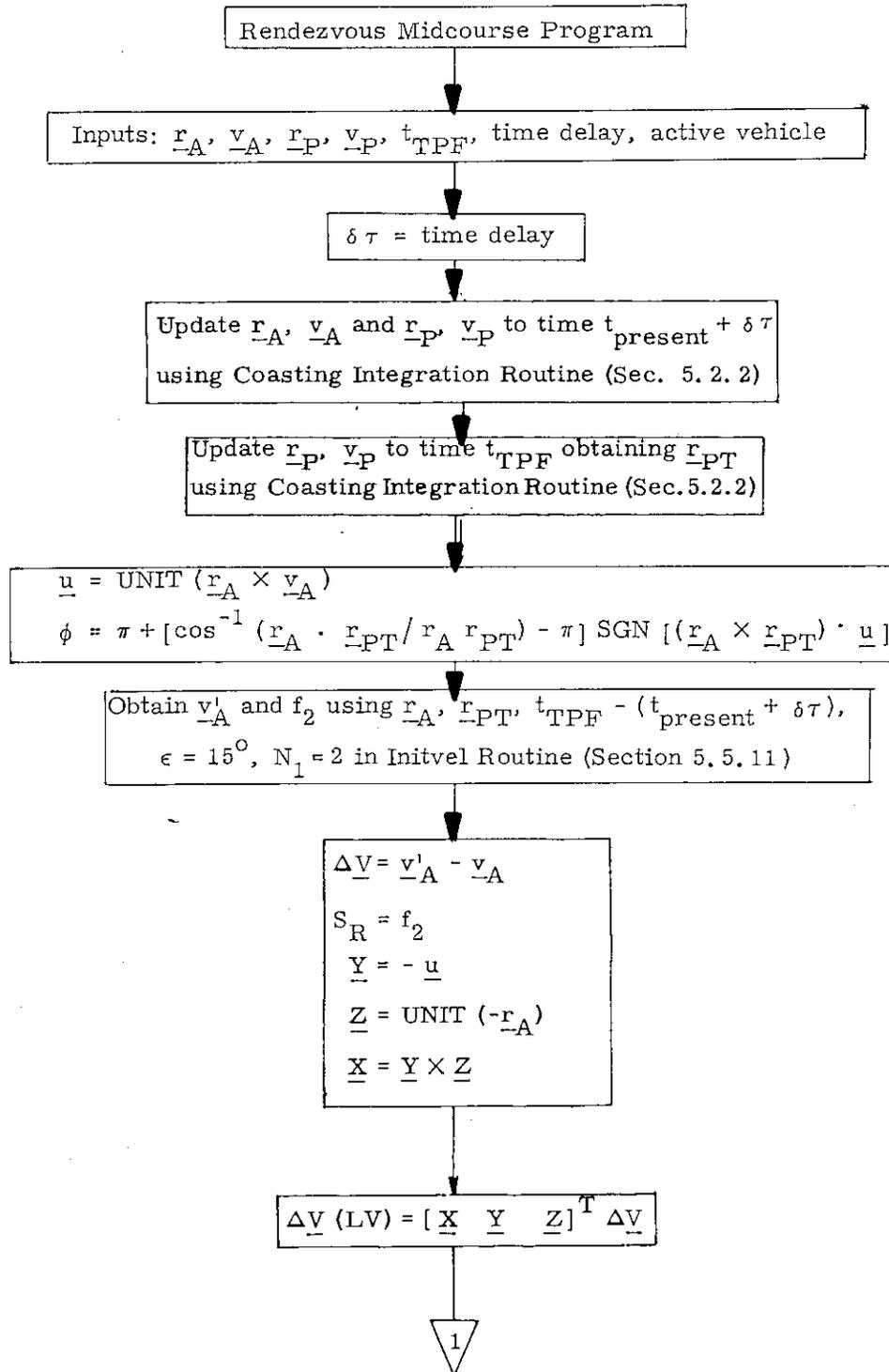


Figure 4.2-6 Rendezvous Midcourse Program
(page 1 of 2)

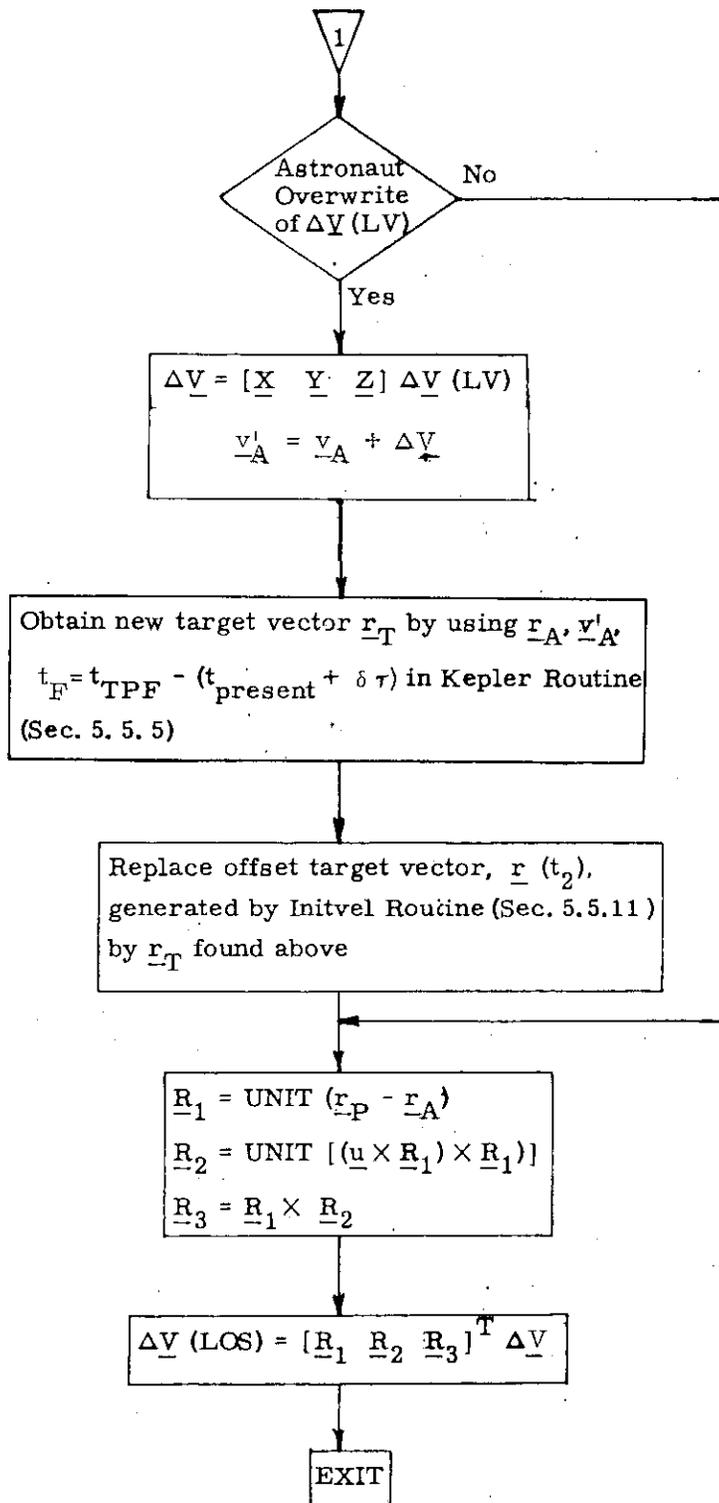


Figure 4.2-6 Rendezvous Midcourse Program
(page 2 of 2)

5.4.2.4 TPI Search Program

The objective of the TPI Search Program is to determine the transfer trajectory having a minimum total velocity change, within the constraint of a safe pericenter, for a specified TPI time. The minimum total velocity determined is the sum of the impulsive velocities for the TPI and TPF maneuvers. This program is used to establish the target parameters for a direct intercept trajectory between the LM and CSM orbits. The manual terminal rendezvous maneuver is normally initiated from such an intercept trajectory. The TPI Search Program is used to provide the required TPI maneuver parameters for either direct transfer rendezvous profiles or concentric flight plan rendezvous TPI maneuvers if ΔV minimization is desired beyond nominal preselected astronaut input target parameters. This program can be used in both earth and lunar orbits and consists of two separate programs in Section 4. When used as program P-17, the active vehicle which establishes the intercept trajectory is the CSM. When used as P-77, the active vehicle is the LM.

The desired time t_{TPI} of the TPI maneuver is required as an input to this program. The basic output parameters displayed to the astronaut are:

- 1) $\pm \phi$ Initial relative phase angle between the two vehicles at t_{TPI} + indicating that the active vehicle is ahead of the passive vehicle with respect to orbital motion.

- 2) $\pm \Delta h$ Altitude difference between the two vehicles at t_{TPF} + indicating that the passive vehicle has a larger radial magnitude than the active vehicle.
- 3) K Preferred central angle search sector. K = +1 for central angles greater than 180° and K = -1 for central angles less than 180° .

The search option input defined in Section 4 has the following correspondence to the variable K:

search option	K
1	-1
2	+1

- 4) ΔV_{TPI} Impulsive velocity required for the TPI maneuver.
- 5) $\pm \Delta V_{\text{TPF}}$ Impulsive velocity required for the terminal rendezvous maneuver, + indicating that the active vehicle is ascending at TPF, - if descending.
- 6) h_P Altitude of the pericenter above a reference radius vector.
- 7) ωt Central transfer angle of the passive vehicle.
- 8) P Pericenter code. P = 1 if pericenter occurs between the TPI and TPF maneuvers and P = 2 if pericenter occurs after the TPF maneuver.

In general, non-coplanar transfer problems normally involve two minimum total velocity trajectories, one having a central angle greater than 180° and the other less than 180° . The program selects a sector (determined by the relative altitudes of the two vehicles at the TPI time) in which the most acceptable solution will probably be found. Upon displaying this sector to the astronaut, he may proceed with this choice or change the sector by changing the value of K.

Due to the excessive ΔV requirements for non-coplanar two-impulse transfers near central angles of 180° , the search sectors are restricted from the 180° region by an angle which depends upon the magnitude of the angle between the active vehicle position vector at TPI and the passive vehicle orbital plane.

The general TPI Search Program target vectors and relative angles are illustrated in Fig. 4.2-7 with the program logic outlined in Fig. 4.2-8. The subscripts A and P refer to the active and passive vehicles, respectively. The minimum pericenter altitude h_L is defined as

Lunar Orbit: $h_L = 35,000$ ft. about the landing site
radius.

Earth Orbit: $h_L = 85$ nm above the launch pad radius.

The following are the major computation sequences:

1. Upon activation of the program, the active and passive vehicle state vectors are advanced to the specified TPI time by means of the Kepler Routine. At this point the relative phase angle ϕ , altitude difference Δh , and search sector flag K are computed and displayed to the astronaut and a response is requested. The astronaut may proceed, command the initial search to the other sector by changing K or change the TPI time.

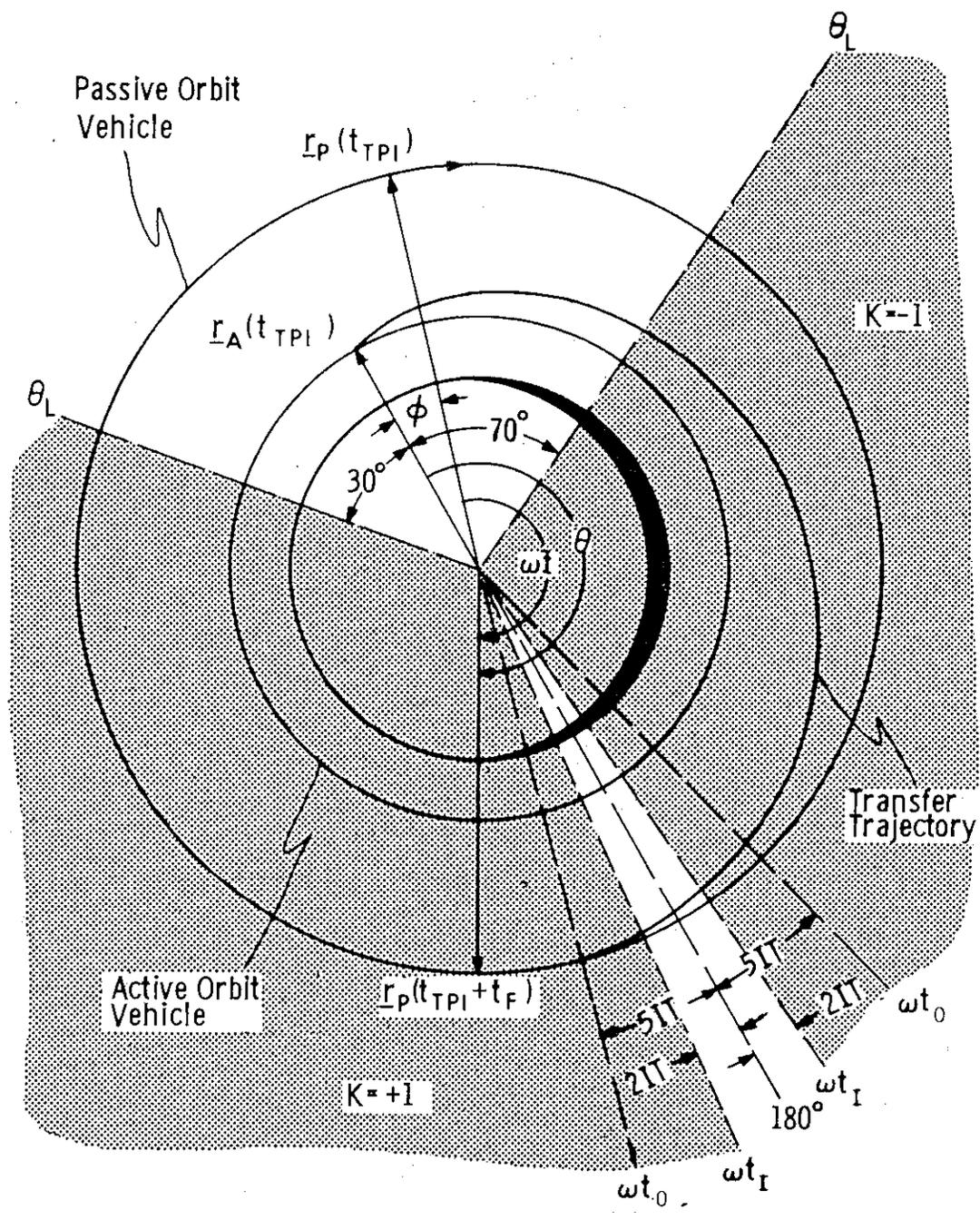


Figure 4.2-7 Central Angle Sectors for the TPI Search Program

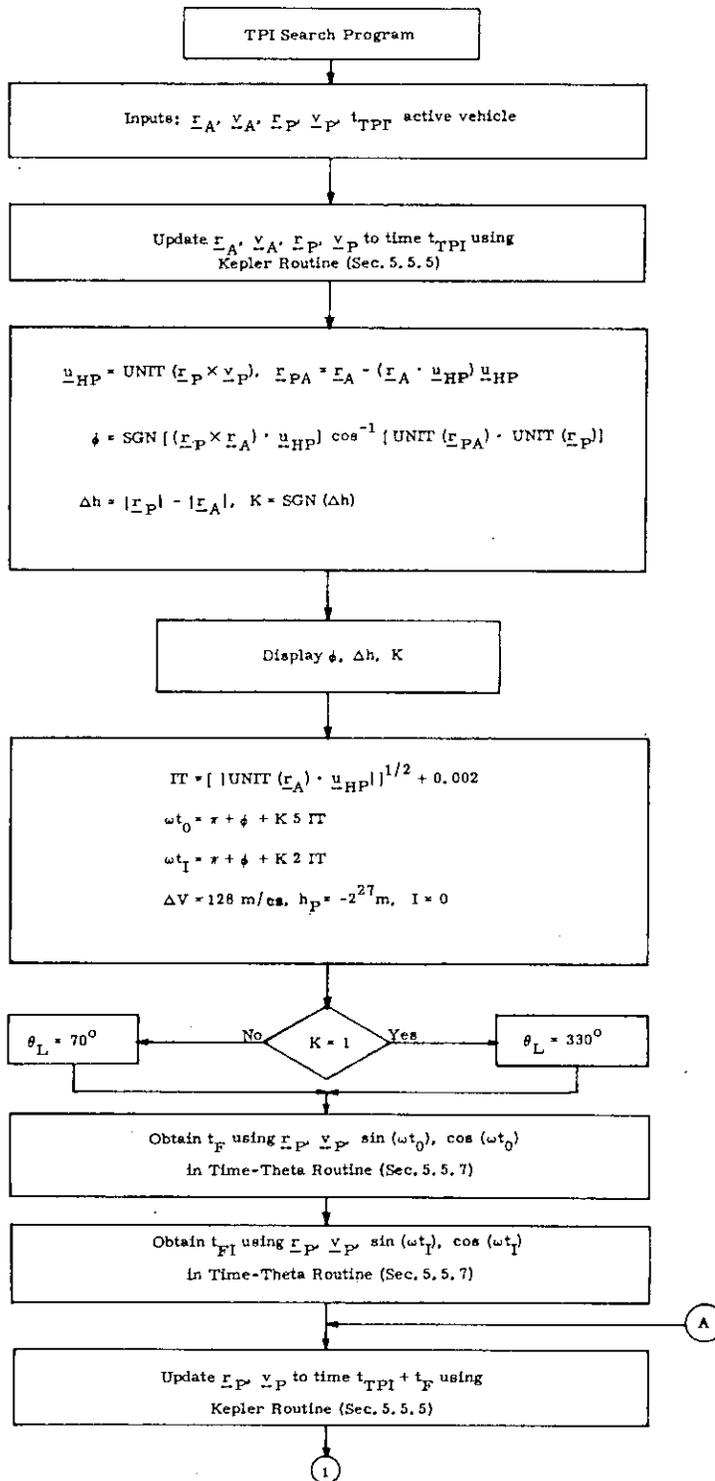


Figure 4.2-8 TPI Search Program
Logic Diagram (Page 1 of 2)

2. An initial guess ωt_0 is next made of the transfer angle of the passive vehicle corresponding to a minimum ΔV for the active vehicle which is used to start the search iteration. An inner transfer angle limit ωt_I for the passive vehicle is set to avoid excessive ΔV conditions, and the outer search limits θ_L of the active vehicle are set as indicated in Fig. 4.2-8. Initial values are next chosen for ΔV , h_P and I . The Time Theta Routine is then used to determine the passive vehicle transfer times t_F and t_{FI} to traverse the angles ωt_0 and ωt_I respectively.

3. The minimum ΔV trajectory search begins by advancing the passive vehicle to the initial aim point at time t_{TPI} plus t_F by means of the Kepler Routine. The Initvel Routine is then called with N_1 and the cone angle ϵ equal to zero and 15° respectively. The active vehicle's velocity \underline{v}_1 and \underline{v}_2 after the TPI maneuver and before the TPF maneuver, respectively, along with the transfer angle θ and the rotation projection switch f_2 (used in the powered flight steering program as S_R) are obtained from this routine. After calculating the maneuver magnitudes, the Periapo Routine is used to calculate the pericenter altitude h_P .

4. The program enters a control sequence whose effect is as follows:

1. If the height of pericenter h_P and its previous value h_{P0} are both less than h_L and furthermore h_P is less than h_{P0} an alarm is displayed indicating that there are no safe pericenters in the sector being searched.

2. If, as before, both $h_P < h_L$ and $h_{P0} < h_L$, but now $h_P > h_{P0}$, the program sets the time of flight increment Δt equal to K400 sec., increments the time of flight, and continues searching.

3. If the pericenter changes from safe to unsafe, the program interpolates to find the time of flight at which the change occurred, and accepts this value as the solution, together with ΔV_{TPI} , ΔV_{TPF} , h_P .

4. If the pericenter changes from unsafe to safe, the program also interpolates and accepts the solution provided ΔV is greater than its previous value ΔV_0 . If not it sets $\Delta t = K150$ sec., increments the time of flight, and continues searching.

5. If the present pericenter is safe and $|\Delta V - \Delta V_0| < 2$ ft/sec. the present time of flight is accepted as the solution.

6. If none of the above conditions holds, the time of flight is incremented by Δt if $\Delta V < \Delta V_0$ or diminished by $\Delta t/2$ if $\Delta V > \Delta V_0$, and the search function continues.

7. If the incremented time of flight t_F exceeds t_{FF} or the active vehicle transfer angle θ exceeds θ_L , the last solution is accepted if a safe pericenter exists, otherwise an alarm code is displayed, indicating no solution in this sector.

5. A sign (SG_2) is affixed to ΔV_{TPF} to indicate whether the active vehicle is ascending or descending on the passive vehicle at TPF and the solution set consisting of ΔV_{TPI} , ΔV_{TPF} , h_P is displayed to the astronaut. If the astronaut is satisfied with these trajectory parameters, he may request that the central transfer angle ωt of the passive vehicle be displayed to be recorded by him as an input parameter for the TPI Pre-thrust program P-34 along with the initial t_{TPI} . In addition, this display will also contain the pericenter code P which indicates whether the pericenter of the transfer trajectory occurs between the TPI and TPF maneuvers or beyond the TPF maneuver. The other central angle sector may be searched by returning to 1 and changing K, or a new trajectory solution attempted by returning to 1 and changing the input TPI time until a satisfactory transfer trajectory has been found.

5.4.2.5 Stable Orbit Rendezvous

The Stable Orbit Rendezvous Program P-38 (CSM active) or P-78 (LM active) is defined by the following three maneuvers. The first maneuver (Stable Orbit Initiation SOI) defines a transfer of the active vehicle to a point on the orbit of the passive vehicle, a given distance ahead of, or behind the passive vehicle, as measured along the passive vehicle orbital path. The second maneuver (Stable Orbit Rendezvous SOR) is designed to place the active vehicle in the same orbit as the passive vehicle with a separation between the two vehicles determined by the first maneuver. The position of the active vehicle relative to the passive vehicle after this second maneuver is referred to as the stable orbit point. The third maneuver defines the final transfer from the stable orbit point to the terminal rendezvous point and is calculated using the pre-TPI maneuver program (Section 5.4.2.2). All the maneuvers are based on precision trajectory computations.

After the SOI maneuver, the Rendezvous Midcourse Maneuver Program (Section 5.4.2.5.3) is available for computing midcourse corrections.

•
•
•
•
•
•
•
•
•

The Stable Orbit Rendezvous Program has two phases, which must be called sequentially to compute the first and second rendezvous maneuvers. The input parameters are:

- 1) Choice of active vehicle (P-38 CSM, P-78 LM).
- 2) Ignition time of maneuver t_{IG} .
- 3) ωt , central angle of travel during the transfer, based on the passive vehicle orbit.
- 4) S (0 for the first maneuver, 1 for the second maneuver).
- 5) Stable orbit displacement distance δr . Positive δr indicates that the stable orbit point is behind the passive vehicle. (Specified only when S = 0).

The active vehicle state vector $\underline{r}_A, \underline{v}_A$ and passive vehicle state vector $\underline{r}_p, \underline{v}_p$ are available in the guidance computer.

The trajectory configuration for the first two maneuvers is illustrated in Fig. 4.2-9 and the program logic is outlined in Fig. 4.2-10. Both phases of the program start with the update of the state vectors to the ignition time and the conversion of ωt to the transfer time t_F .

5.4.2.5.1 Stable Orbit Initiation Maneuver

If S is set equal to zero, the program computes the stable orbit initiation SOI maneuver. The passive vehicle is first updated to the intercept time t_I , equal to t_{IG} plus t_F . This point is next used to convert the stable orbit displacement distance δr to a time δt by approximating the orbit of the passive vehicle with a circular orbit and by using an orbital velocity equal to the vehicle's horizontal velocity. The passive vehicle is now updated to the target time t_T , equal to t_I minus δt , to obtain the target vector \underline{r}_T for the first maneuver (see Fig. 4.2.10).

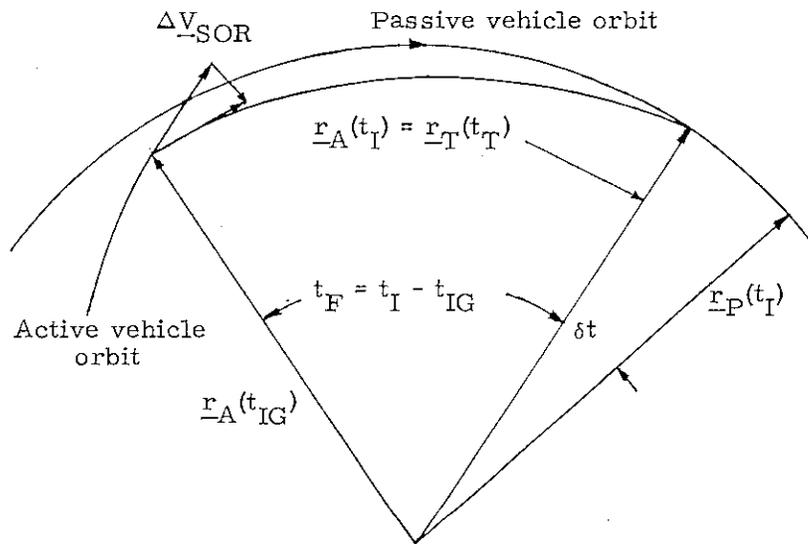
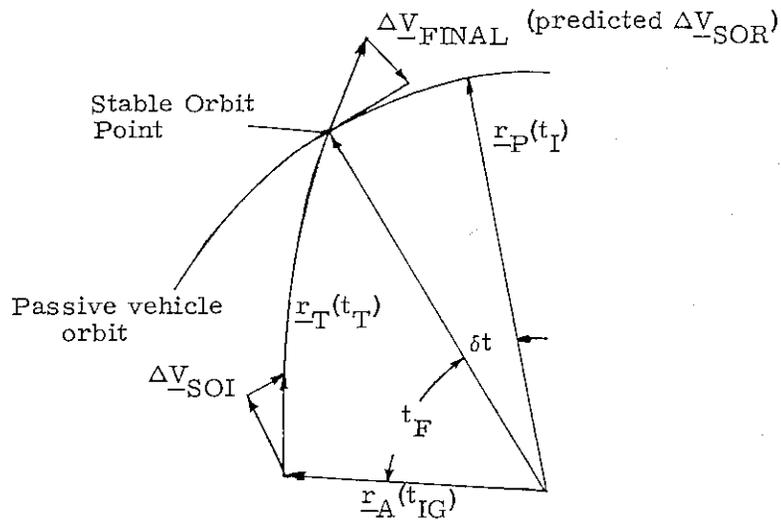


Figure 4.2-9 Geometry of the SOI and SOR Maneuvers

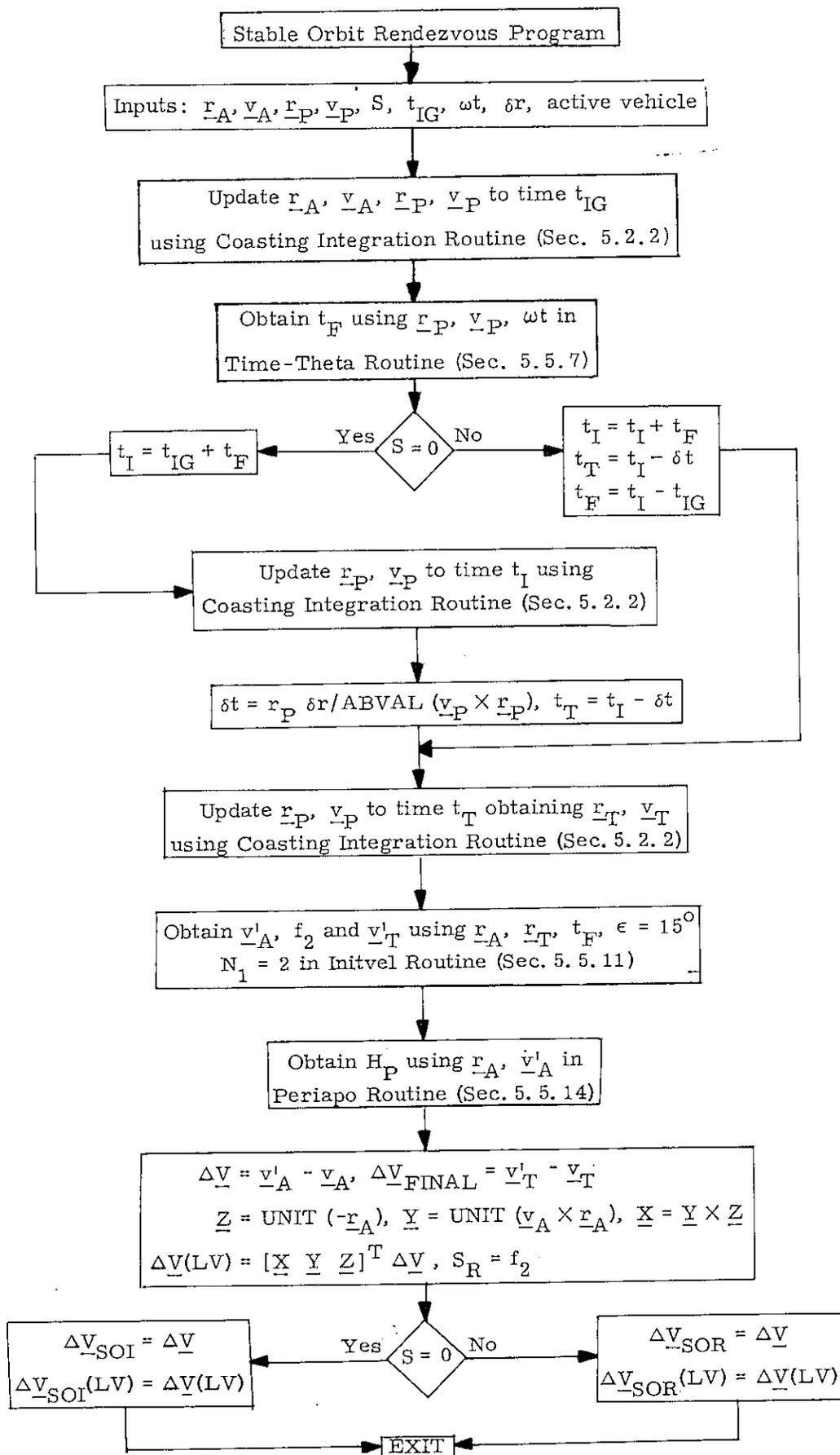


Fig. 4.2-10; Stable Orbit Rendezvous Program

The Initvel Routine is called with N_1 and the cone angle ϵ equal to two and 15° respectively. This results in the precision offsetting procedure to be used twice in succession. The rotation projection switch f_2 is obtained from the Initvel Routine for use in the powered flight steering program as S_R .

The following displays are computed.

- 1) ΔV_{SOI} : ΔV for the SOI maneuver.
- 2) ΔV_{FINAL} : Passive vehicle's velocity minus the active vehicle's velocity, both computed at the stable orbit point.
- 3) $\Delta \ddot{V}_{SOI}(LV)$: Local vertical coordinates
- 4) H_p : Perigee altitude following the stable orbit initiation maneuver.
- 5) t_I : Time the active vehicle intercepts the stable orbit point.

5.4.2.5.2 Stable Orbit Rendezvous Maneuver

If S is set equal to one, the program computes the stable orbit rendezvous SOR maneuver. In order for this maneuver to place the active vehicle in the orbit of the passive vehicle in a precise manner, the SOR ignition time should be offset to account for the finite burn time. When the transfer trajectory time t_F is added to the calculated SOI intercept time t_I (stored in the SOI maneuver program), the SOR intercept time t_I is obtained (see Fig.4.2-10).

The target vector \underline{r}_T for the SOR maneuver can be obtained by updating the passive vehicle to the target time t_T , equal to t_I minus δt (δt is stored in the SOI maneuver program). The transfer time t_F , based on the offset ignition time, is equal to the new intercept time minus the ignition time.

After using the Initvel Routine, as in the Stable Orbit rendezvous maneuver, to compute the transfer trajectory, the following displays are calculated:

- 1) ΔV_{SOR} : ΔV for the SOR maneuver
- 2) ΔV_{FINAL} : Approximately equal to zero
- 3) $\Delta V_{\text{SOR}}(\text{LV})$: Local vertical coordinates
- 4) H_{P} : Perigee altitude following the SOR maneuver

5.4.2.5.3 Stable Orbit Midcourse Maneuver

This program, corresponding to Program P-39 (CSM active) or P-79 (LM active) of Section 4, computes a midcourse correction maneuver. The astronaut may call this program after completing the SOI maneuver, but in general no later than 10 minutes before the intercept time. The program logic is outlined in Fig. 4.2-11.

There is one astronaut input: choice of the active vehicle (P-39 CSM, P-79 LM). The active vehicle state vector \underline{r}_A , \underline{v}_A , the passive vehicle state vector \underline{r}_P , \underline{v}_P , the time delay, the target time t_T and the intercept time t_I are available in the guidance computer. The time delay ($\delta \tau_3$ for P-39 and $\delta \tau_7$ for P-79) is the time required to prepare for the thrust maneuver and is stored in either of two erasable locations.

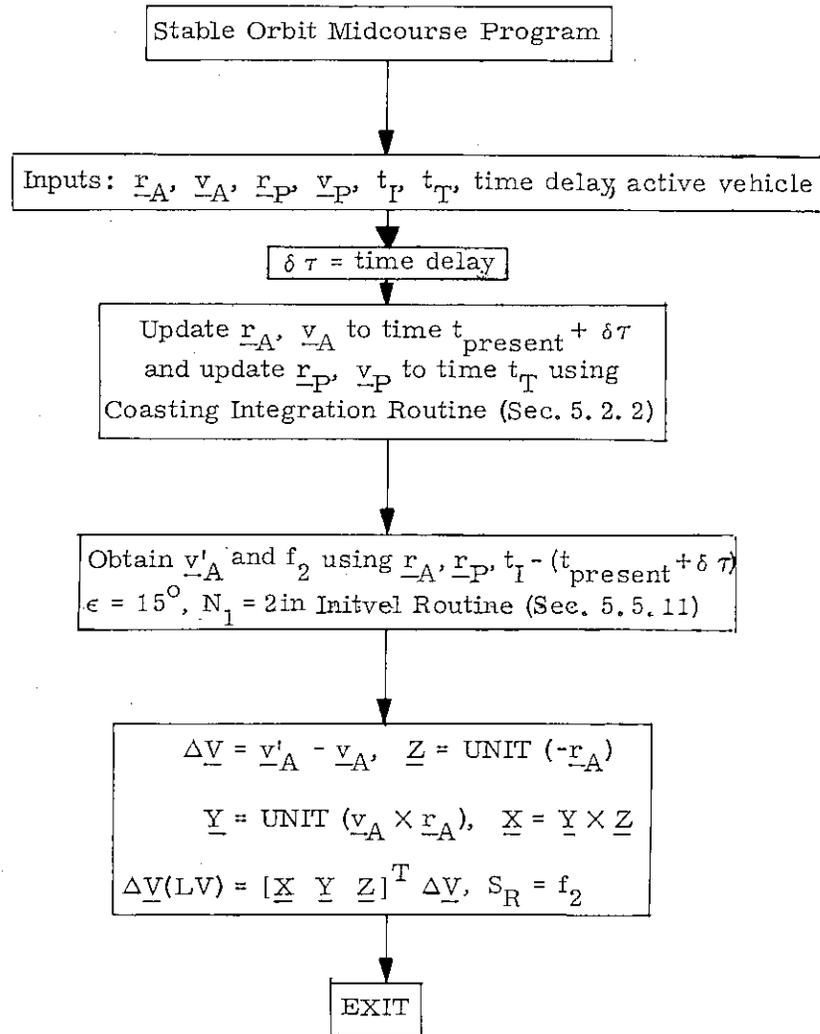


Fig. 4.2-11 Stable Orbit Midcourse Program

When the program is initiated the number of navigation measurements processed since the last maneuver and the time to ignition are displayed. Based on this information and additional displays discussed in Section 5.6.7, the astronaut may elect to proceed with the midcourse maneuver at some time. When he does so, the program updates the active vehicle to the present time plus the time delay $\delta \tau$ and updates the passive vehicle to the target time using the Coasting Integration Routine. The Initvel Routine is then called with a transfer time equal to the intercept time minus the ignition time to obtain the initial velocity v'_A for the transfer ellipse. The N_1 switch is set equal to two, resulting in a double use of the offsetting procedure. After obtaining the maneuver ΔV , it is rotated into a local vertical coordinate system, resulting in the display $\Delta V(LV)$. In addition, M, MGA and TPI are displayed as in the Pre-TPI program displays.

5.4.3 RETURN TO EARTH TARGETING

5.4.3.1 Introduction

The Return to Earth Targeting Program provides to the astronaut the ability to compute on board the spacecraft the targeting for a powered maneuver which returns the spacecraft to an earth reentry point with proper conditions for a safe reentry. This program is intended as a backup in the event of a communication or ground system failure and, thus, is completely independent of inputs from the ground system. The method of computation, however, does not compromise certain critical trajectory characteristics such as fuel usage and transit time for the expediency of minimum computer storage usage or computational speed.

Specifically this program provides powered maneuver targeting for :

- (1) Returning from near earth orbits
- (2) Returning from trajectories resulting from a translunar injection powered maneuver failure.
- * (3) Returning from translunar midcourse

* Outside moon's sphere of influence only

- * (4) Returning from transearth midcourse
- * (5) Midcourse corrections during transearth midcourse
- * (6) Changes in transit time during transearth midcourse

Two modes of operation are provided: time critical and fuel critical. The former establishes trajectories which return to earth in a minimum of time, the latter, with a minimum of fuel.

The return trajectory is constrained to be in the plane which the spacecraft was in prior to the return so that optimum fuel utilization will be achieved.

Although explicit control of the spacecraft landing site is not a feature of this program, it is possible to achieve landing site selection through adjustment of the astronaut input to the program. When on translunar or transearth trajectories, the adjusted parameter is the desired fuel utilization. When in orbit near the earth, the designated return time is the adjusted parameter. The astronaut need only to note the earth track of the landing site as a function of the adjusted parameter to select the proper parameter value.

The computations take place in two steps. The first is the generation of a conic two body solution to the problem. This takes a relatively short computational time. The second step, the computation of a precise trajectory, takes a relatively long computational time.

The Return to Earth Program is described in increasing detail in the following subsections:

Sec. 5.4.3.2	Program Input-Output	pg. 5.4-46
Sec. 5.4.3.3	General Computational Procedure	pg. 5.4-48
Sec. 5.4.3.4	Logical Description	pg. 5.4-58
Sec. 5.4.3.5	Detailed Flow Diagrams	pg. 5.4-71

5.4.3.2 Program Input-Output

The Return to Earth Program may be called by the astronaut at any time the computer is operable. Should the designated time of return place the spacecraft at a point in space where the program is incapable of producing a successful solution, an alarm will occur. The current calling designation for this program is P-37 for both modes.

Return-to-Earth Targeting

After calling the program, the astronaut must load the following information into erasable memory via the keyboard.

- (1) objective velocity change to be utilized during the maneuver
- (2) time the maneuver is to be executed
- (3) reentry angle (normally set to zero and internally calculated*)

Should a minimum fuel return be desired, the objective velocity change should be set to zero. Should the objective change be less than the minimum required, the program will select the minimum. When the objective velocity change is greater than the minimum, the return will require that amount of velocity change.

This program is not designed to operate from low altitude polar orbits unless the astronaut selects the reentry angle.

The program provides the following displays when both the conic trajectory computation and the precision trajectory computation are complete.

- (1) latitude of the landing site
- (2) longitude of the landing site
- (3) velocity change required (vector)

*See 5.4.3.4.2 for detailed explanation of its use

- (4) velocity magnitude at 400,000 ft. entry altitude measured above the Fischer ellipsoid.
- (5) flight path angle at 400,000 ft. entry altitude
- (6) transit time to 400,000 ft. entry altitude from ignition.

The astronaut must select either the SPS or RCS engine after the precision trajectory calculation. The following quantities are displayed after the engine selection:

- (7) middle gimbal angle at ignition
- (8) time of ignition
- (9) time from ignition

The precision trajectory displays can be significantly different from the conic displays, especially in the case of long transit time returns; therefore the astronaut should check the new displays before accepting the precise solution.

The cross product steering constant, c , of Section 5.3.3.4 is set equal to 0.5 in this program. In addition the rotation switch, S_R , which is used in the Lambert Aim Point pre-thrust computation of Section 5.3.3.3.2 is set to zero.

5.4.3.3 General Computational Procedure

This section contains a general description of the Return-to-Earth computational procedure as illustrated in Figures 4.3-1, 4.3-2, and 4.3-3.

The Return to Earth Targeting Program, as previously mentioned, may be used in either the fuel critical or time critical mode. When used in the fuel critical mode, the program generates a trajectory which meets specified constraints on reentry flight path angle and reentry altitude, and minimizes the impulsive velocity change required to achieve this trajectory. In the time critical mode, the program generates a trajectory which meets the specified reentry constraints and returns the spacecraft in the shortest possible time. The astronaut selects either mode by entering via the keyboard the objective velocity change (Δv_D). If Δv_D is set to zero, the routine will provide a fuel critical trajectory. If Δv_D is not set to zero, the routine will provide a time critical trajectory if possible, or select a fuel critical trajectory if Δv_D is less than the minimum required impulsive velocity change. In addition, during transearth coast, a negative Δv_D may be selected -- resulting in a slower return, i. e., slower than the minimum fuel return.

The Return to Earth Targeting Program has two major phases, an initial two body conic phase illustrated in Figure 4.3-1 and a numerically integrated precision trajectory phase illustrated in Figure 4.3-2. A conic routine which provides initial conditions for return trajectories for both phases is illustrated in Figure 4.3-3.

5.4-48

Revised COLOSSUS
 Added GSOP # R-577 PCR # 690 Rev. 5 Date 1-21-69

5.4.3.3.1 Nomenclature

The following is a list of the nomenclature used in Figs. 4.3-1, 4.3-2, and 4.3-3.

$\underline{y}(t_0) = (\underline{x}(t_0), \underline{v}(t_0))$	CSM state vector maintained in the CMC
$\underline{y}_1(t_1) = (\underline{x}(t_1), \underline{v}_1(t_1))$	pre-return state vector
$\underline{y}_2(t_1) = (\underline{x}(t_1), \underline{v}_2(t_1))$	post-return state vector
$\underline{y}(t_2) = (\underline{x}(t_2), \underline{v}(t_2))$	state vector at reentry
$\underline{y}(t_2)_{\text{PRE}}$	final state vector of a precision trajectory
R_{CON}	final radius of a conic trajectory
R_{PRE}	final radius of a precision trajectory
R_{D}	desired final radius (reentry radius)
$\gamma(t_2)_{\text{CON}}$	final flight path angle of a conic trajectory
$\gamma(t_2)_{\text{D}}$	desired final flight path angle
$\gamma(t_2)_{\text{PRE}}$	final flight path angle of a precision trajectory
$x(t_1)$	cotangent of the post-return flight path angle
Δv	impulsive velocity change
Δv_{D}	desired velocity change
ϵ_1 to ϵ_{10}	tolerances

t_{21} time of flight ($t_2 - t_1$)
 n_1 counter
 f_1 flag set to $\left\{ \begin{array}{l} 0 \text{ initial conic phase} \\ 1 \text{ precision trajectory phase} \end{array} \right.$

5.4.3.3.2 Discussion of the Conic Phase

The following discussion is intended to supplement the information contained in Fig. 4.3-1.

The pre-return CSM state vector ($\underline{y}_1(t_1)$) is obtained by numerical integration. An initial conic trajectory is then generated which meets the reentry and velocity change constraints of the problem. The reentry radius and flight path angle constraints, however, are known only as a function of the latitude and velocity magnitude at reentry. Therefore an iterative procedure is employed to generate this initial conic trajectory. Based on the pre-return radius magnitude, an estimate of the reentry constraints is made. This initial estimate is used to compute a conic trajectory which satisfies the Δv_D requirement. The final velocity magnitude and latitude are now available for use in a more precise calculation of the reentry constraints. An iterative process has now been started which quickly converges on a suitable initial conic trajectory.

During the above iterative process slight changes in the reentry flight path angle can produce significant changes in the minimum required velocity change. If the time critical mode is being used, it is possible that during a particular pass through the iteration loop the velocity change desired may be less than the minimum required. This does not necessarily preclude a time critical solution, and therefore the iteration is continued using the minimum required velocity change.

A description, in general terms, of the procedure CONICRETURN, used to obtain the initial state vector of the conic return trajectory, will follow later in this section.

After an initial conic trajectory is computed, an estimate is made of the landing site, and the results are displayed. Based on this display the astronaut may elect to continue with the calculation of a numerically integrated trajectory.

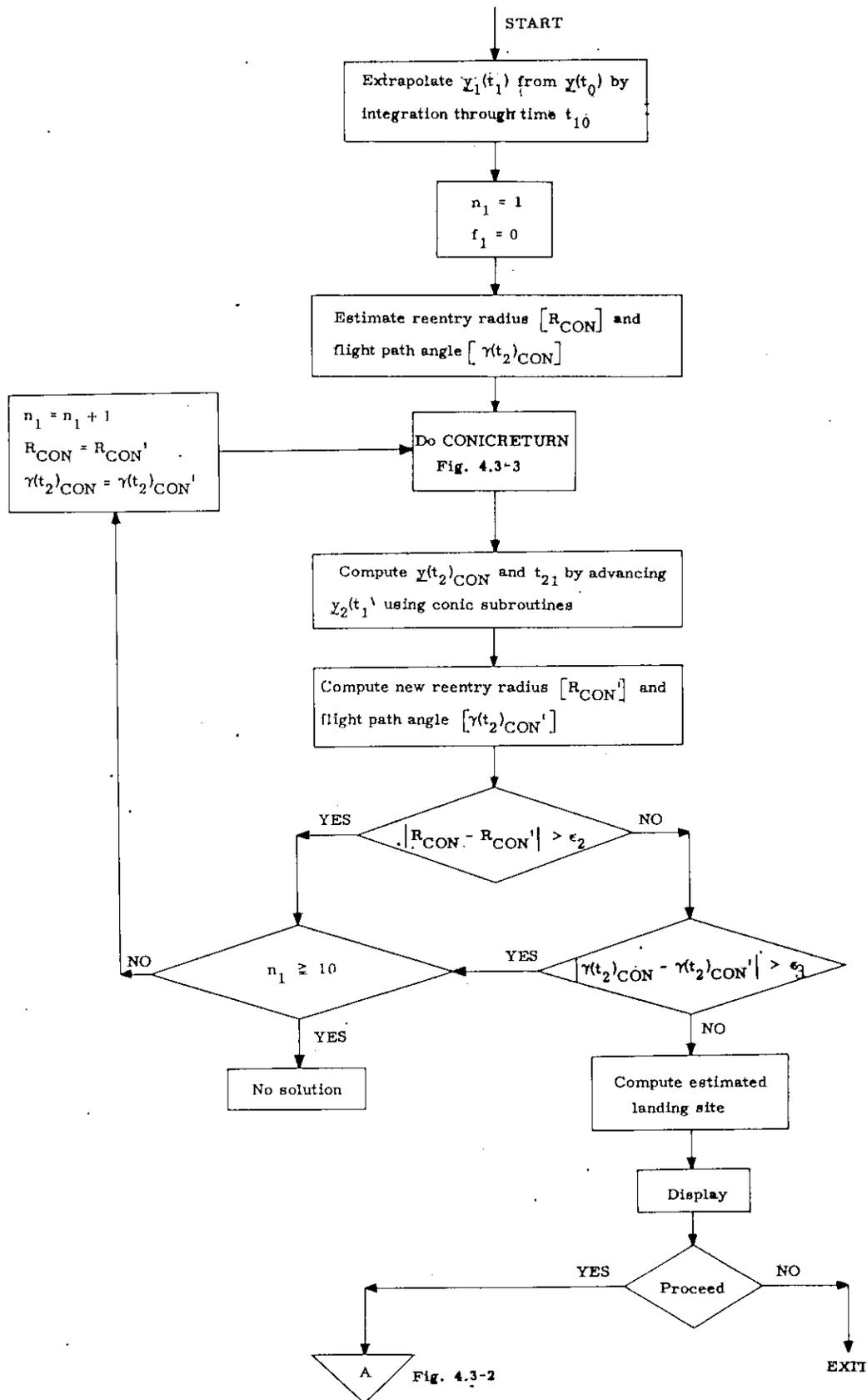


FIGURE 4.3-1 SIMPLIFIED LOGIC DIAGRAM OF RETURN TO EARTH TARGETING

5.4.3.3.3 Discussion of the Precision Phase

The following discussion is intended to supplement the information contained in Fig. 4.3-2.

The numerically integrated trajectory must satisfy the reentry constraints achieved by the initial conic trajectory and meet the Δv_D requirement as closely as possible. An iterative procedure is again employed to generate this trajectory. The post-return state vector of the previous conic trajectory is used to initialize the Coasting Integration Routine. The integration proceeds until the desired reentry flight path angle has been achieved. The reentry radius of the next conic trajectory is then offset to compensate for the error in the reentry radius of the numerically integrated trajectory. An iterative process has now been initiated which converges to the precise trajectory which meets the objective velocity change and the reentry constraints. Next a target is computed for the Lambert Aimpoint Powered Flight Routine, the landing site is calculated, and the results are displayed.

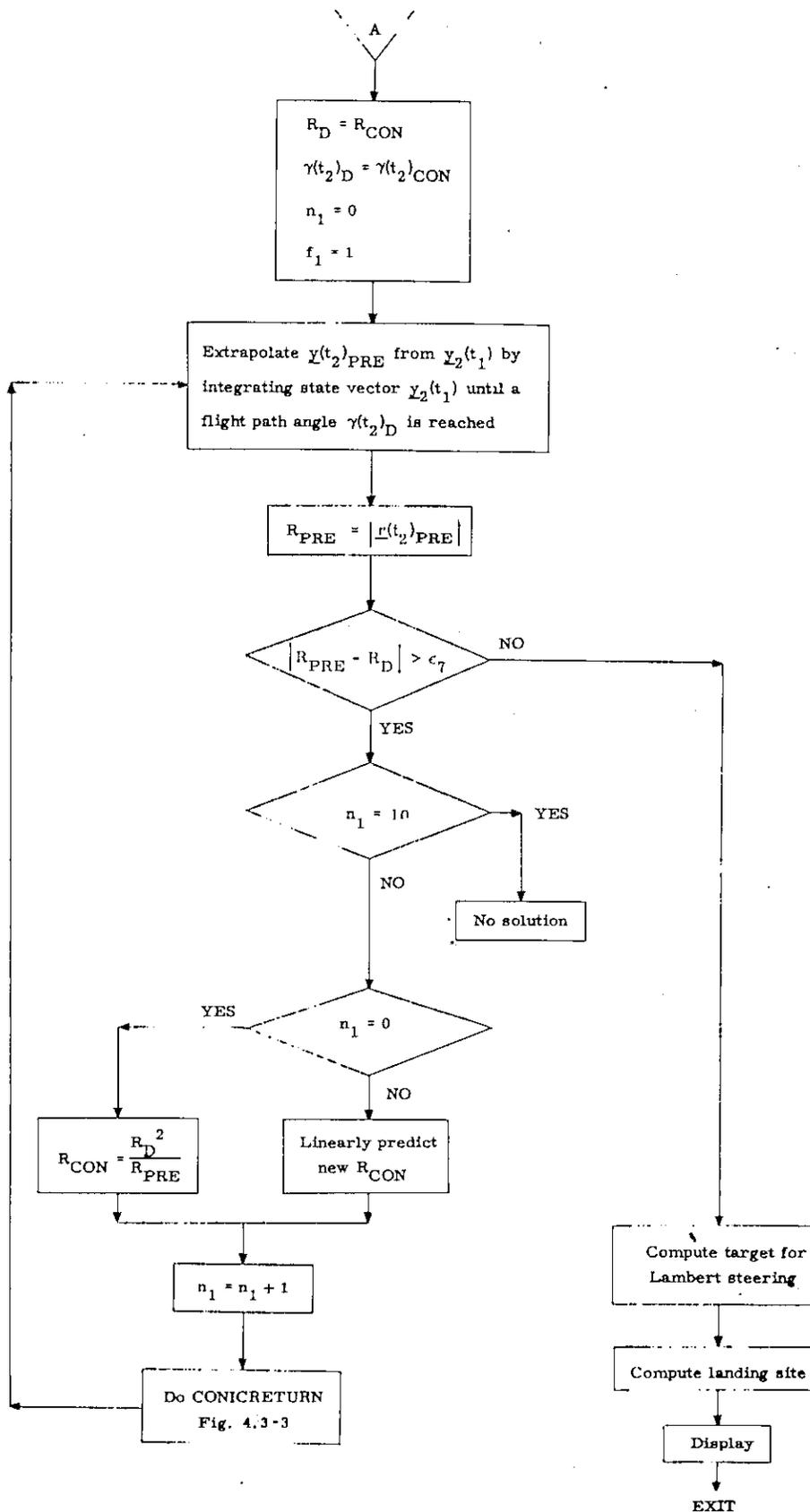


FIGURE 4.3-2 SIMPLIFIED LOGIC DIAGRAM OF RETURN TO EARTH TARGETING

5.4.3.3.4 Discussion of the Conic Routine

The following discussion is intended to supplement the information contained in Fig. 4.3-3.

The conic routine solves the following two conic problems.

- (1) Given an initial position vector ($\underline{r}(t_1)$), a pre-return velocity vector ($\underline{v}_1(t_1)$), a final flight path angle ($\gamma(t_2)_{CON}$), and a final radius magnitude (R_{CON}), compute a post-return velocity vector ($\underline{v}_2(t_1)$) which minimizes the velocity change (Δv).
- (2) Given the above quantities, compute a post-return velocity vector ($\underline{v}_2(t_1)$) which meets the objective velocity change (Δv_D).

An iterative procedure is again used. The independent variable of the procedure is the cotangent of the post-return flight path angle ($x(t_1)$), and the dependent variable is the velocity change (Δv).

Initial bounds on $x(t_1)$ are computed based on the maximum allowable semi-major axes. The upper bound constrains return trajectories which pass through apogee from going too near the moon, and the lower bound keeps the reentry velocity for direct return flights from exceeding the maximum allowable reentry velocity.

During the second phase of the Return to Earth Targeting Program, the calculation of a numerically integrated trajectory, additional bounds are imposed on the independent variable $x(t_1)$. In the case of a return from earth orbit, the resulting trajectory may be nearly circular, and will pass through the desired reentry flight path angle twice at nearly the same radius. Additional limits on $x(t_1)$ are computed to prevent conic trajectories generated during the search for a numerically integrated trajectory from oscillating between these two solutions.

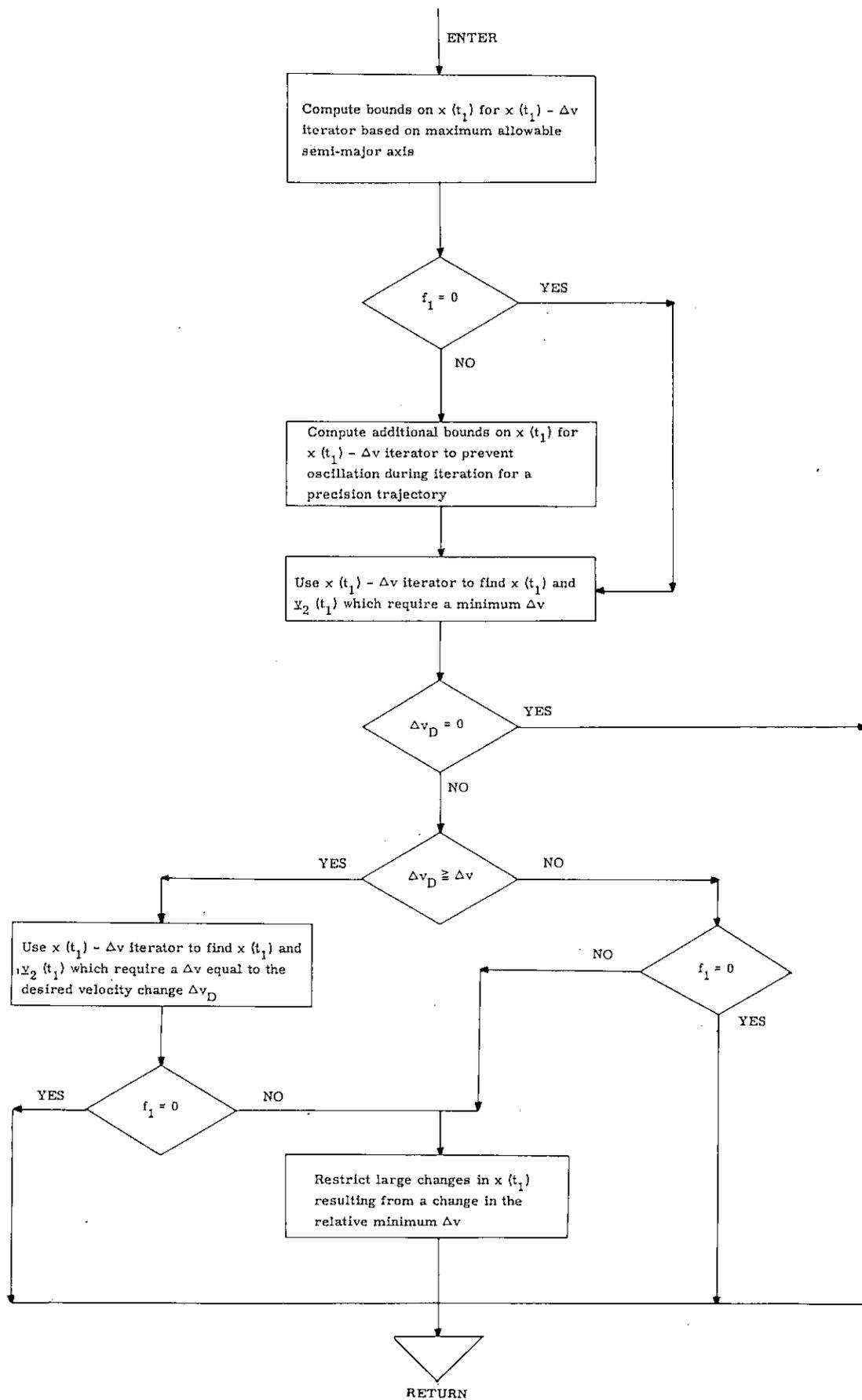


FIGURE 4.3-3 SIMPLIFIED LOGIC DIAGRAM OF RETURN TO EARTH TARGETING
- CONIC RETURN ROUTINE

5.4-56

Rev. 5 - 3/69

In all cases the $x(t_1) - \Delta v$ iterator is used to compute a post-abort velocity vector which requires a minimum Δv . In the event that a time critical answer is required, another iteration using the $x(t_1) - \Delta v$ iterator will be made to compute a post-return velocity vector which requires a Δv equal to Δv_D . If the velocity change desired is less than the minimum required, the second iteration will be omitted.

During the calculation of the numerically integrated trajectory, the intermediate conics are additionally restricted to prevent large changes in the post-return flight path angle. This additional restraint is necessary in the time critical mode for certain near earth returns because both absolute and relative minimum Δv solutions exist.

5.4.3.4 Logical Description

This section contains logical flow diagrams of the outside the sphere portion of the Return to Earth Targeting Program. Some items not fully discussed in the text are explicitly illustrated in the diagrams. Certain self contained procedures are detailed in the next section.

5.4.3.4.1 Nomenclature

A list of the nomenclature used in this section and not previously defined follows.

t_{IG}	desired ignition time
t_{21}	time of flight ($t_2 - t_1$)
$p(t)$	primary body $\left\{ \begin{array}{l} 0 \text{ for earth} \\ 1 \text{ for moon} \end{array} \right.$
$x(t_1)_{LIM}$	limit on the cotangent of the post-return flight path angle
$x(t_1)_{MAX}, x(t_1)_{MIN}$	upper and lower bounds on the cotangent of the post-return flight path angle
$\Delta x(t_1)$	change in $x(t_1)$
$\Delta x(t_1)_{MAX}$	maximum allowed change in $x(t_1)$
$x(t_2)$	final cotangent of the flight path angle
$x(t_2)_{PRE}$	final cotangent of the flight path angle for a precision trajectory
P_{CON}	semi-latus rectum of a conic trajectory
R_{ERR}	precision trajectory reentry radius error

μ_E earth's gravitation constant

MA_1, MA_2 maximum major axes

\underline{n} normal vector

ϕ_2 $\left\{ \begin{array}{l} + 1 \text{ near apogee solution} \\ - 1 \text{ near perigee solution} \end{array} \right.$

ϕ_4 precision trajectory direction switch

f_2 iterator mode flag

n_1, n_2 counters

β_1 to β_{14} intermediate variables

5.4.3.4.2 Discussion of the Logical Flow Diagram

The following discussion is intended to supplement the information contained in the referenced diagrams.

Three inputs are required in addition to the CSM state vector (Fig. 4.3-4). They are the objective velocity change desired (Δv_D), the desired time of ignition (t_{IG}), and the reentry angle ($\gamma(t_2)_D$). The reentry angle should nominally be set to zero, allowing the program to compute a reentry angle in the center of the reentry corridor. The astronaut may select the reentry angle if a minimum Δv solution lying on the edge of the reentry corridor is required. This may be necessary in the event the reaction control system is used for the return. See Fig. 4.3-15 for a detailed explanation of the reentry angle calculations.

The Coasting Integration Routine is used to advance the state vector to the desired return time. A check is made to insure that the state vector is not inside the moon's sphere of influence. If the state vector is outside the sphere of influence, precomputations are executed. They are explained in detail later. A conic trajectory is then generated which satisfies the estimated reentry constraints and Δv_D requirements. The procedure CONICRETURN, which computes the initial state vector of this conic, will be discussed later. The Time Radius Subroutine, described in Section 5.5.8, is used to compute the time of flight to the desired radius and advance the state vector through that time of flight.

Based on this final state vector (Fig.4.3-5), a new reentry radius (R_{CON}') and a new value for the cotangent of the reentry flight path angle ($x(t_2)'$) can be computed. If they are sufficiently close to the previously used values, the conic trajectory is accepted. Otherwise a new conic is computed which satisfies the new reentry constraints. A counter is employed to protect against an excessive number of iterations.

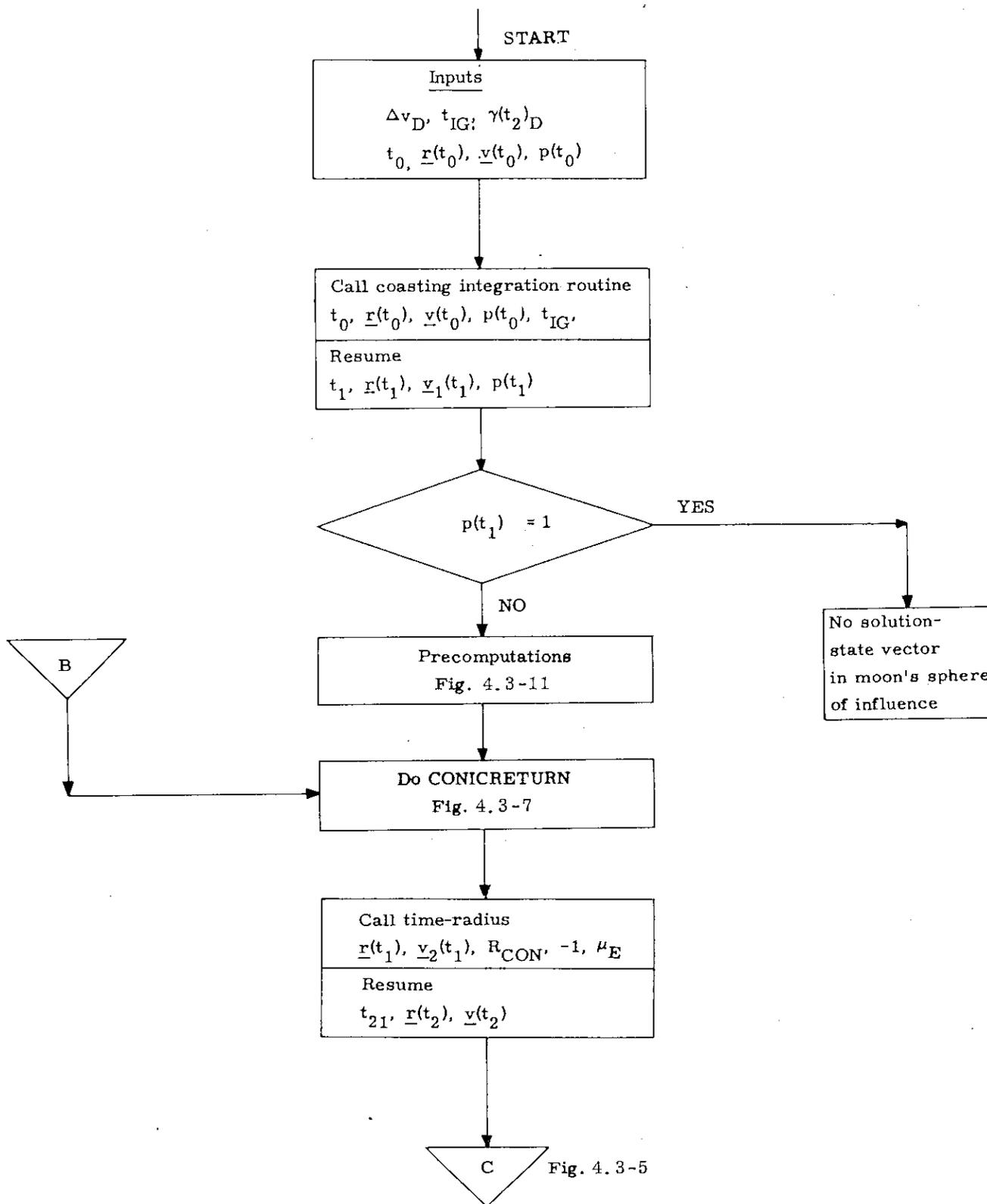


FIGURE 4.3-4 LOGICAL DIAGRAM OF RETURN TO EARTH TARGETING

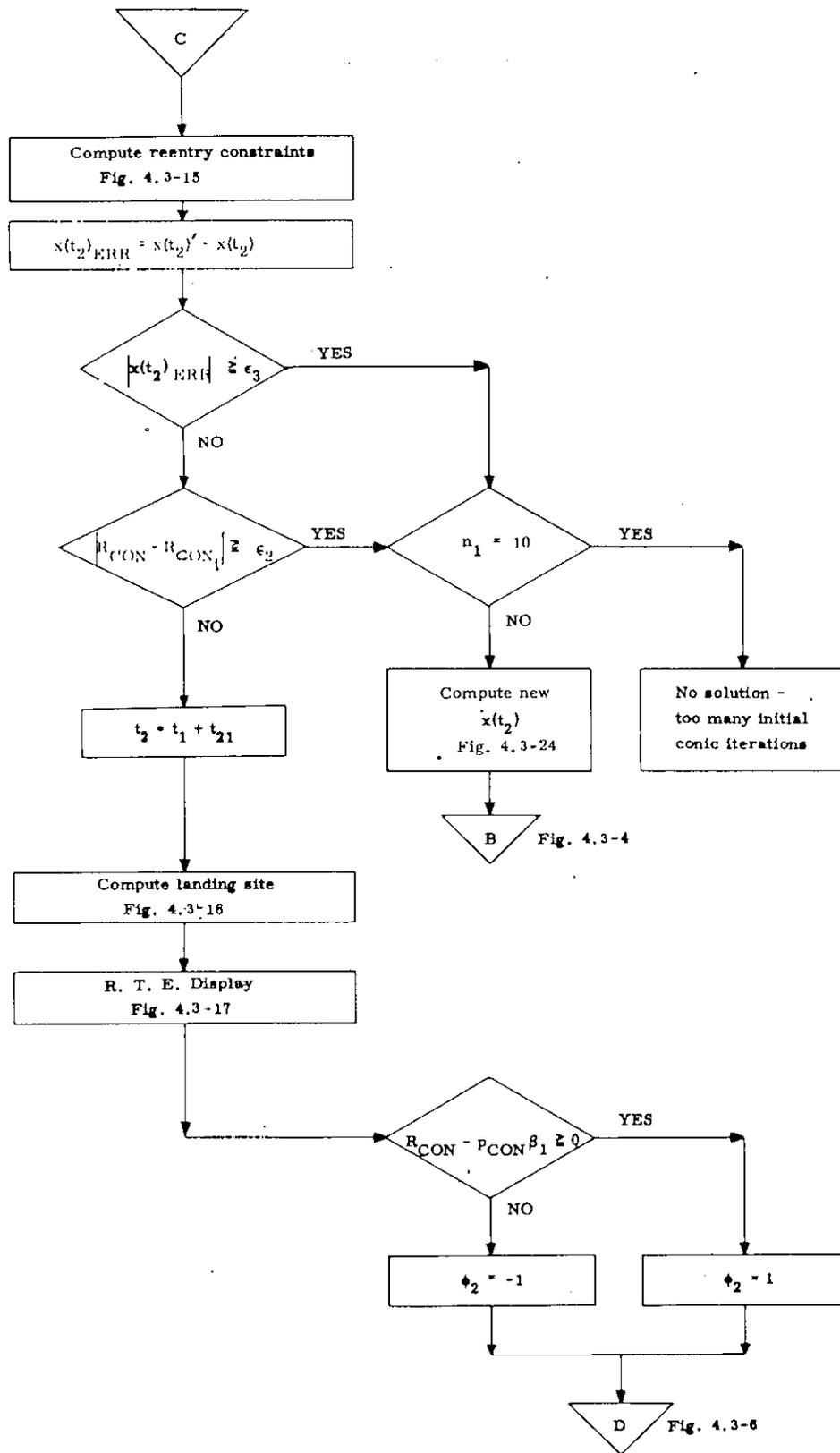


FIGURE 4.3-5 LOGICAL DIAGRAM OF RETURN TO EARTH TARGETING

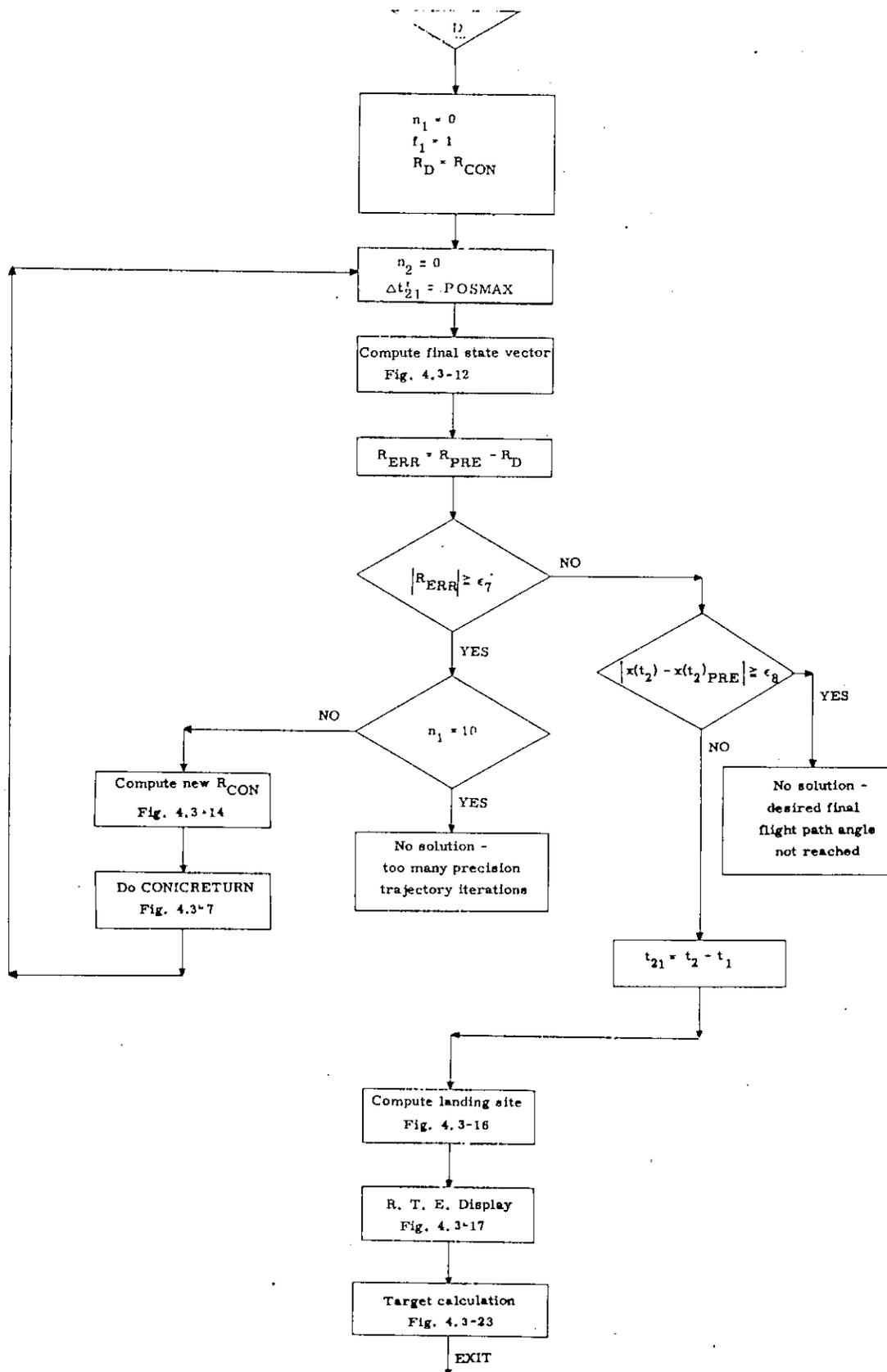


FIGURE 4.3-6 LOGICAL DIAGRAM OF RETURN TO EARTH TARGETING

If the initial conic is accepted, an estimate is made of the latitude and longitude of the landing site. For long flight times, however, differences in time of flight between the conic and numerically integrated trajectory will cause an error in the conically estimated landing site longitude. The results of the preceding calculations are displayed, and based on this information the astronaut may elect to proceed with the calculation of a numerically integrated trajectory or exit the program. As explained in Section 5.4.3.3, return trajectories from near earth orbits may be nearly circular, and therefore pass through the desired flight path angle twice with approximately the same radius. The unique solution reached by the initial conic is tested to determine whether the final state vector was achieved during the first or second pass through the desired reentry flight path angle. A switch ϕ_2 is set appropriately, and with one exception all succeeding precision and conic trajectories are required to behave in a similar manner. The exception will be explained in detail in Section 5.4.3.4.3.

The post-return state vector of the conic trajectory is now advanced until the desired flight path angle of reentry is reached (Section 5.4.3.5). A radius error will exist between the desired final radius (R_D) and the final radius magnitude of the numerically integrated trajectory. If this error is small, the precision trajectory is accepted. Otherwise a new conic trajectory is generated which uses an offset final radius magnitude. If the error in the final radius is small, a final check is made to insure that the reentry flight path angle is acceptable.

5.4.3.4.3 Discussion of Logical Flow Diagram - CONICRETURN

The procedure CONICRETURN, illustrated in Figures 4.3-7 and 4.3-8, is used to generate initial state vectors for the Return to Earth Targeting Program. Equations required to initialize CONICRETURN each time it is used are described in Section 5.4.3.5. The conic problems solved by the routine were described in Section 5.4.3.3.4.

The logical structure shown in Figs. 4.3-7 and 4.3-8 is based in the following facts.

- (1) There is only one minimum if the pre-return velocity vector has a negative radial component.
- (2) There may be two minima if the pre-return velocity vector has a positive radial component. In this case the absolute minimum post-return velocity vector has a positive radial component and the relative minimum has a negative component.

As explained in Section 5.4.3.3.4 and 5.4.3.4.2, it is necessary to compute additional bounds on $x(t_1)$ during the search for a precision trajectory. The logic used to do this is shown on the lower portion of Fig. 4.3-7.

A check is made to determine whether the locus of possible solutions includes trajectories which achieve the radius R_{CON} and the flight path angle $\gamma(t_2)$ near apogee, and if so the variable β_6 will be positive*.

*With one exception every conic solution generated by this routine achieves the desired reentry flight path angle ($\gamma(t_2)$) twice during the passage from apogee to perigee. The first time $\gamma(t_2)$ is reached during the passage from apogee to perigee will be referred to as near apogee, and the second time will be referred to as near perigee. The desired reentry angle will be achieved only once if $\gamma(t_2)$ equals the maximum flight path angle of the conic.

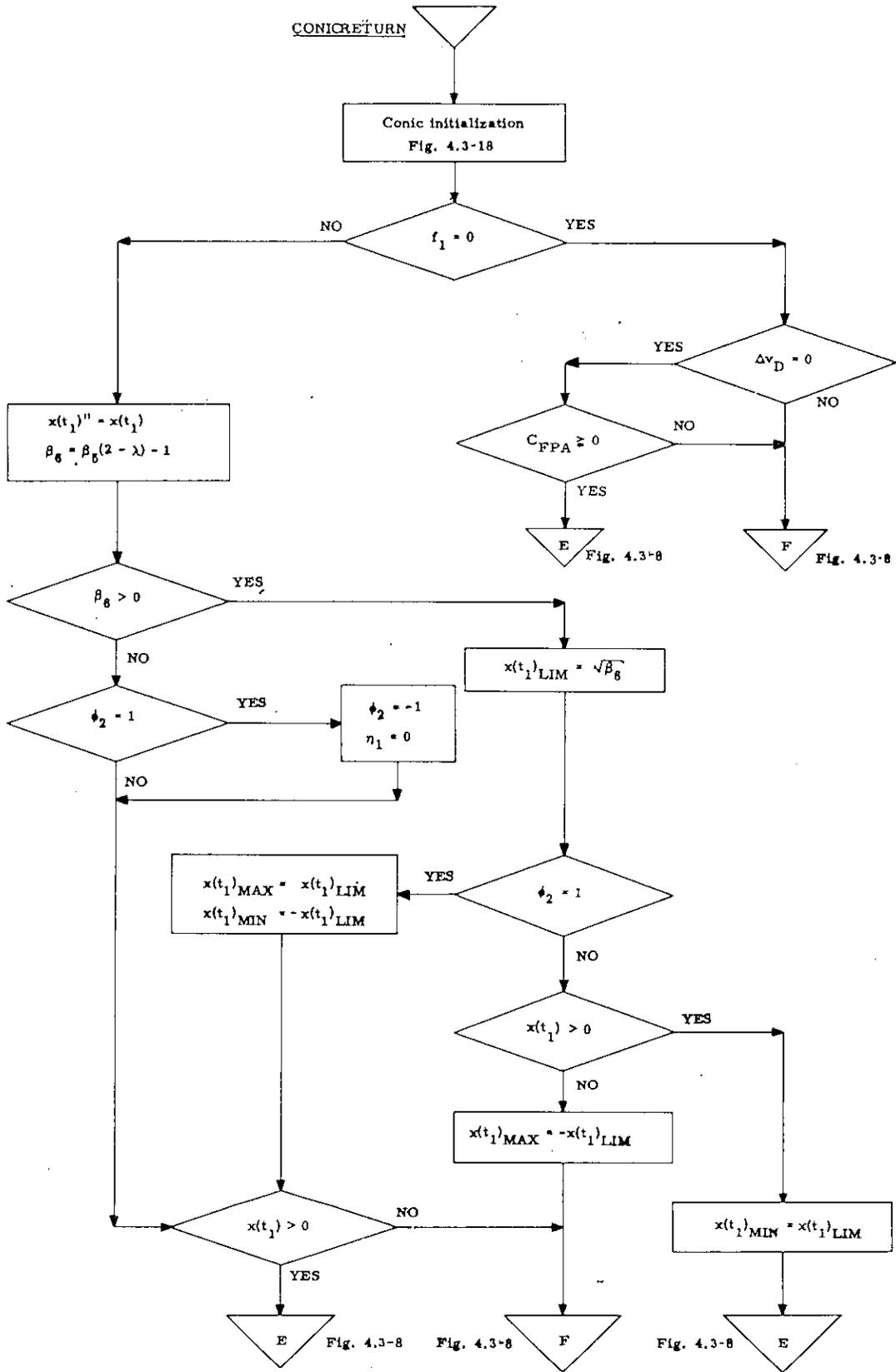


FIGURE 4.3-7 LOGICAL DIAGRAM OF RETURN TO EARTH TARGETING - CONIC RETURN ROUTINE

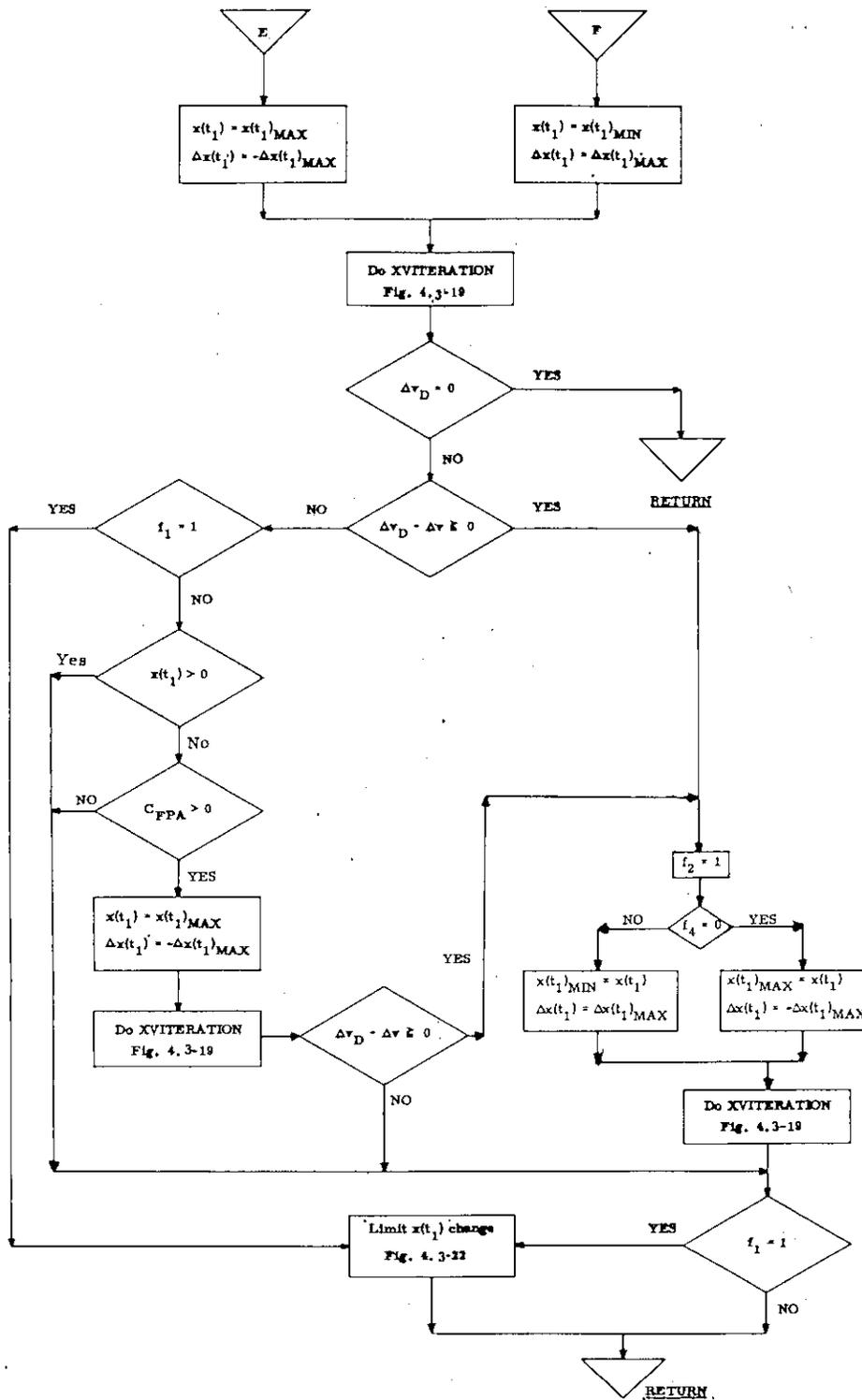


Figure 4.3-8 Logical Diagram of Return to Earth Targeting --
Conic Return Routine

5.4-68

Revised

COLOSSUS

Added

GSOP #R-577

PCR #690

Rev. 5

Date 1-21-69

If β_6 is negative, only near perigee solutions are possible and hence no additional bounds on $x(t_1)$ are computed. The initial conic trajectory may have achieved the desired reentry flight path angle near apogee, however, and this type of solution is no longer possible due to a change in R_{CON} . This is the one exception mentioned in Section 5.4.3.4.2. The value of ϕ_2 must be set to -1, forcing all subsequent conic solutions to reach the desired reentry flight path angle near perigee, and then the entire precision trajectory phase is recycled.

If β_6 is positive, it is used to calculate $x(t_1)_{LIM}$. Three types of solutions may exist.

$$(1) -x(t_1)_{LIM} < x(t_1) < x(t_1)_{LIM}$$

If $x(t_1)$ takes a value in this region, the desired reentry flight path angle ($\gamma(t_2)$) will be achieved near apogee.

$$(2) x(t_1) < -x(t_1)_{LIM} \text{ or } x(t_1)_{LIM} < x(t_1)$$

If $x(t_1)$ takes a value in these regions, the desired reentry flight path angle ($\gamma(t_2)$) will be achieved near perigee.

$$(3) x(t_1) = x(t_1)_{LIM} \text{ or } x(t_1) = -x(t_1)_{LIM}$$

If $x(t_1)$ takes on either of these values, the desired reentry flight path angle ($\gamma(t_2)$) will equal the maximum angle achieved by the conic.

The above information is used to set new bounds on $x(t_1)$ which will force the conic to reach a near apogee or near perigee solution as prescribed by the initial conic.

If the program is being operated in the fuel critical mode, the $x(t_1) - \Delta v$ iterator is used to compute an absolute minimum Δv solution.

If the program is being operated in the time critical mode, the $x(t_1) - \Delta v$ iterator will be used to compute a solution requiring a velocity change equal to the objective velocity change.

5.4.3.5 Detailed Flow Diagrams

Certain portions of the flow diagrams mentioned in the previous sections are illustrated in Figs. 4.3-11 through 4.3-24.

A list of nomenclature used in the following diagrams and not defined previously follows:

λ	radius ratio
p	semi-latus rectum
α	1/semi-major axis
$\underline{r}_2(t_2)$	target aim point
\underline{r}_{LS}	vector through landing site
θ_{LONG}	landing site longitude
θ_{LAT}	landing site latitude
\dot{m}	mass flow rate
F	thrust
f_3	flag set to $\begin{cases} 0 \text{ RCS jets} \\ 1 \text{ SPS engine} \end{cases}$
f_4	flag set for transearth coast slowdown ($-\Delta v_D$)
f_5	flag set to $\begin{cases} +1 \text{ posigrade} \\ -1 \text{ retrograde} \end{cases}$
t_{LS}	time of landing
Δt_B	length of burn
$\Delta \underline{v}_{LV}$	velocity change vector in local vertical coordinates
a_{MG}	middle gimbal angle
$\theta_1, \theta_2, \theta_3$	intermediate variables
Δt_{21}	adjustment of t_{21}
ΔR_{CON}	change in the final radius of a conic trajectory
P37RANGE	$\begin{cases} =0 \text{ AUGKUGL entry range used} \\ \neq 0 \text{ actual entry range used} \end{cases}$

5.4-71

Revised COLOSSUS

Added GSOP #R-577 PCR # 690, 691, 261 Rev. 5 Date 11-19-68

The diagrams are essentially self-explanatory; however, a discussion of two of the diagrams will follow:

Precomputations

The equations illustrated in Figure 4.3-11 are used to compute basic variables required during the remaining portions of the program.

As mentioned in the introduction, the pre-return and post-return state vectors are required to lie in the same plane. If the pre-return position and velocity vectors are nearly colinear, the plane of the return trajectory will be defined to prevent reentry near the earth's poles.

Retrograde return to earth trajectories are not allowed.

Final State Vector Computation

The method illustrated in Figures 4.3-12 and 4.3-13 is used to extrapolate a post-return state vector to the desired reentry flight path angle. An iterative procedure is again used.

First the post-return state vector is advanced through the best estimate of the transit time to reentry using the Coasting Integration Routine. Since it is not necessary to explicitly compute the flight path angle of the resulting reentry state vector, the cotangent of the flight path angle is compared with the cotangent of the desired reentry flight path angle. If they are sufficiently close, the resulting reentry state vector is accepted.

If the desired cotangent of the reentry flight path angle is not achieved, an iterative procedure is started. This procedure uses Conic Subroutines to compute the time of flight (Δt_{21}) to the desired reentry flight path angle.

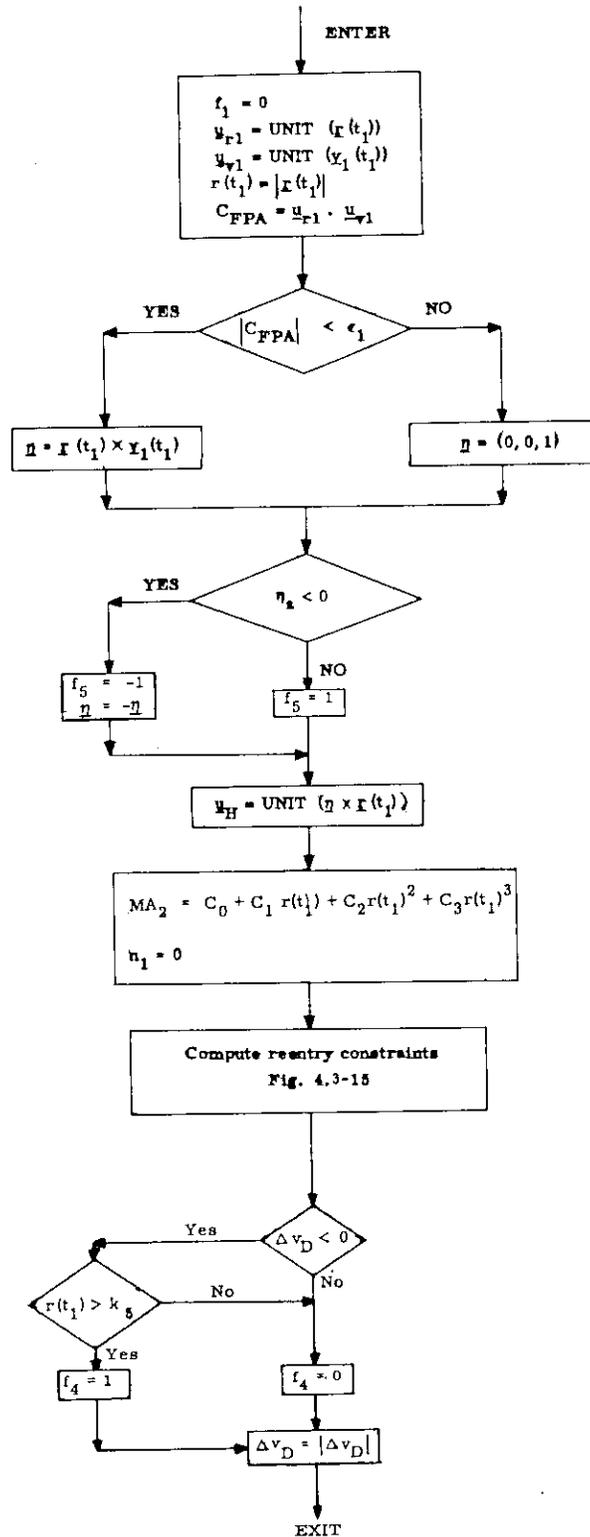


Figure 4.3-11 Precomputations

5.4-73

Revised COLOSSUS

Added GSOP #R-577 PCR # 690, 691

Rev. 5

Date 1-21-68

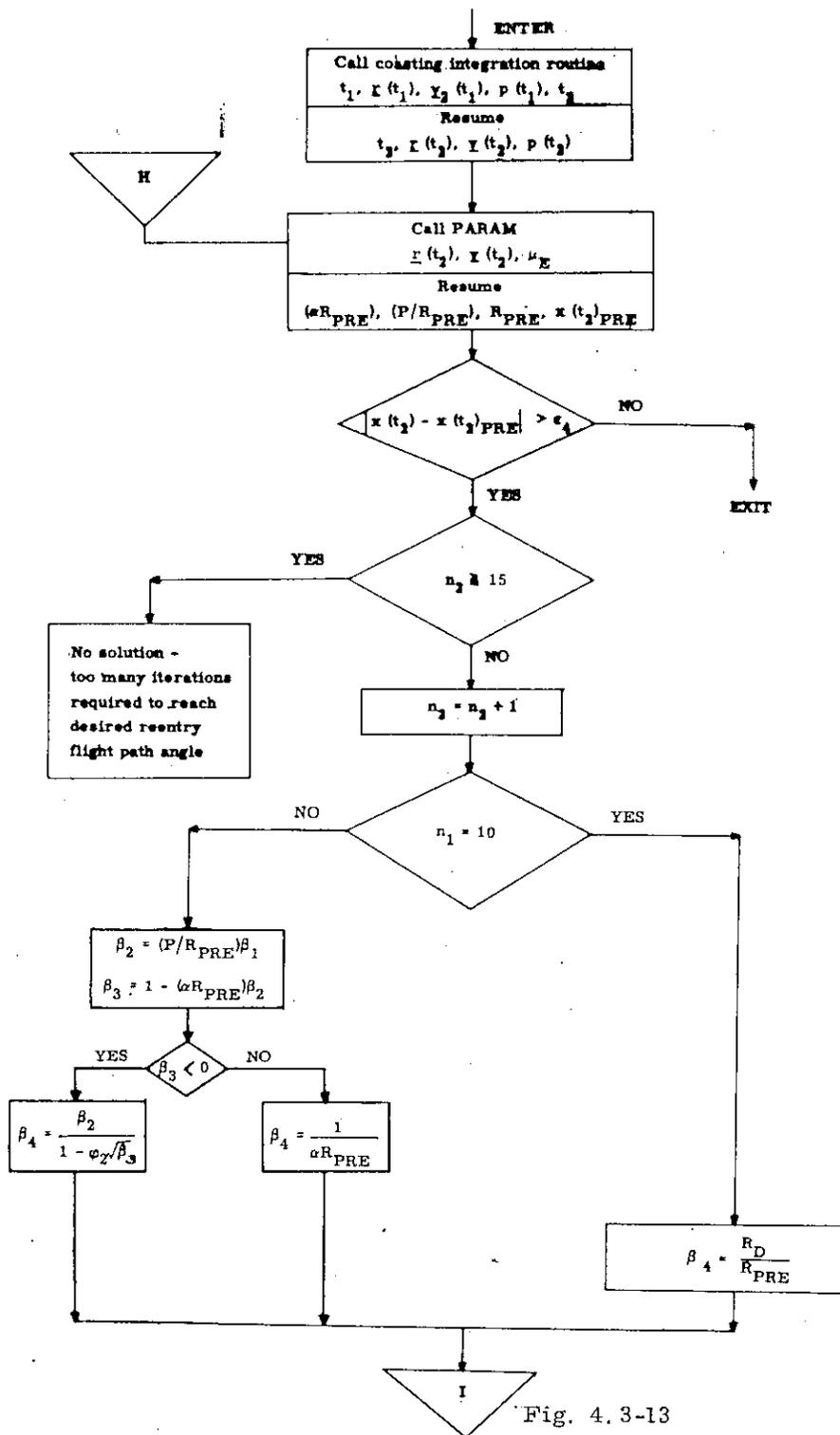


Fig. 4.3-13

Figure 4.3-12 Final State Vector Computation

5.4-74

Revised COLOSSUS

Added GSOP # R-577 PCR # 733

Rev. 5

Date 2-4-68

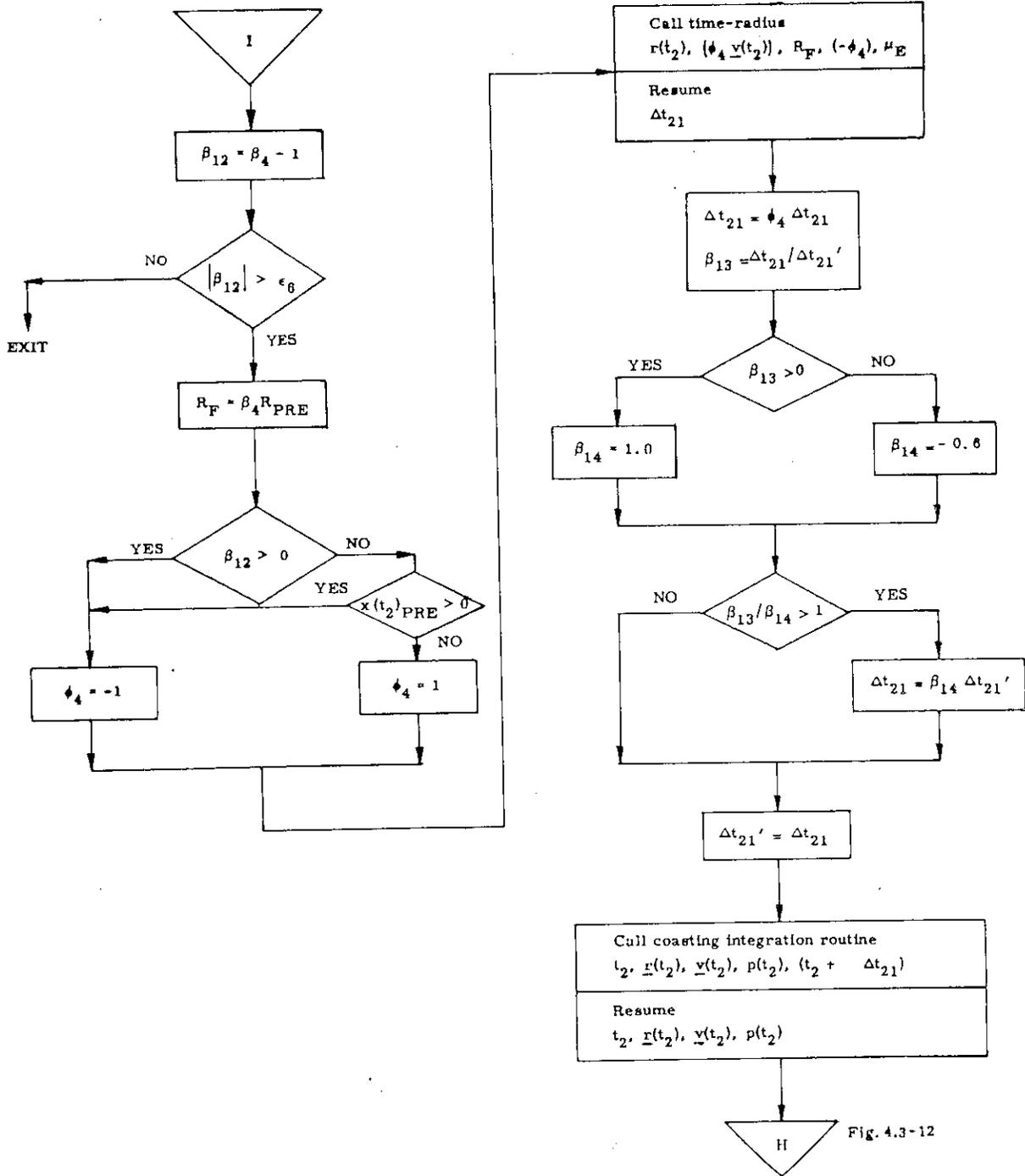


FIGURE 4.3-13 FINAL STATE VECTOR COMPUTATION

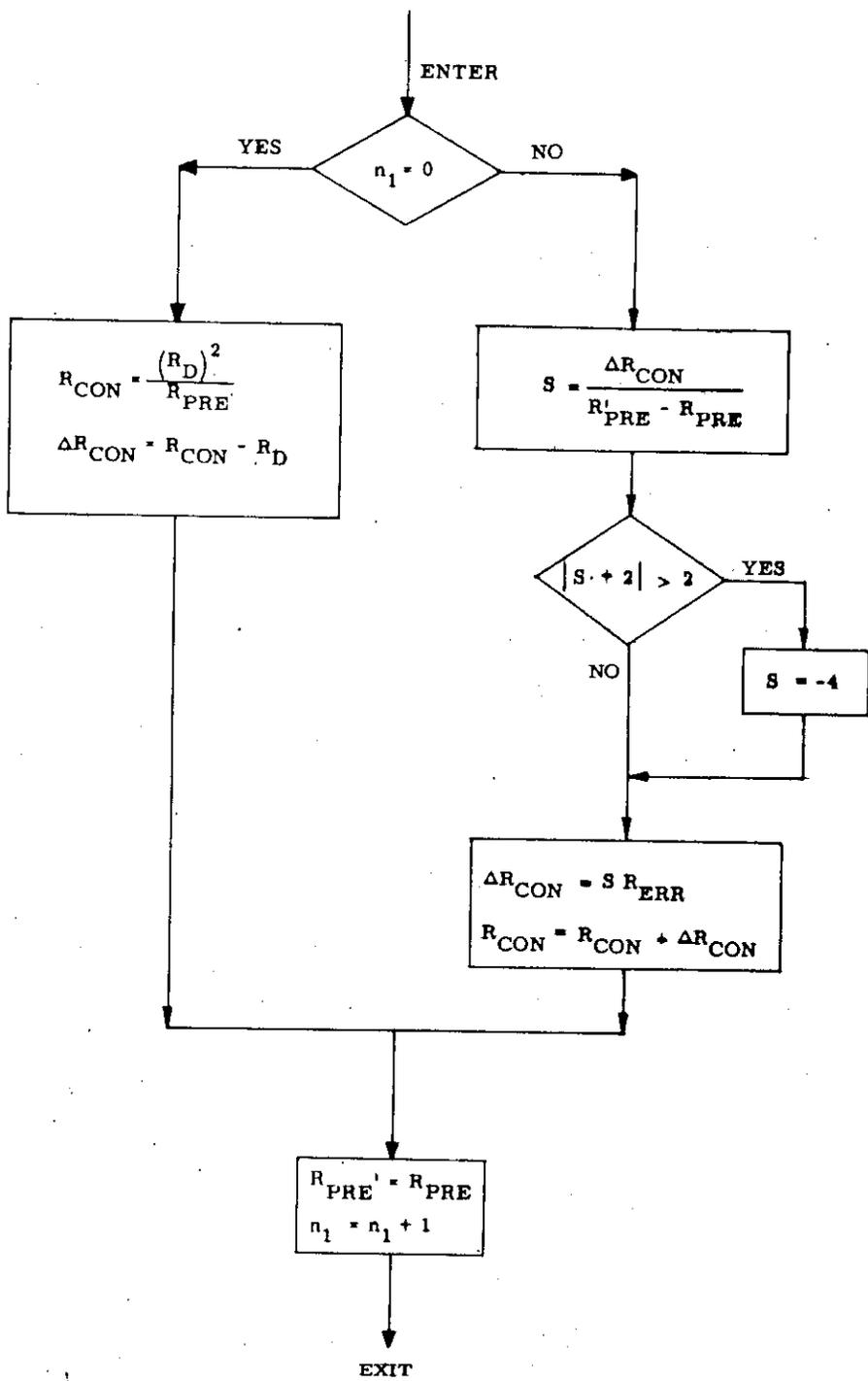


FIGURE 4.3-14 NEW R_{CON} COMPUTATION

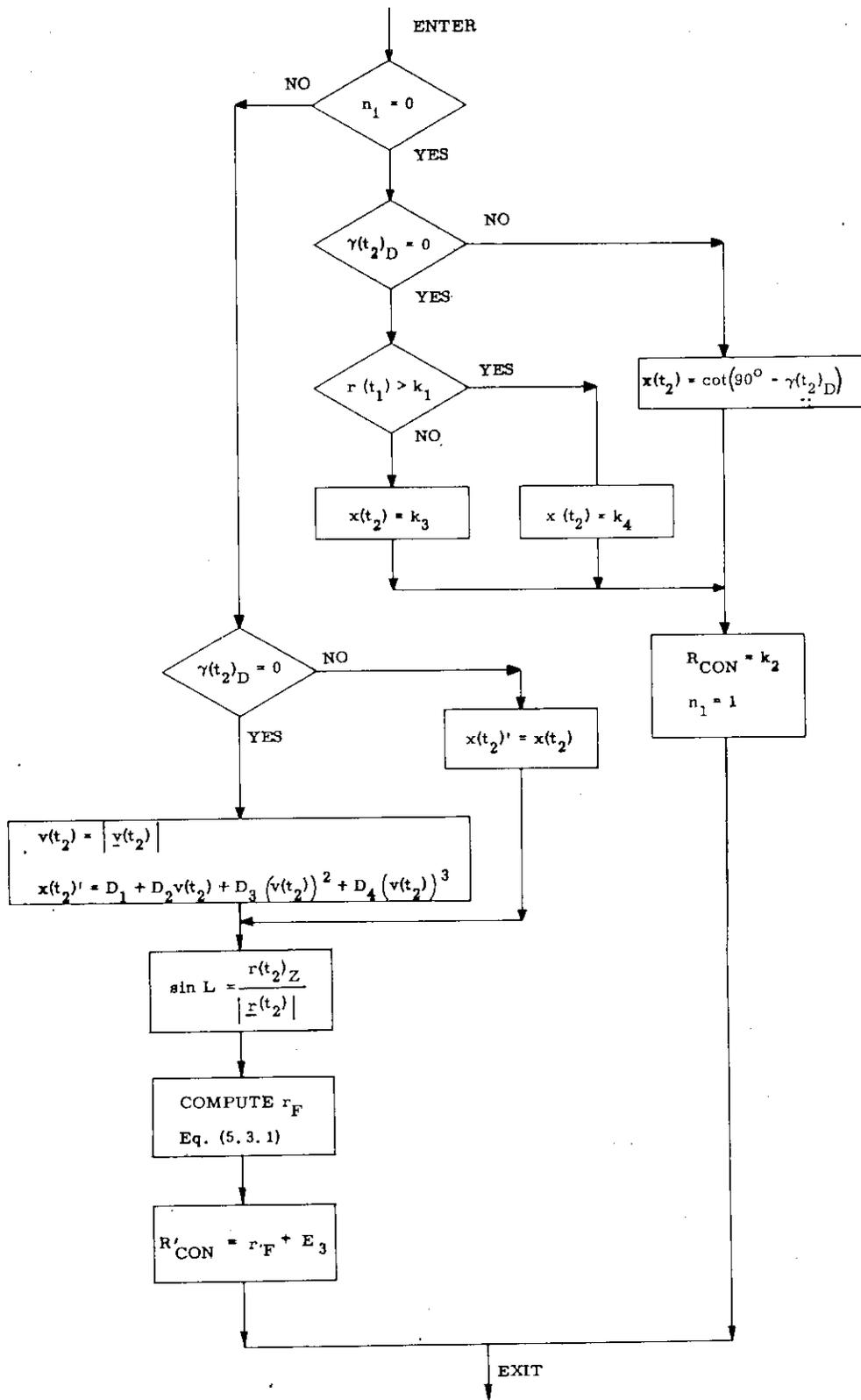


FIGURE 4.3-15 REENTRY CONSTRAINT COMPUTATION

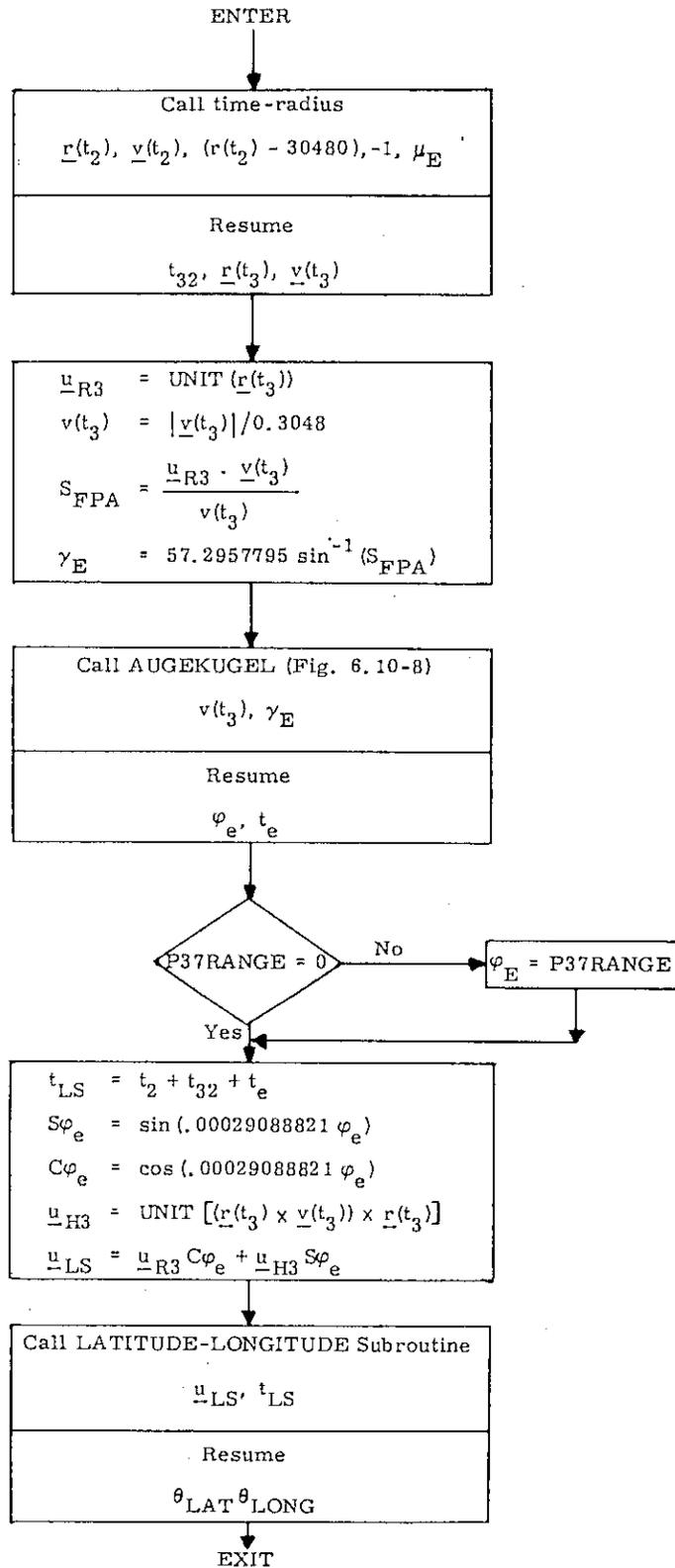


Figure 4.3-16 Landing Site Computation

5.4-78

Revised COLOSSUS

Added GSOP #R-577 PCR # 261 Rev. 5 Date 11-19-68

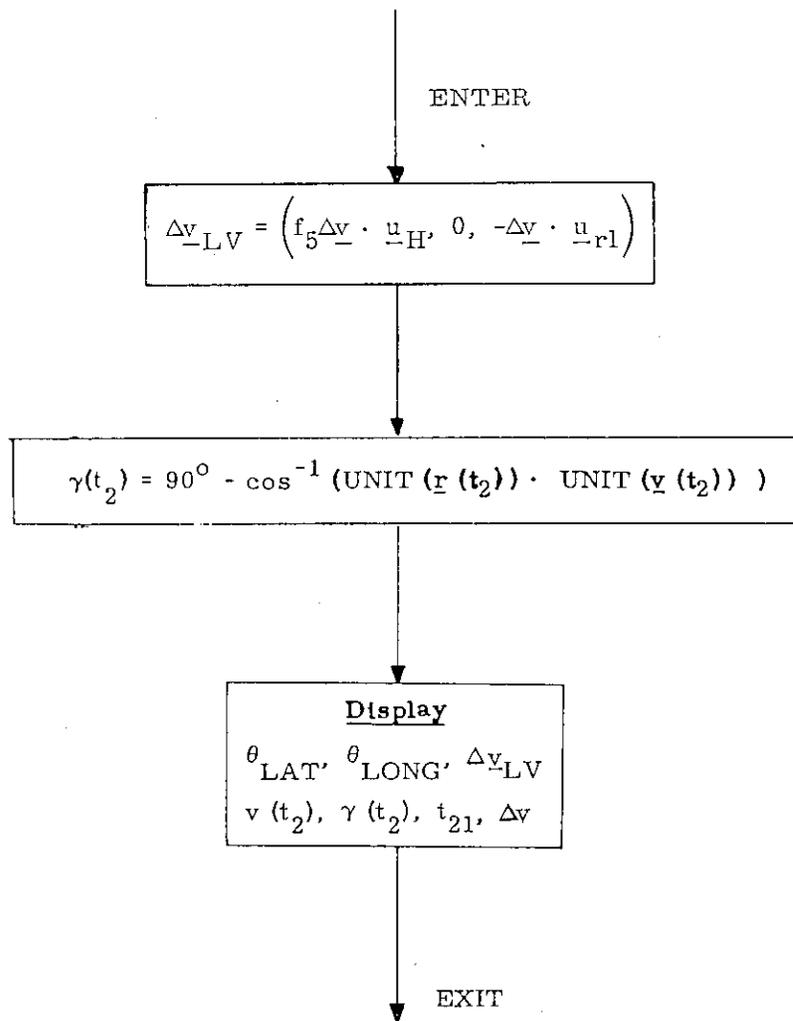


FIGURE 4.3-17 RETURN TO EARTH DISPLAY

5.4-79

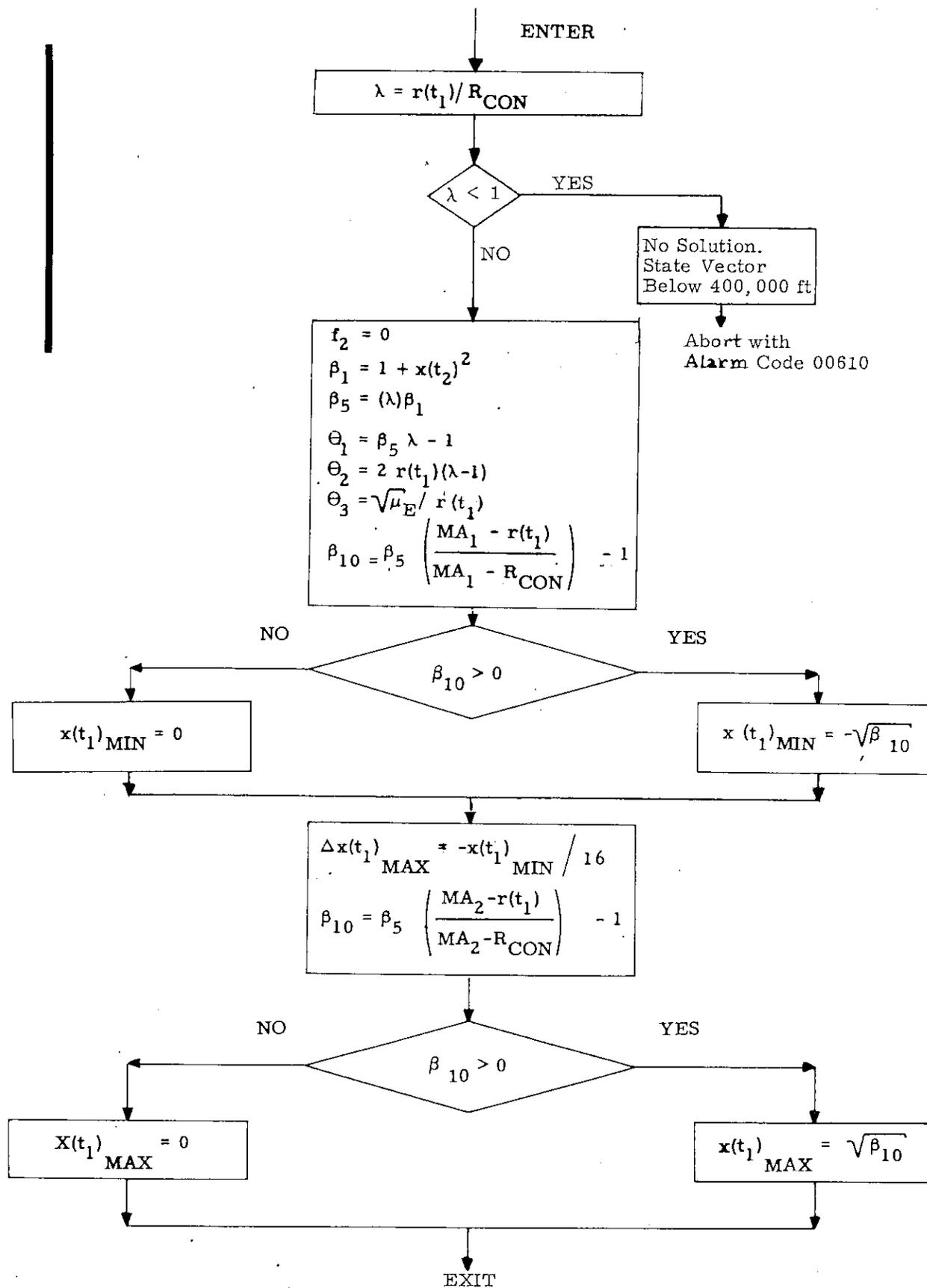


Figure 4.3-18 Conic Initialization

5.4-80

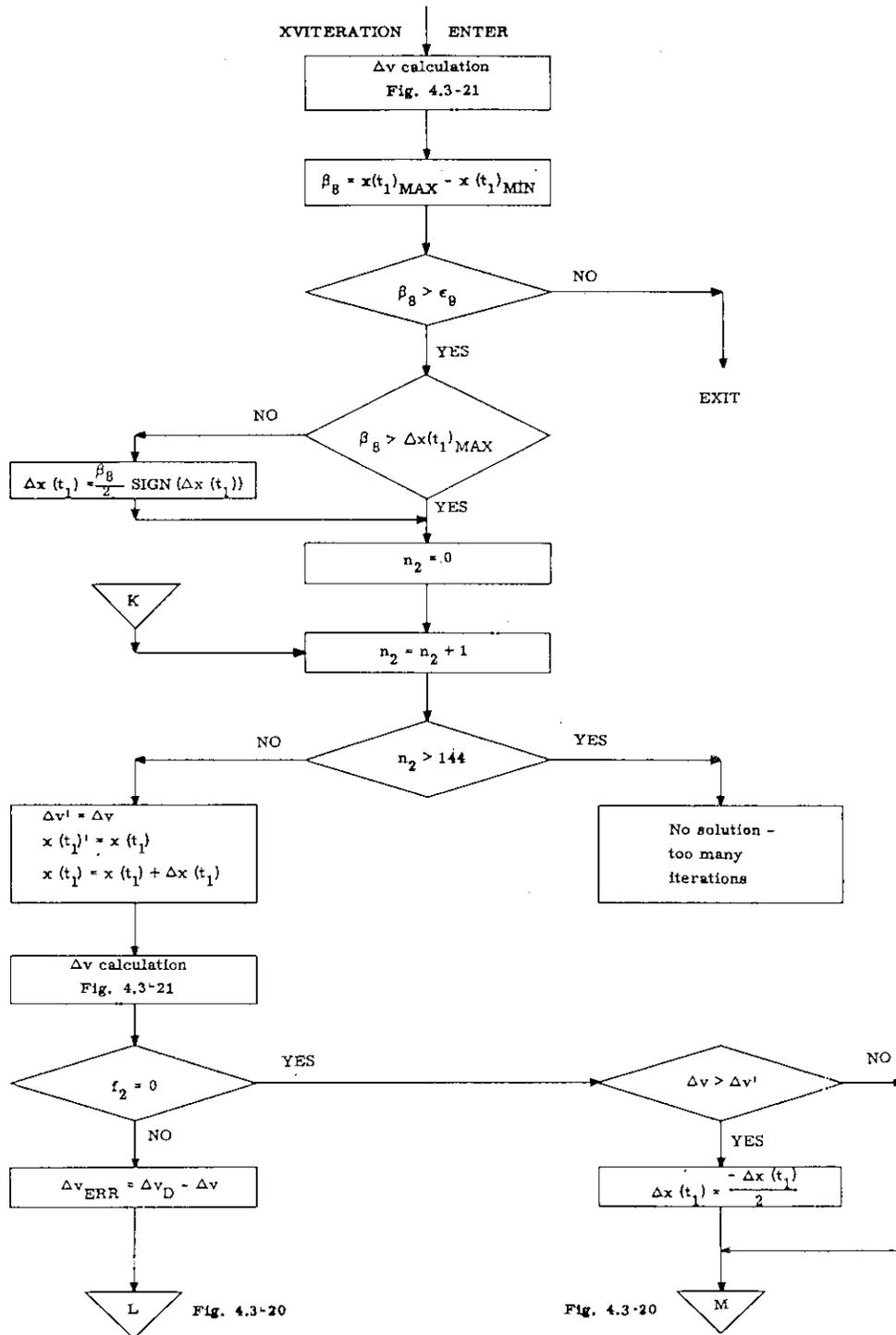


FIGURE 4.3-19 $x(t_1) - \Delta v$ ITERATOR

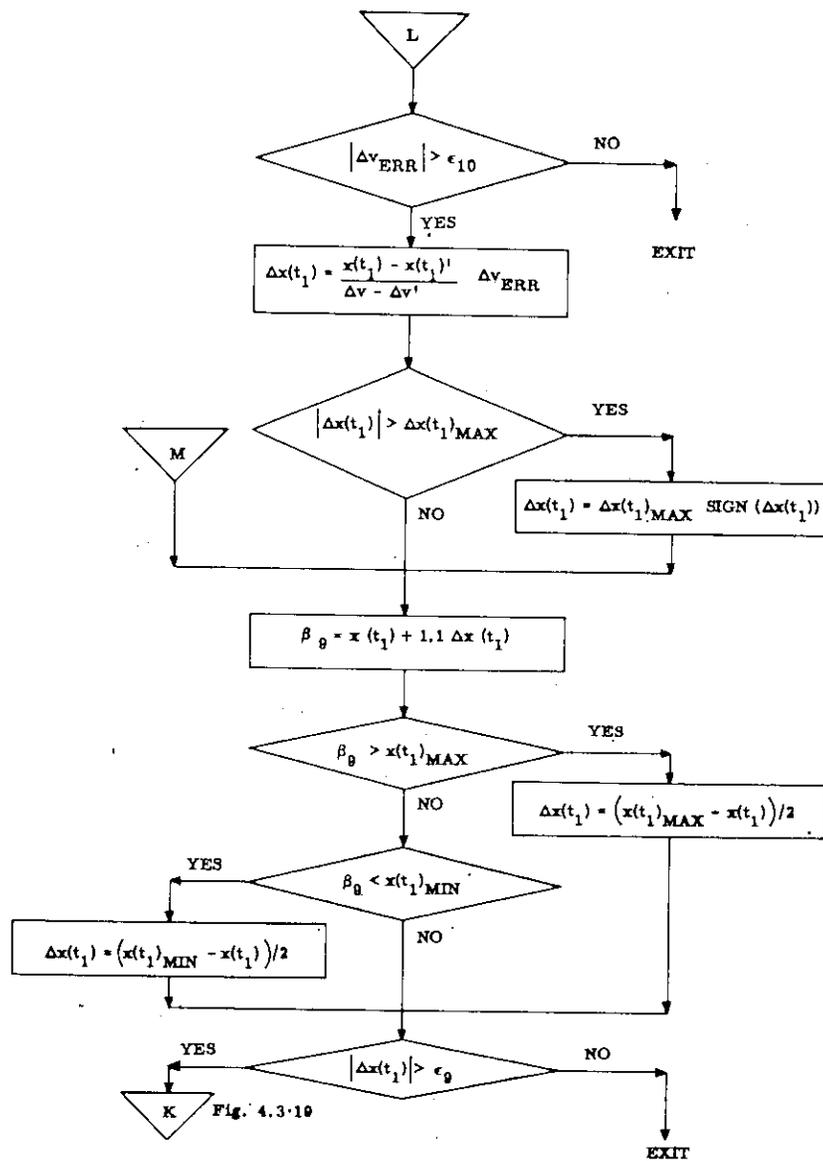


FIGURE 4.3-20 $x(t_1) - \Delta v$ ITERATOR

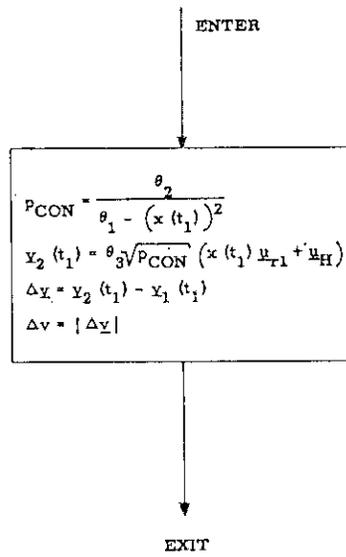


FIGURE 4.3-21 Δv CALCULATION

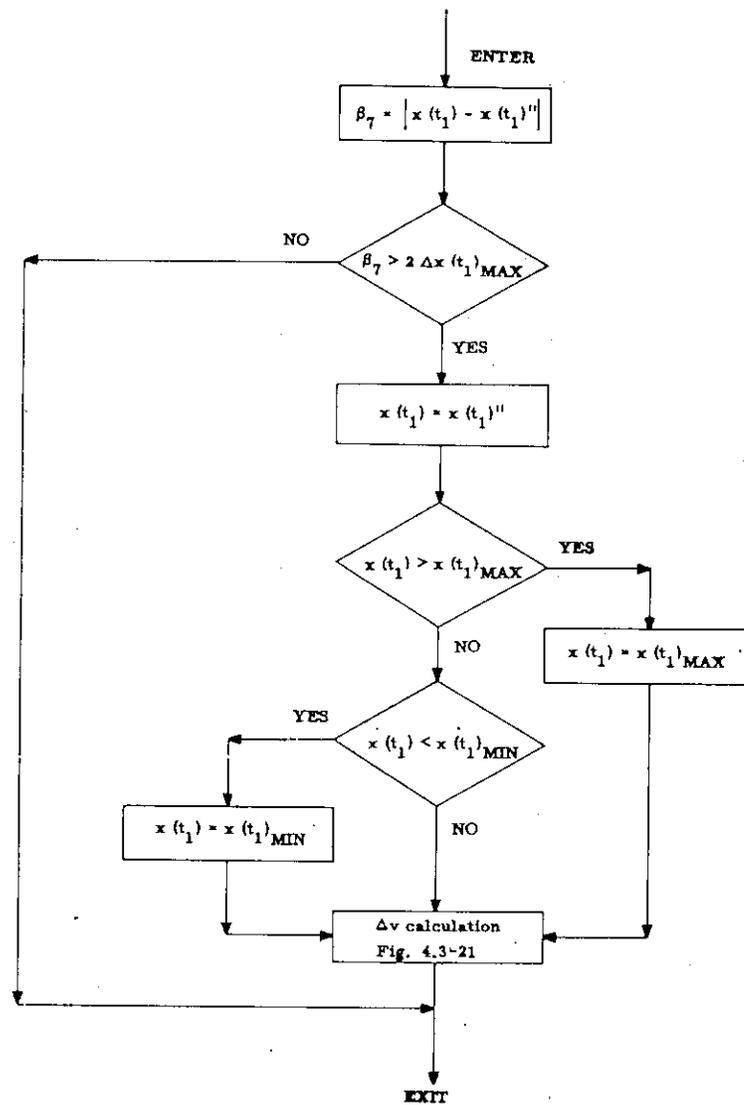


FIGURE 4.3-22 LIMIT $x(t_1)$ CHANGE

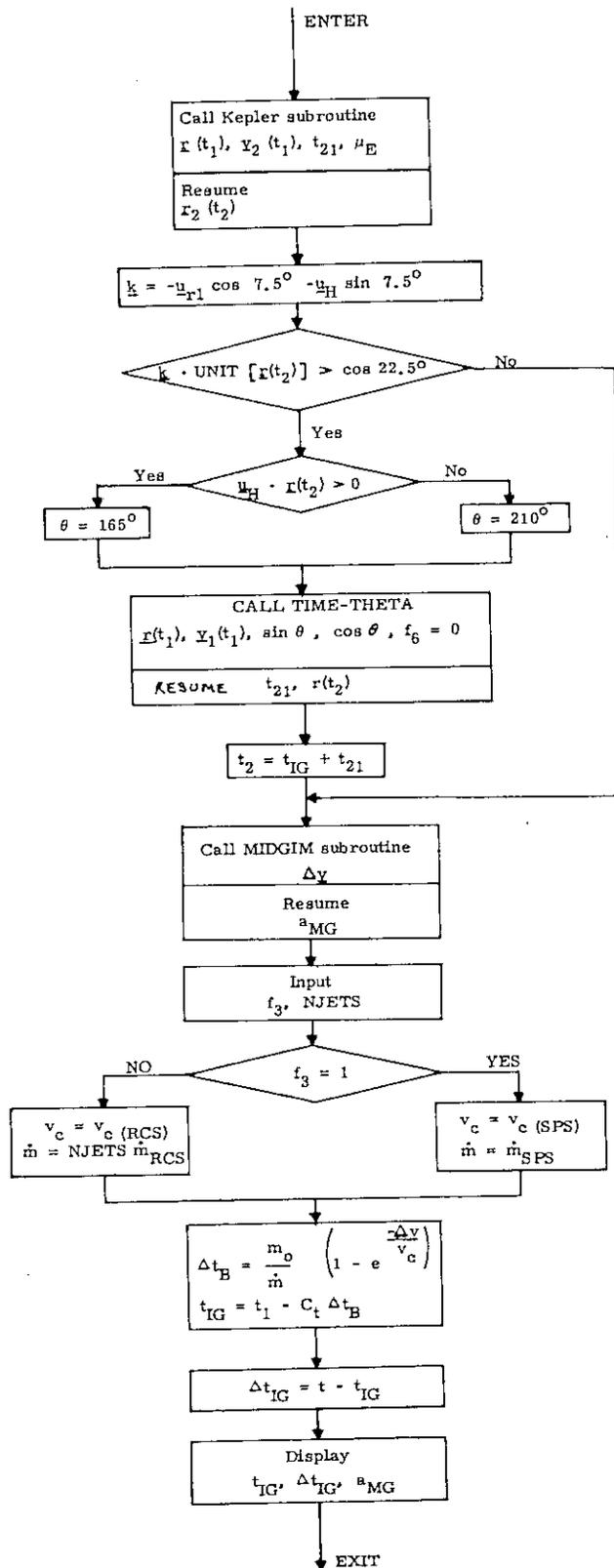


Figure 4.3-23 Target Computation

5.4-85

Rev. 5 - 3/69

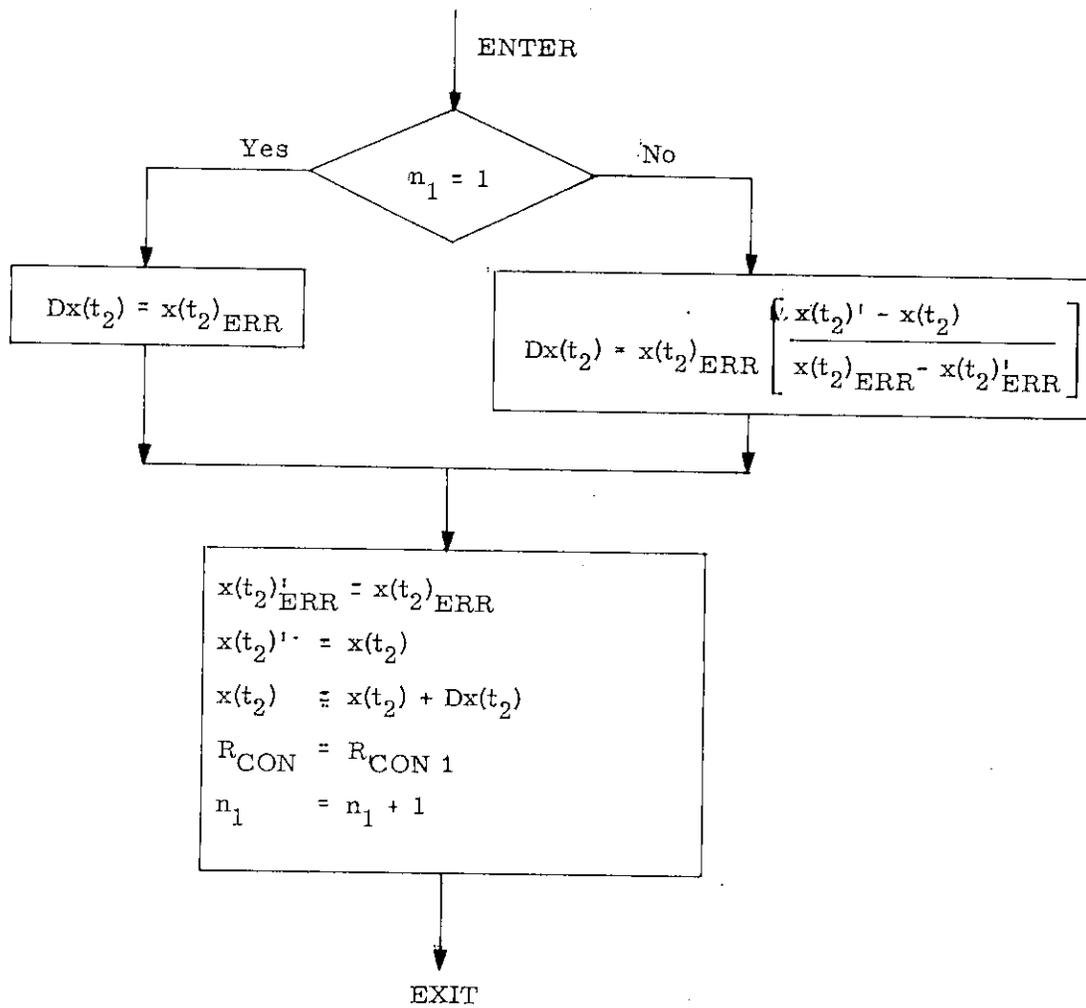


FIGURE 4.3-24 NEW $x(t_2)$ COMPUTATION