

5.3 POWERED FLIGHT NAVIGATION AND GUIDANCE

5.3.1 GENERAL COMMENTS

The objective of the powered flight guidance and navigation routines is to maintain an estimate of the LM state vector during thrusting maneuvers, and to control the thrust direction and duration such that the desired velocity cut-off conditions specified by the targeting routines of Section 5.4 are achieved. The powered flight navigation routine, used to maintain an estimate of the vehicle state vector during all thrusting conditions, is referred to as the Average-G Routine, and is presented in Section 5.3.2.

There are three basic powered flight guidance concepts used in the LGC for the Lunar Landing mission. The first is a velocity-to-be-gained concept with cross product steering (Section 5.3.3.4) that is used in each of the following two procedures:

1. Lambert Aim Point Maneuver Guidance (Section 5.3.3.5).
2. External ΔV Maneuver Guidance (Section 5.3.3.3.1).

These two procedures, based on the cross product steering concept, differ only in the unique generation of the desired velocity vector, \underline{v}_R .

The second basic guidance concept is the Quadratic Explicit Guidance of Section 5.3.4 which is used for the powered lunar landing maneuver. This concept is used to control the throttleable LM Descent Propulsion System (DPS) such that specified target or aim point conditions are achieved during the various phases of the landing maneuver. The landing maneuver target conditions for each phase are chosen to satisfy various DPS throttle and visibility constraints.

The third LGC guidance concept is the Linear Explicit Guidance of Section 5.3.5 used to control powered ascent maneuvers from the lunar surface to desired ascent injection conditions. This guidance concept is also used to control abort maneuvers initiated during the powered landing maneuver as described in Section 5.4.3.

5. 3. 2 POWERED FLIGHT NAVIGATION - AVERAGE-G
ROUTINE

The purpose of the Powered Flight Navigation Sub-routine is to compute the vehicle state vector during periods of powered flight steering. During such periods the effects of gravity and thrusting are taken into account. In order to achieve a short computation time the integration of the effects of gravity is achieved by simple averaging of the gravity acceleration vector. The effect of thrust acceleration is measured by the IMU Pulsed Integrating Pendulous Accelerometers (PIPA's) in the form of velocity increments (Δv) over the computation time interval (Δt). The computations are, therefore, in terms of discrete increments of velocity rather than instantaneous accelerations. The repetitive computation cycle time Δt is set at 2 seconds to maintain accuracy and to be compatible with the basic powered flight cycle.

The Average-G Routine, in contrast to the Coasting Integration Routine, is used when a short computing time is required such as during powered flight. The Average-G Routine computations are illustrated in Figs. 3.2-1 and 3.2-2. In these figures the definitions of the following parameters are:

- $\underline{r}(t)$ Vehicle position vector at time t .
- $\underline{v}(t)$ Vehicle velocity vector at time t .
- P_C Planet designator $\left\{ \begin{array}{l} 0 \text{ Earth} \\ 1 \text{ Moon} \end{array} \right.$
- Δt Computation cycle of 2 seconds.

$\Delta \underline{v}(\Delta t)$	The velocity vector sensed by the IMU PIPA's over the time interval Δt . This velocity vector increment is initially sensed in IMU or Stable Member Coordinates and then transformed to the Basic Reference Coordinate System when required.
$\underline{g}_p(t)$	Previous gravity acceleration vector at time t . This is a required initialization parameter and is supplied by the calling program.
\underline{u}_r	Unit vector in the direction of \underline{r} .
\underline{u}_z	Unit vector in the direction of the polar axis of the earth. = (0, 0, 1)
μ_E	Earth gravitational constant.
μ_M	Moon gravitational constant.
r_E	Equatorial radius of the earth.
J_{2E}	Second-harmonic coefficient of the earth's potential function.
$\underline{g}_b(t)$	Component of the earth gravity acceleration vector representing earth oblateness effects.

With reference to Fig. 3.2-2 it can be seen that a single oblateness term is included in the earth gravity subroutine computation, but none for the lunar case.

The PIPA measured velocity $\Delta \underline{v}$ is compensated for instrument errors as described in Section 5.6.13 prior to being transformed into the Basic Reference Coordinate System and processed in the Average-G Subroutine of Fig. 3.2-1.

It should be noted that in the LGC there are two modes to the Average-G Routine during lunar orbit phases. In the first mode the Average-G Routine computations are referenced to the Basic Reference Coordinate System. The second mode is used in the lunar landing programs P-63 to 66 and ascent and abort programs P-12, P-70 and P-71 in which the Average-G Routine computations are referenced to the Stable Member Coordinate System. In either mode the Average-G Routine output is finally referenced to the Basic Reference Coordinate System coordinates for telemetry down-link and extended verb computation purposes.

In addition, an up to date estimate of vehicle mass is made, similar to that used in descent (Section 5.3.4.3).

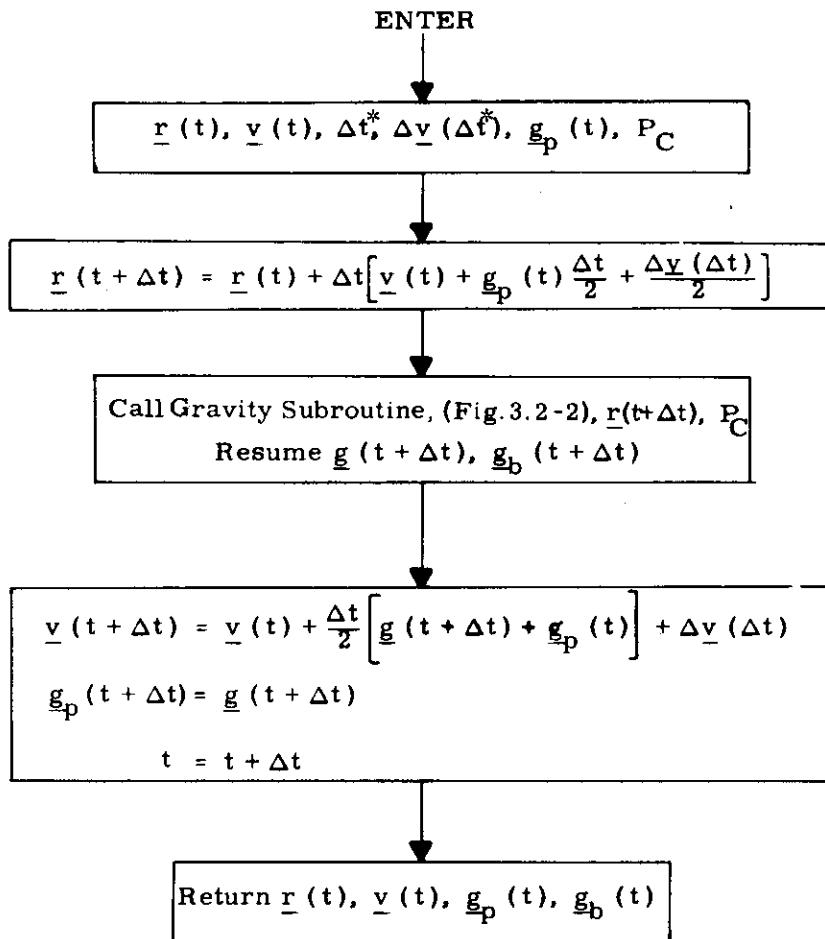


Figure 3.2-1 Average-G Subroutine

*This Δt is the actual computation cycle time rather than the nominal 2 sec.

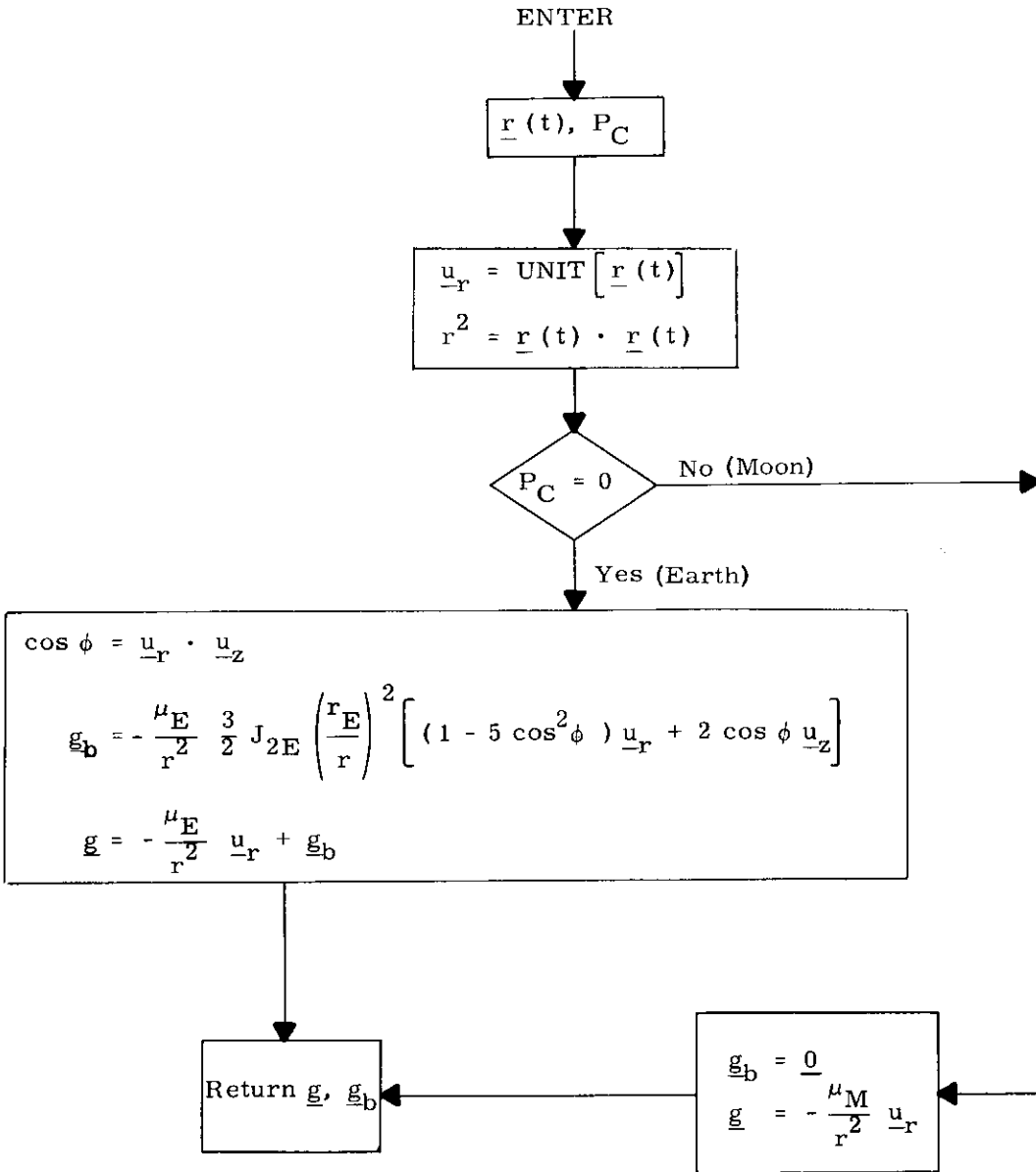


Figure 3.2-2 Gravity Subroutine

5.3.3 POWERED FLIGHT GUIDANCE USING CROSS PRODUCT STEERING

5.3.3.1 Introduction

Cross product steering is used to control the following maneuvers:

1. DOI (Descent Orbit Injection)
2. CSI (Coelliptic Sequence Initiation)
3. CDH (Constant Differential Altitude)
4. TPI (Transfer Phase Initiation, CFP and Stable Orbit Rendezvous)
5. TPM (Rendezvous Midcourse Corrections, CFP and Stable Orbit Rendezvous)
6. TEI (Transearch Injection SPS Backup)
7. External ΔV (RTCC or CMC Targeted Maneuver)

The DOI, CSI, CDH and External ΔV maneuvers are controlled by the External ΔV Guidance Mode (Section 5.3.3.3.1) in which the required cut-off velocity, v_R , is specified by either a pre-thrust targeting program (P-32 or P-33) or an external source (P-30). The TPI, TPM and TEI Backup maneuvers are controlled by a Lambert Aim Point Maneuver Guidance mode in which the v_R is computed each computing cycle by the ASTEER subroutine (for which the semi-major axis "a" is periodically recomputed by the LAMBERT subroutine) during the maneuver to establish the desired intercept trajectory. Both External ΔV and Lambert Aim Point guidance modes use the cross product steering concept in the LGC to control the thrust direction along the velocity-to-be-gained vector, and to terminate thrust when the desired velocity increment has been achieved.

Four subroutines are used repetitively in sequence (Section 5.3.3.2) during cross product controlled maneuvers to accomplish this function. These are:

1. The Powered Flight Navigation Average-G Routine which computes the state vector accounting for the effects of thrust acceleration and gravity.
2. The Cross-Product Steering Subroutine which has 3 functions:
 - a) generation of steering commands to the vehicle autopilot.
 - b) computation of time-to-go before engine shut-off and the issuance of engine-off commands.
 - c) incremental updating of the velocity-to-be-gained vector, (in the external ΔV mode only).
3. The Velocity-to-be-Gained Subroutine which repetitively solves the Lambert intercept problem when in the Lambert Aim Point guidance mode.
4. The ΔV Monitor Subroutine which detects engine ignition and unexpected thrust termination.

The Average-G Routine is described in Section 5.3.2. The other subroutines listed above are described in Sections 5.3.3.4 to 5.3.3.6. The Pre-Thrust Subroutines of Section 5.3.3.3 initialize the powered maneuver programs for either the External ΔV or Lambert Aim Point guidance modes, and for the selected engine for the maneuver.

5.3-9

Revised LUMINARY
 Added GSOP # 567 PCR # 647 Rev. 7 Date 11/69

The LGC powered flight programs which use cross product steering are:

P-40 Descent Propulsion System (DPS)

P-41 Reaction Control System (RCS)

P-42 Ascent Propulsion System (APS)

Active steering and engine-off commands are provided by programs P-40 and P-42. Maneuvers using RCS translation control (P-41) are manually controlled and terminated by the astronaut while the LGC displays the required velocity-to-be-gained in spacecraft coordinates.

The functions of the External ΔV Pre-Thrust Program, P-30, are described in the pre-thrust subroutine description of Section 5.3.3.3.1.

5.3-10

Revised LUMINARY
 Added GSOP # 567 PCR # 823 Rev. 7 Date 11/69

5.3.3.2 Powered Flight Guidance Computation Sequencing

The time sequencing of the powered flight sub-routines for External ΔV steering is shown in Figs. 3.3-1 thru 3.3-3 and for Lambert Aim Point steering in Figs. 3.3-4 thru 3.3-6.

Figures 3.3-1 and 3.3-4 show the sequencing during the ignition countdown which starts approximately 30 secs. before the nominal ignition time. Figures 3.3-2 and 3.3-5 show the normal sequencing for an engine-on time greater than 6 seconds as predicted by the pre-thrust subroutines. Figures 3.3-3 and 3.3-6 illustrate the sequencing during engine thrust termination.

The basic computation cycle time of the steering is 2 seconds and, as shown on the above figures, is initiated by the reading of the PIPA ΔV registers. The various sub-routines utilized during the 2 second cycle are sequenced in time as shown.

5.3-11

Revised
 Added

LUMINARY

GSOP # 567

PCR # 647

Rev. 7

Date 11/69

If the calculations conducted prior to the thrusting period (pre-thrust) indicate that the maneuver objective can be reached with a 6 second thrust period or less, provision is included (switches S_I and s_W) to preclude the generation of steering commands and engine commands during the thrusting period. In this case at the time of ignition the engine-off signal is set to be issued at the end of the estimated thrust period.

In addition to timing information, the sequence diagram of Figs. 3.3-1 to 3.3-6 also show the basic information utilized by each subroutine and its source.

The guidance computer program which controls the various subroutines to create a powered flight sequence is called the Servicer Routine. The sequence diagrams of Figs. 3.3-1 to 3.3-6 define what the Servicer Routine does, but do not show the logic details of how these functions are accomplished.

The subroutines listed in Figs. 3.3-1 to 3.3-6 are described in Section 5.3.2 and the following Sections 5.3.3.3 to 5.3.3.6. These sections should be referenced in tracing the powered flight computation sequencing.

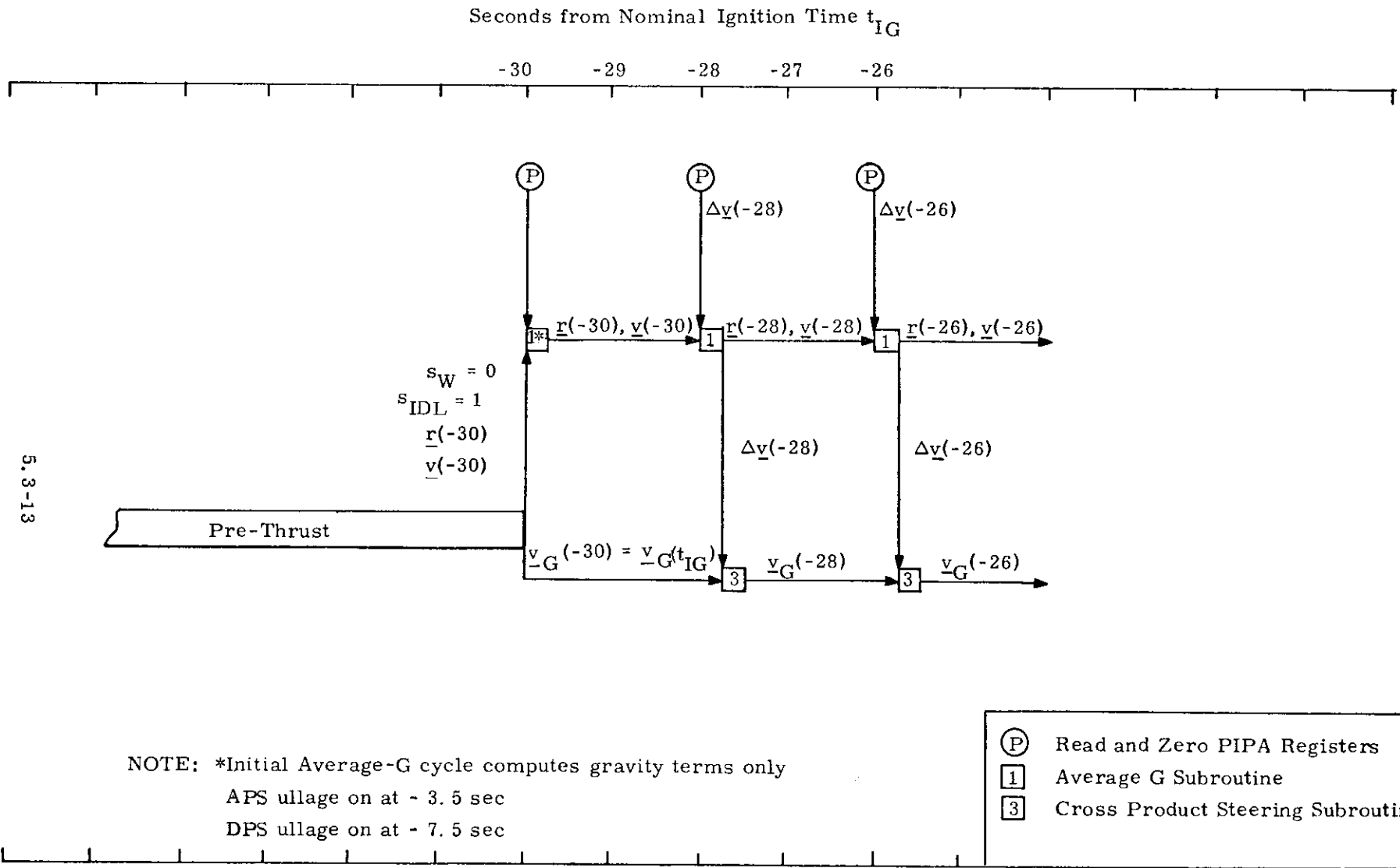


Fig. 3.3-1 Ignition Countdown - External ΔV Subroutine Sequencing for APS and DPS Maneuvers

5.3-14

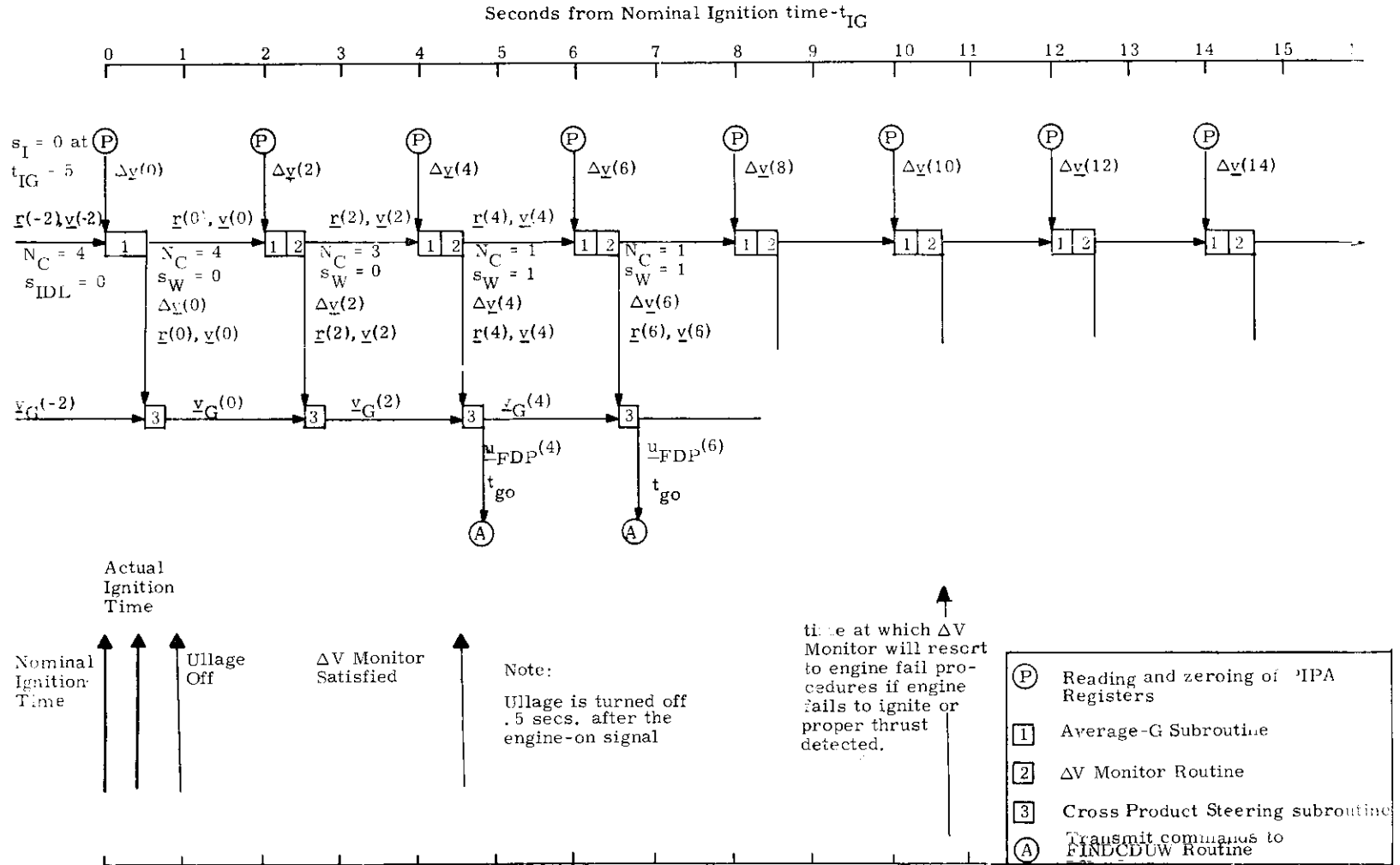


Fig. 3.3-2 Normal External ΔV Subroutine Sequencing (Maneuver time greater than 6 sec) for APS and DPS

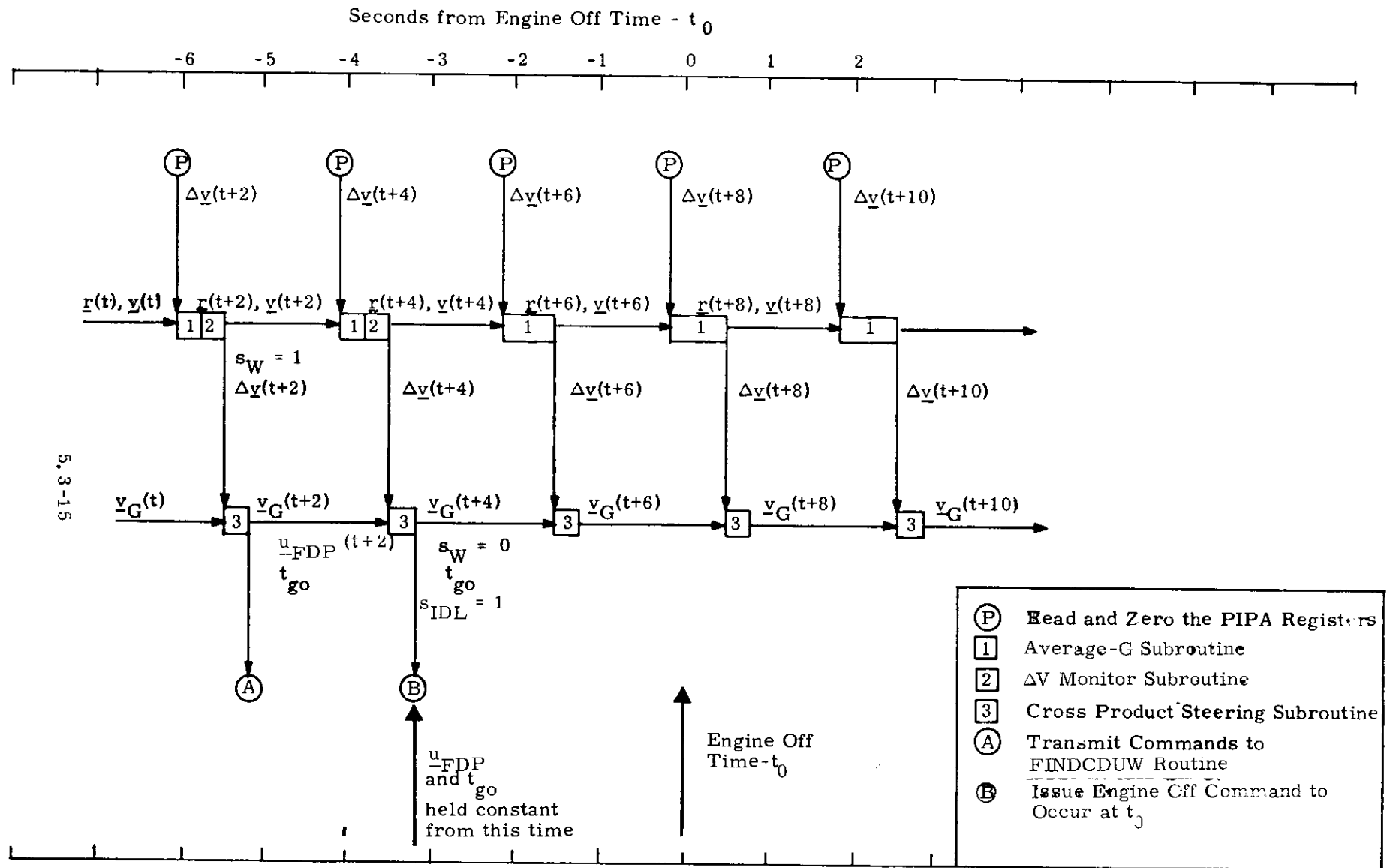


Fig. 3.33 Engine Off External ΔV Subroutine Sequencing for APS and DPS Maneuvers

Revised LUMINARY
 Added GSOP # 567
 PCR # 647
 Rev. 7
 Date 11/69

5.3-16

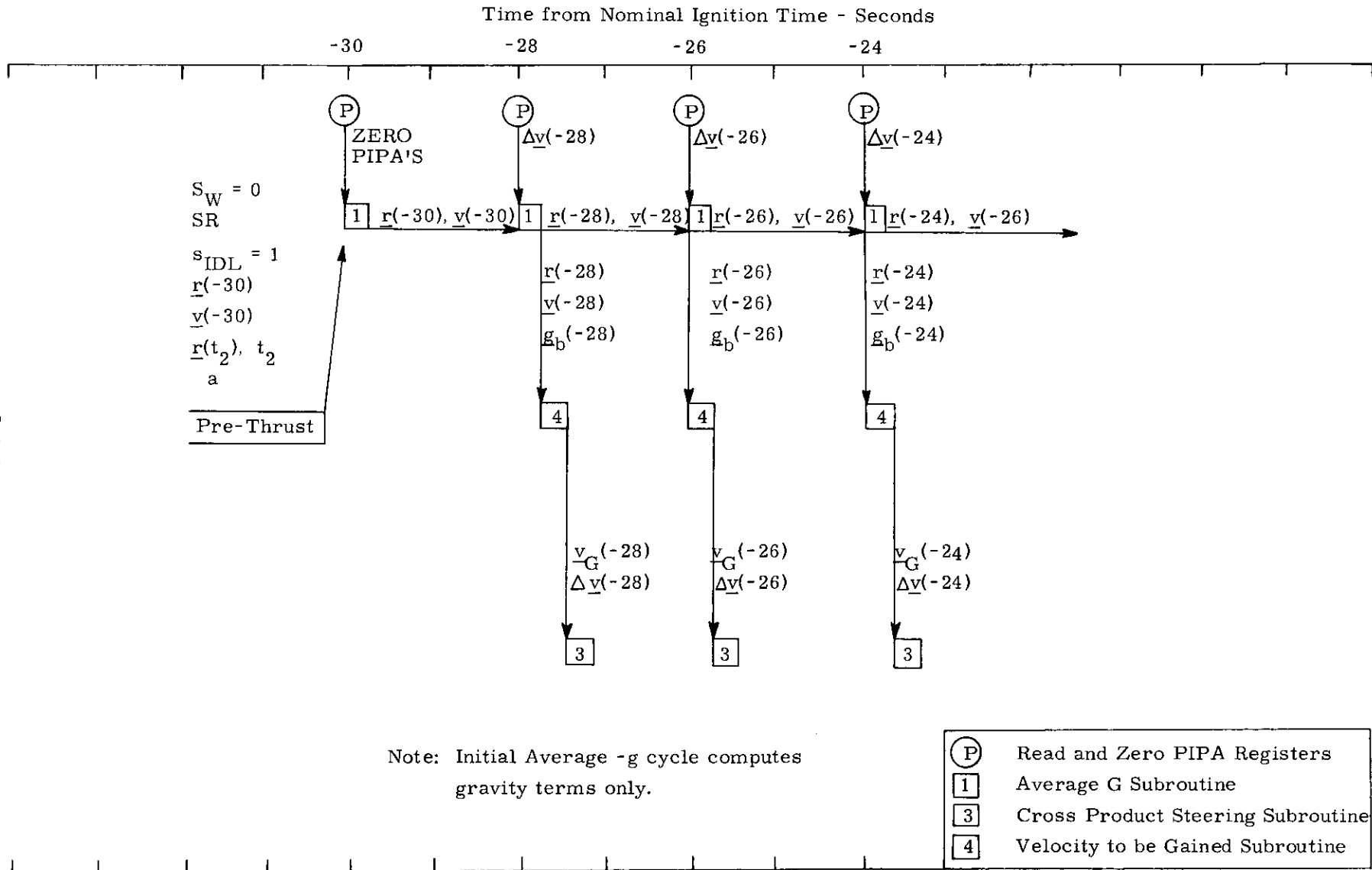


Fig. 3.3-4 Ignition Countdown - Lambert Aim Point Subroutine Sequencing for APS & DPS Maneuvers

Revised
 Added

LUMINARY
 GSOP # 567

PCR # 647

Rev. 7

Date 11/69

5.3-17

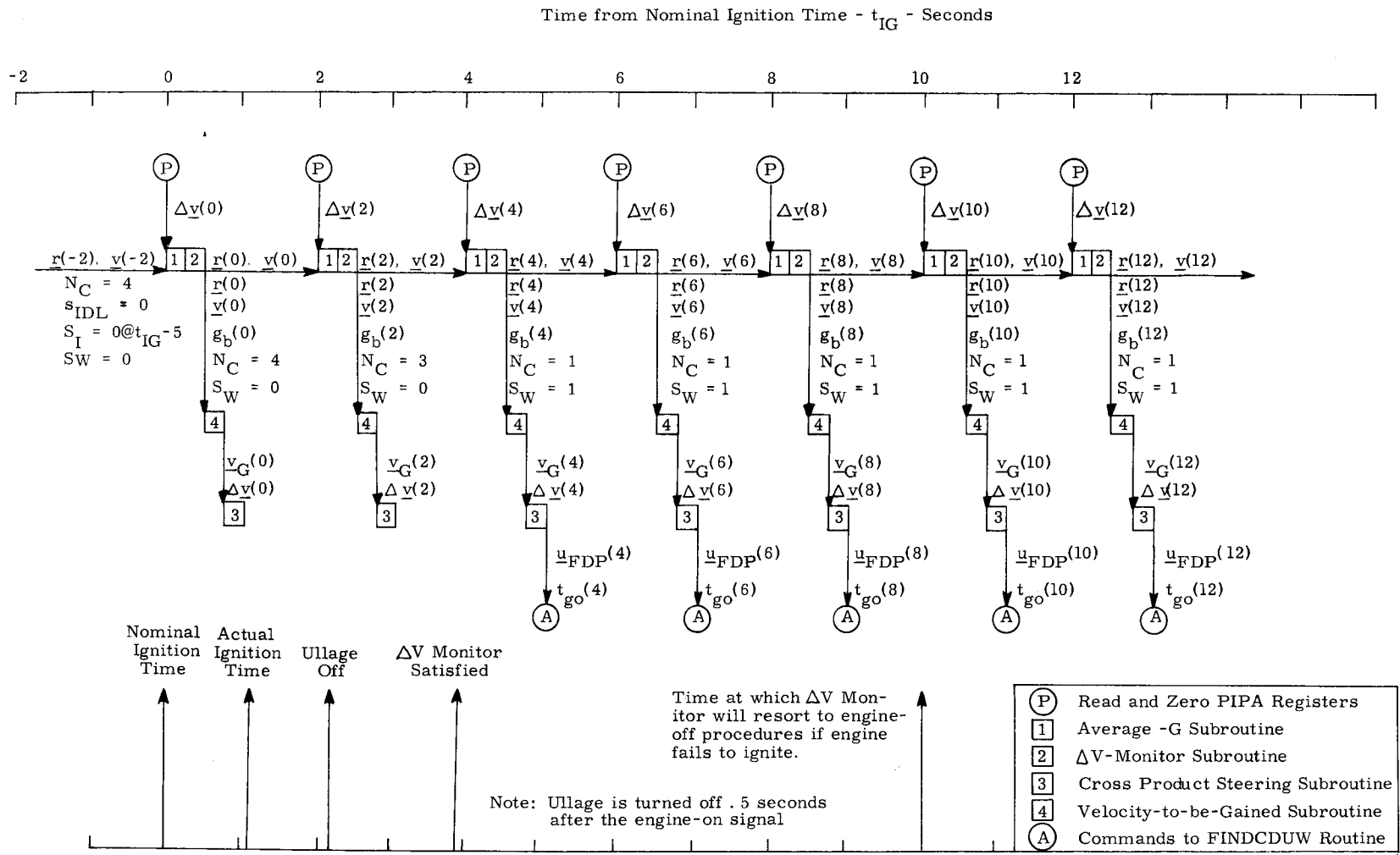


Fig. 3.3-5 Normal Lambert Aim Point Subroutine Sequencing (Maneuver time greater than 6 sec) for APS and DPS

Revised
 Added

LUMINARY
 GSOP # 567

PCR # 647

Rev. 7

Date 11/69

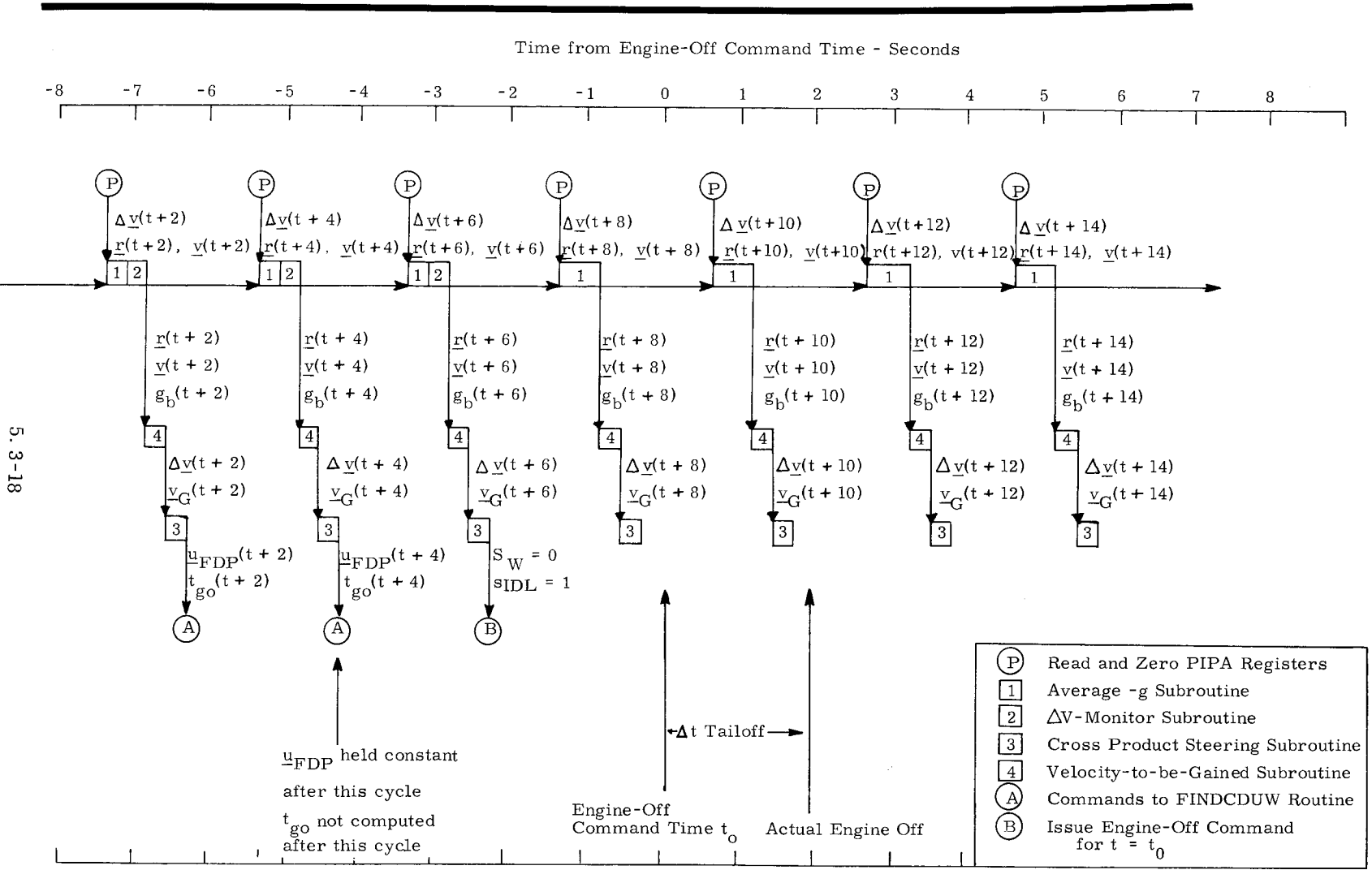


Fig. 3.3-6 Engine-Off Lambert Aim Point Subroutine Sequencing

5.3.3.3 Pre-Thrust Computations

The objectives of the computations required prior to thrusting maneuvers are to determine the following factors:

- 1) The desired thrust direction at ignition.
- 2) The estimated duration of the powered maneuver to determine if there will be enough time to allow active steering, and in the case of the DPS if the throttle can be advanced to the full throttle position (FTP) after the 10% thrust start sequence.
- 3) Whether an IMU realignment is required to avoid gimbal lock.
- 4) Various parameters and variables required by subsequent powered flight routines.

The two major guidance modes using cross product steering are the External ΔV guidance mode and the Lambert Aim Point Maneuver guidance mode. The pre-thrust computations required for the External ΔV mode are presented in Section 5.3.3.3.1. Those required for the Lambert Aim Point Maneuver mode are described in Section 5.3.3.5.1. The initial IMU alignment computations and maneuver time logic is summarized in Section 5.3.3.3.3.

5.3.3.3.1 External ΔV Maneuver Pre-Thrust Computations

External ΔV maneuver guidance is normally used to control the CSI and CDH maneuvers of the concentric flight plan rendezvous profile, or an externally targeted maneuver in which a

constant thrust attitude is desired. The guidance program accepts input data via the DSKY (P-30) or the telemetry uplink (P-27) in the form of 3 components of an impulsive $\Delta \underline{V}_{LV}$ expressed in a local vertical coordinate system of the active vehicle at the ignition time t_{IG} . An approximate compensation for the finite maneuver time is made within the program by rotating the $\Delta \underline{V}_{LV}$ vector, and the guidance program issues commands to the spacecraft control system so as to apply the compensated velocity increment along an inertially fixed direction. The active vehicle state vector is normally either available or can be extrapolated to the ignition time in the LGC. If External ΔV guidance is used in cislunar space for midcourse corrections, the state vector must be uplinked for the ignition time along with the 3 components of the desired impulsive $\Delta \underline{V}_{LV}$.

The pre thrust computations required for the External ΔV guidance mode are shown in Fig. 3.3-7. The following parameter definitions refer to this figure.

$\Delta \underline{V}_{LV}$ Specified velocity change in the local vertical coordinate system of the active vehicle at the time of ignition. This is an input parameter.

$$\Delta \underline{V}_{LV} = \begin{pmatrix} \Delta V_X \\ \Delta V_Y \\ \Delta V_Z \end{pmatrix}$$

t_{IG} Ignition time, an input parameter.

$\Delta \underline{V}_P$ The inplane velocity components of $\Delta \underline{V}_{LV}$ in the Basic Reference Coordinate System.

θ_T The approximate central angle traveled during the maneuver.

m Vehicle mass.

$\Delta \underline{V}$ Specified velocity change in Basic Reference Coordinates.

MGA Angle equivalent to the IMU middle gimbal angle when the vehicle X axis is aligned along $\Delta \underline{V}$. This angle is used to check for gimbal lock tolerance.

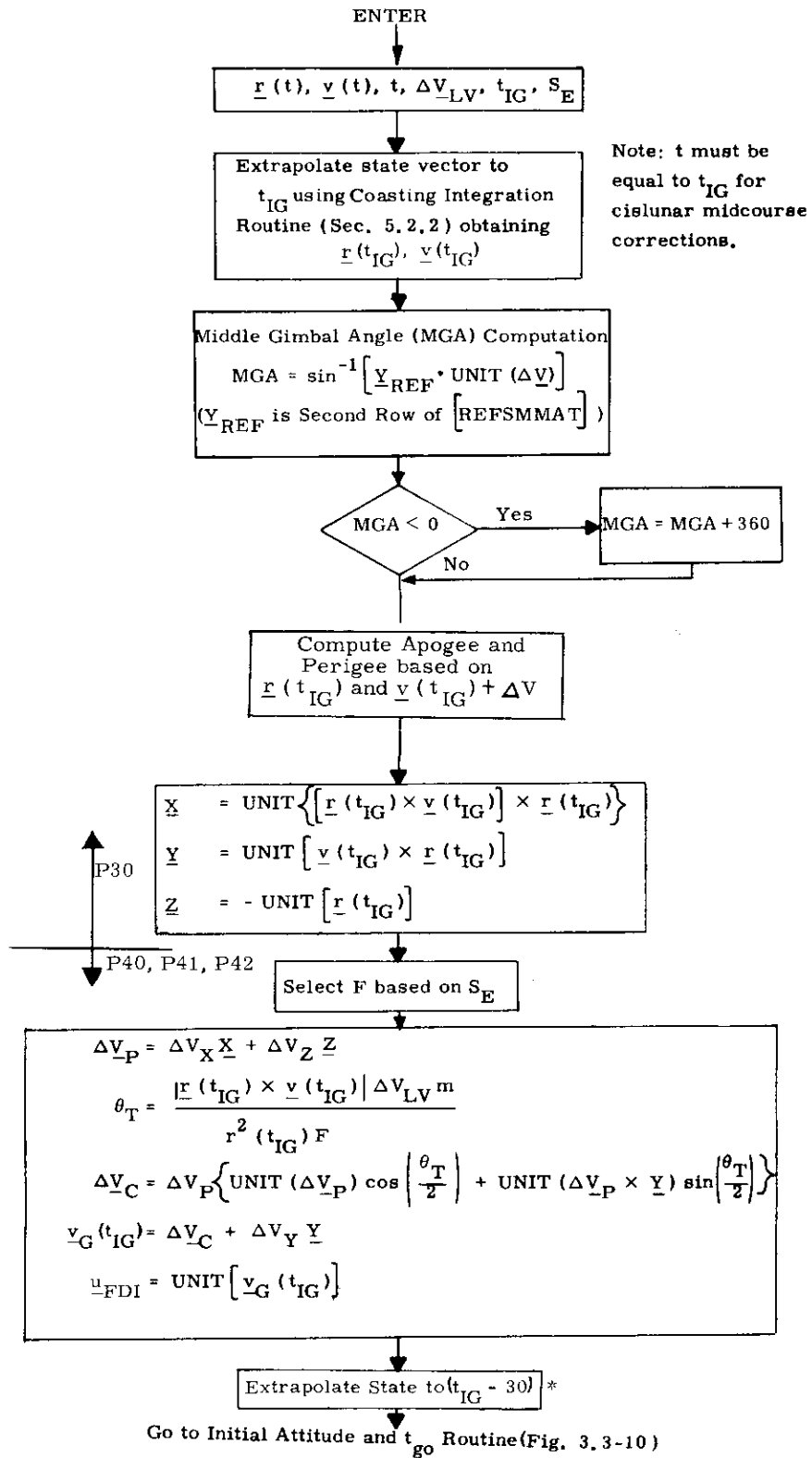


Fig. 3.3-7 External ΔV Prethrust Routine

* See Section 5.3.3.3.4 for ignition delay procedure

Revised
 Added

LUMINARY
 GSOP # 567

5.3-21

PCR # 647

Rev. 7 Date 11/69

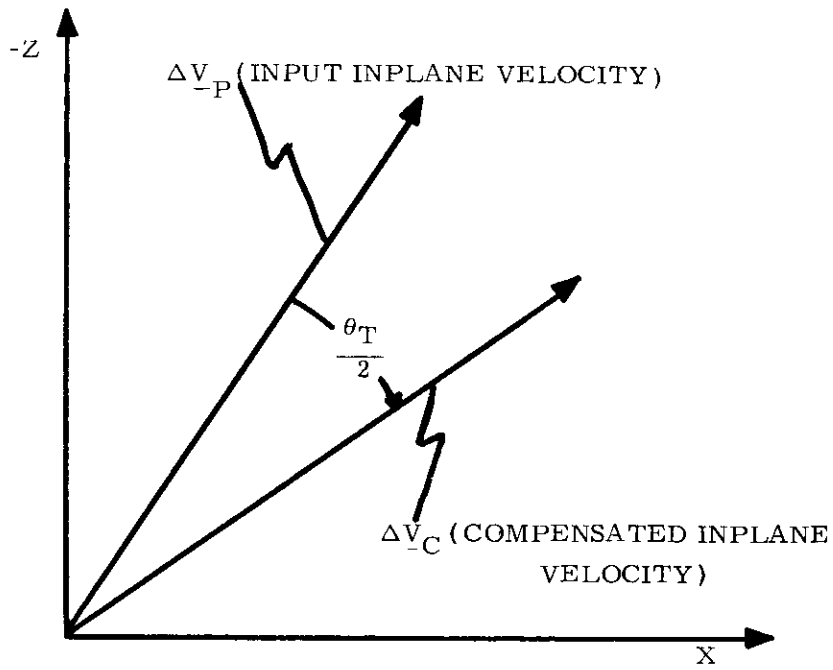


Figure 3.3-8 Inplane External ΔV Maneuver Angle Compensation

S_E Engine Select Switch $\left\{ \begin{array}{l} -1 \text{ P-40 (DPS)} \\ 0 \text{ P-41 (RCS)} \\ 1 \text{ P-42 (APS)} \end{array} \right.$

F Prestored nominal thrust based on the engine selection switch S_E
 F_{DPS} for the DPS (P-40)
 F_{APS} for the APS (P-42)
 F_{RCS} for the RCS (P-41)

The values of the various engine thrust levels are listed in Sections 5, 8 and 6. It should be noted that the DPS thrust level, F_{DPS} , is for the full throttle position (FTP) and the 26 second 10% thrust start sequence is not accounted for in the predicted maneuver time for the External ΔV guidance mode. In the case of P-41, RCS, the astronaut can select either a 2 or 4 jet translation maneuver.

$\Delta \underline{V}_C$ The compensated inplane velocity-to-be-gained vector.

$\underline{v}_G(t_{IG})$ Total velocity to-be-gained at t_{IG} .

\underline{u}_{FDI} Unit vector in the desired initial thrust direction in the Basic Reference Coordinate System.

The inplane External ΔV maneuver angle compensation involving $\Delta \underline{V}_P$ and $\Delta \underline{V}_C$ is illustrated in Fig. 3.3-8.

5.3.3.3.2 Lambert Aimpoint Maneuver Pre-Thrust Computations:

As indicated in Fig. 3.3-4, the prethrust calculations of P40, P41 or P42 preceding $t_{IG}-30$ for a Lambert Maneuver provide a value of semi-major axis "a" for use by the ASTEER subroutine. Two cases of the generation of this parameter are pertinent: 1) when the central angle between the position vector at t_{IG} and the offset target vector (from a targeting program) is greater than 165° and 2) where this angle is less than 165° .

In the former case the Lambert subroutine is called with the cone angle set to 45° (SR = 1) and is recalled as frequently as possible to avoid computational difficulties associated with the 180° transfer case. In the latter case, the Lambert subroutine is called once with the cone angle set to 10° and is called subsequently to update "a" only on the basis of time elapsed since the last computation of "a" (an erasable quantity UT).

5.3-24

Revised LUMINARY
 Added GSOP # 567 PCR # 647 Rev. 7 Date 11/69

NOTE: Pages 5.3-25 and 5.3-26 deleted per PCR#823



5.3.3.3.3 Initial IMU Alignment and Maneuver Time-to-Go Computations

Both External ΔV and Lambert Aim Point guidance pre-thrust computations require the determination of the IMU middle gimbal angle, MGA, and estimated maneuver time, t_{go} , prior to engine ignition. These operations are shown in Fig. 3.3-7 and -10. The input parameters required for these computations have been defined in Section 5.3.3.3.1. With reference to Fig. 3.3-10, the initial thrust direction unit vector, \underline{u}_{FDI} , and vehicle state vector are used to determine the Preferred IMU Alignment (Section 5.1.4.2). In Fig. 3.3-10 the following parameter definitions apply:

ΔV_K	Required ΔV Monitor Subroutine parameter (Sec. 5.3.3.6).
\dot{m}	Mass flow rate for APS.
$\Delta t_{Tail-off}$ V_e	Cross Product Steering Routine parameters (Sec. 5.3.3.4)
\underline{X}_{SM} \underline{Y}_{SM} \underline{Z}_{SM}	Unit vectors in Basic Reference Coordinates (BRC) of the directions of the IMU X, Y, and Z axes for preferred alignment.
N_C	Required ΔV Monitor Subroutine counter (Section 5.3.3.6).
[REFSMMAT]	Transformation matrix from the BRC System to the IMU or Stable Member Coordinate System (Section 5.6.3.4).
[SMNB]	Transformation matrix from the Stable Member Coordinate System to the Navigation Base Coordinate System (Section 5.6.3.2).
\underline{u}_{FAB}	Unit vector of the assumed thrust acceleration vector along the Navigation Base X axis.

In External ΔV guidance pre-thrust computations, the MGA display is computed as the angle between the input velocity vector \underline{V}_{LV} transformed to the Basic Reference Coordinate System and \underline{Y}_{REF} since \underline{u}_{FDI} computations involving the inplane maneuver angle compensation cannot be made until the engine has been selected by the appropriate P40, 41 or 42 program.

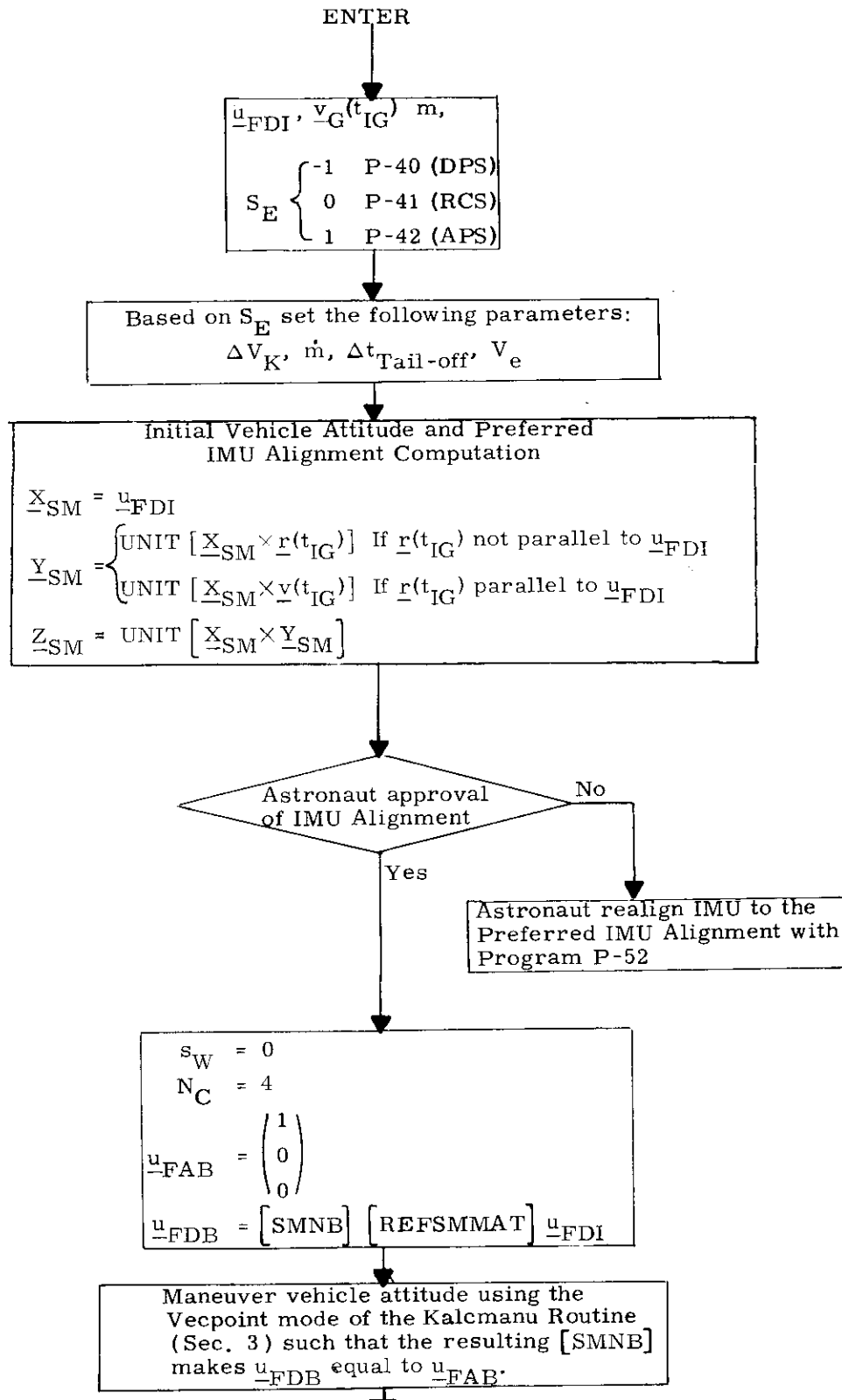


Figure 3.3-10 Initial Attitude and t_{go} Routine
(Page 1 of 2)

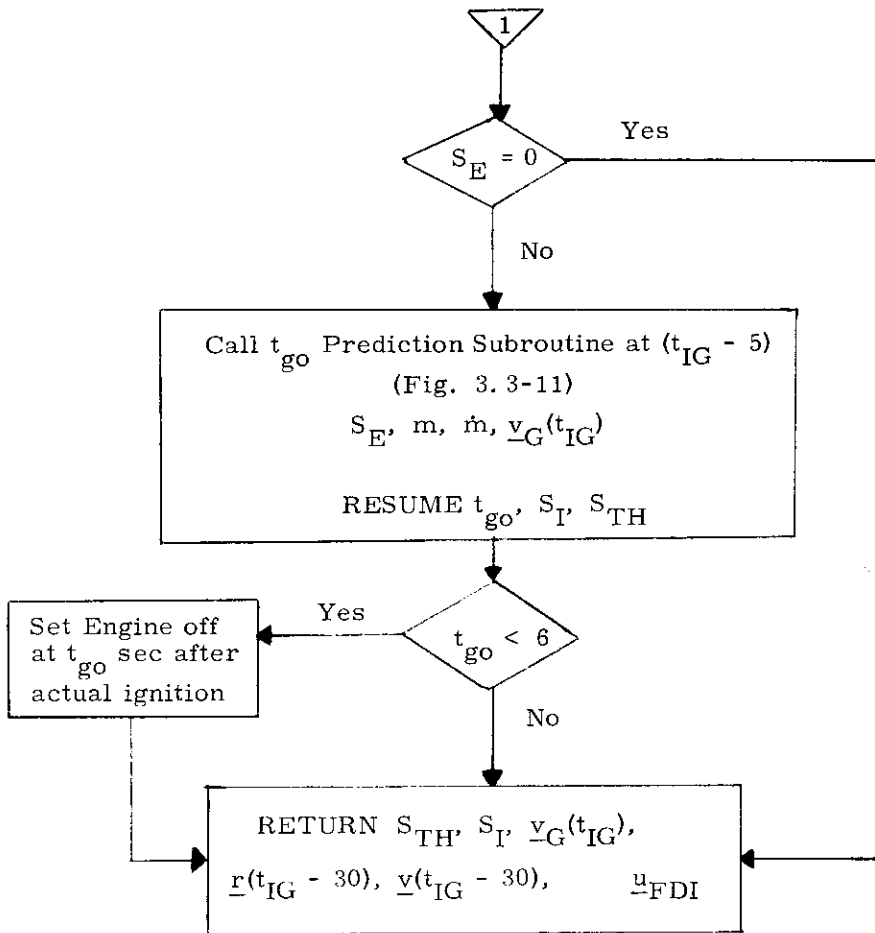


Figure 3.3-10 Initial Attitude and t_{go} Routine
(Page 2 of 2)

5.3-29

If either the DPS or APS were selected for the maneuver, the Time-to-Go Prediction Subroutine of Fig. 3.3-11 is then processed. With reference to Fig. 3.3-11:

- | | | | | |
|-----------|---|---|--|--|
| S_{TH} | DPS throttle inhibit switch | | | |
| | S_{TH} <table border="0"> <tr> <td rowspan="2" style="font-size: 3em; vertical-align: middle;">}</td> <td>0 Allow DPS throttle up</td> </tr> <tr> <td>1 Inhibit DPS throttle up</td> </tr> </table> | } | 0 Allow DPS throttle up | 1 Inhibit DPS throttle up |
| } | 0 Allow DPS throttle up | | | |
| | 1 Inhibit DPS throttle up | | | |
| S_I | Impulse Switch | | | |
| | S_I <table border="0"> <tr> <td rowspan="2" style="font-size: 3em; vertical-align: middle;">}</td> <td>0 Allow steering at the proper time via ΔV Monitor Routine (Section 5.3.3.6)</td> </tr> <tr> <td>1 Indicates short maneuver and no steering is required</td> </tr> </table> | } | 0 Allow steering at the proper time via ΔV Monitor Routine (Section 5.3.3.6) | 1 Indicates short maneuver and no steering is required |
| } | 0 Allow steering at the proper time via ΔV Monitor Routine (Section 5.3.3.6) | | | |
| | 1 Indicates short maneuver and no steering is required | | | |
| F_L | Ullage thrust (2 jet thrust) | | | |
| K_1 | APS impulse velocity acquired in a one second maneuver for a unit mass vehicle | | | |
| K_2 | APS minimum impulse constant | | | |
| K_3 | APS minimum impulse constant equal to the slope of minimum impulse curve. | | | |
| K_4 | 10% DPS engine thrust | | | |
| F_{APS} | APS Thrust | | | |

The initial computation in Fig. 3.3-11 estimates the velocity-to-be-gained after 4 seconds of ullage. If the DPS were selected (P-40), the maneuver time, t_{go} , is then computed on the basis of 10% thrust. If this time is less than 6 seconds no active guidance steering is attempted ($S_I = 1$) and the vehicle attitude is maintained at the pre-thrust alignment throughout the maneuver. If the DPS t_{go} is greater than 95 seconds normal steering will

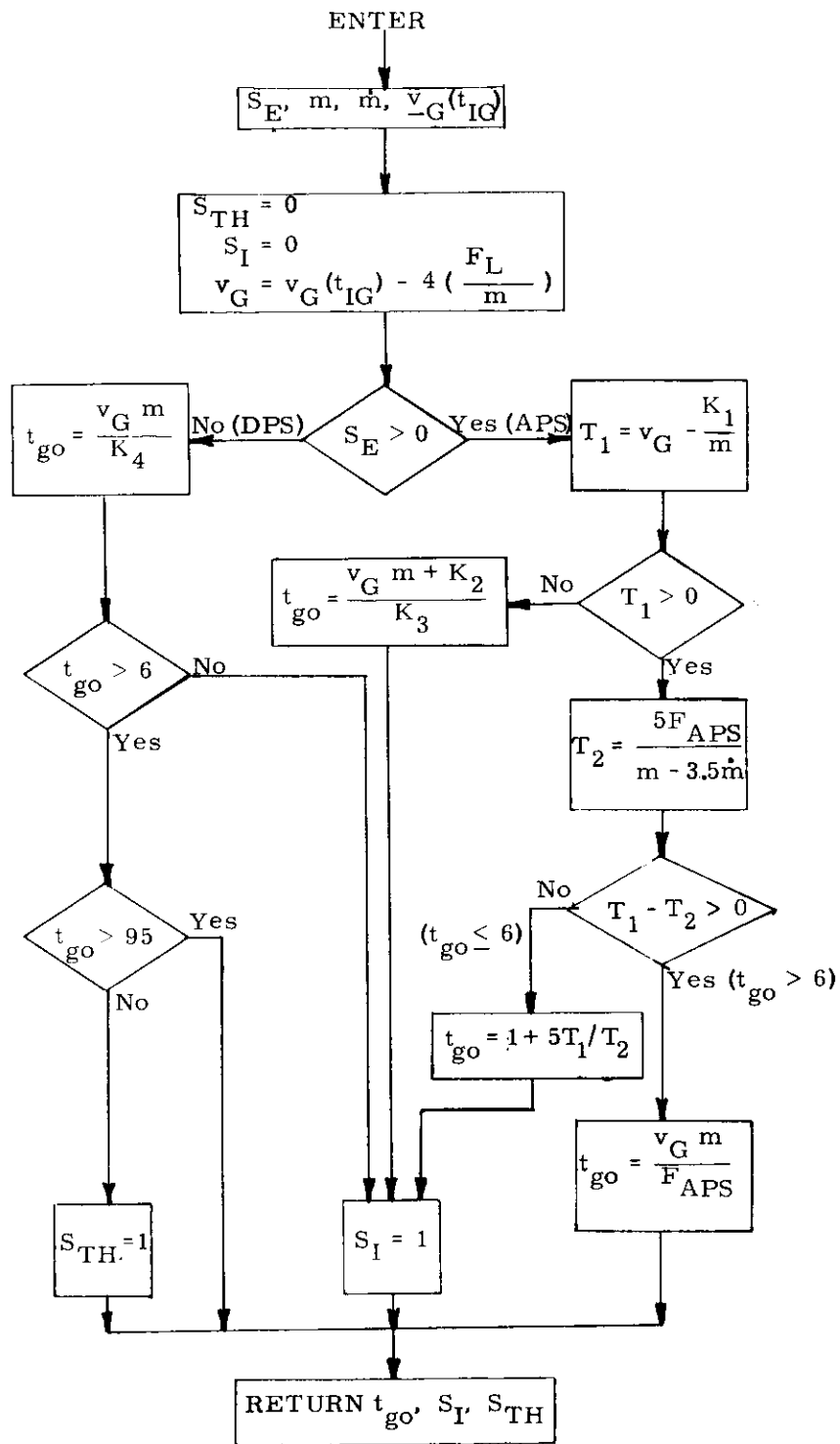


Figure 3.3-11 Time-to-go Predictor Routine

be allowed to throttle the DPS to FTP after the nominal 26 second 10% thrust trim phase. If the DPS t_{GO} is more than 6 seconds but less than 95 seconds, the DPS throttle will be inhibited from throttling up after the nominal 26 second trim phase ($S_{TH} = 1$), and will remain at 10% thrust for the duration of the maneuver. It might be noted that the DPS trim phase duration (e. g. 26 seconds) is an erasable memory parameter and can be modified.

If the APS were chosen (P-42) in Fig. 3.3-11, a check is first made to determine if the maneuver time, is less than one second. If the maneuver time is less than one second, the t_{go} estimate is made on APS minimum impulse test data represented by the constants K_2 and K_3 (See Section 5.8 and Section 6). In this case no active steering is attempted ($S_I = 1$). If the maneuver time is greater than one second but less than 6 seconds, t_{go} is computed as shown in Fig. 3.3-11 and again no active steering is attempted. If the estimated maneuver time is greater than 6 seconds active steering is used.

With reference to Fig. 3.3-10, if the RCS were chosen for the maneuver, (P-41), no t_{go} prediction is made.

If the estimated maneuver time, t_{go} , for either the APS or DPS is less than 6 seconds, the Engine-Off signal is set for t_{GO} seconds after actual ignition as shown in Fig. 3.3-10.

5.3.3.3.4 Ignition Delay Procedures Caused by Pre-Thrust Computations

The normal pre-thrust computations (Section 5.3.3.3.1) require an extrapolation of the LM state vector to thirty seconds prior to nominal ignition time, i. e. $t_{IG} - 30$. If the Coasting Integration Routine of Sec. 5.2.2 does not complete the extrapolation before $t_{IG} - 50$ occurs, then an ignition delay procedure occurs as follows:

- 1.) The astronaut is alerted to this condition by a program alarm.
- 2.) The integration continues one step at a time as shown in MIDTOAVE Routine - Section 5.3.8.
- 3.) The maneuver ignition time is then redefined to be 30 seconds from the resulting LM state vector time, and the normal pre-ignition sequence is started.

5.3-32

Revised LUMINARY
 Added GSOP # 567 PCR # 758* Rev. 7 Date 11/69

5. 3. 3. 4 Cross Product Steering Routine

The cross product steering concept is the basic control concept for both External ΔV and Lambert Aim Point guidance modes. The cross product steering logic is shown in Fig. 3. 3-12. The following parameter definitions not previously described apply to this figure.

Δv	PIPA measured velocity vector over the computation cycle Δt transformed to the Basic Reference Coordinate System.
s_W	A logic switch in the cross product steering routine which when set to 1 allows steering commands and t_{go} calculations to be made. In the external ΔV mode, when set to 0, the velocity-to-be-gained, v_G , is updated by Δv , (Section 5. 3. 3. 5). s_W is set to 1 in the ΔV Monitor Subroutine of Section 5. 3. 3. 6 after engine thrust has been detected, and set to 0 when the computed time-to-go first becomes less than four seconds.
$\Delta t_{Tail-off}$	A negative constant representing the duration of a burn at 40 percent thrust equivalent for DPS, and 100% for APS to the tail-off impulse after the engine off signal is issued. It is initialized to one of 3 values from fixed memory: $\Delta t_{Tail-off}(APS)$ for APS Maneuvers $\Delta t_{Tail-off}(DPS)$ for DPS maneuvers at 40% thrust $\Delta t_{Tail-off}(P70)$ for P70 DPS abort (100% thrust)
V_e	Engine exhaust velocity = $g I_{SP}$
EXTDELVFLG	External ΔV flag 1 = external ΔV maneuver 0 = Lambert Aimpoint Maneuver

[REFSMMAT] Transformation matrix from the BRC system to the IMU or Stable Member Coordinate system (Section 5. 6. 3. 4).

\underline{u}_{FDP} Desired thrust acceleration vector in Platform Coordinates.

FLAUTOX X-Axis Override Flag.
 If set to 0, will be in yaw attitude hold (no rotation about outer gimbal axis) and will bypass the window vector pointing error section of FINDCDUW routine. Also, X axis override is permitted if flag is set to 0.

FLDOC Docking Configuration Flag.
 0 setting indicates LM vehicle only.
 1 setting indicates LM docked to CSM used in TEI SPS backup and return maneuvers.
 This flag set by the astronaut with an extended verb in routine R-03 (Section 4) prior to entering the thrust programs P-40 and P-42.

With reference to Fig. 3. 3-12, the logic switch s_W is set to zero by the sequencing routine. When s_W is zero the only function of the cross product steering routine is to update the velocity-to-be-gained vector \underline{v}_G with $\Delta \underline{v}$ when in the external ΔV mode.

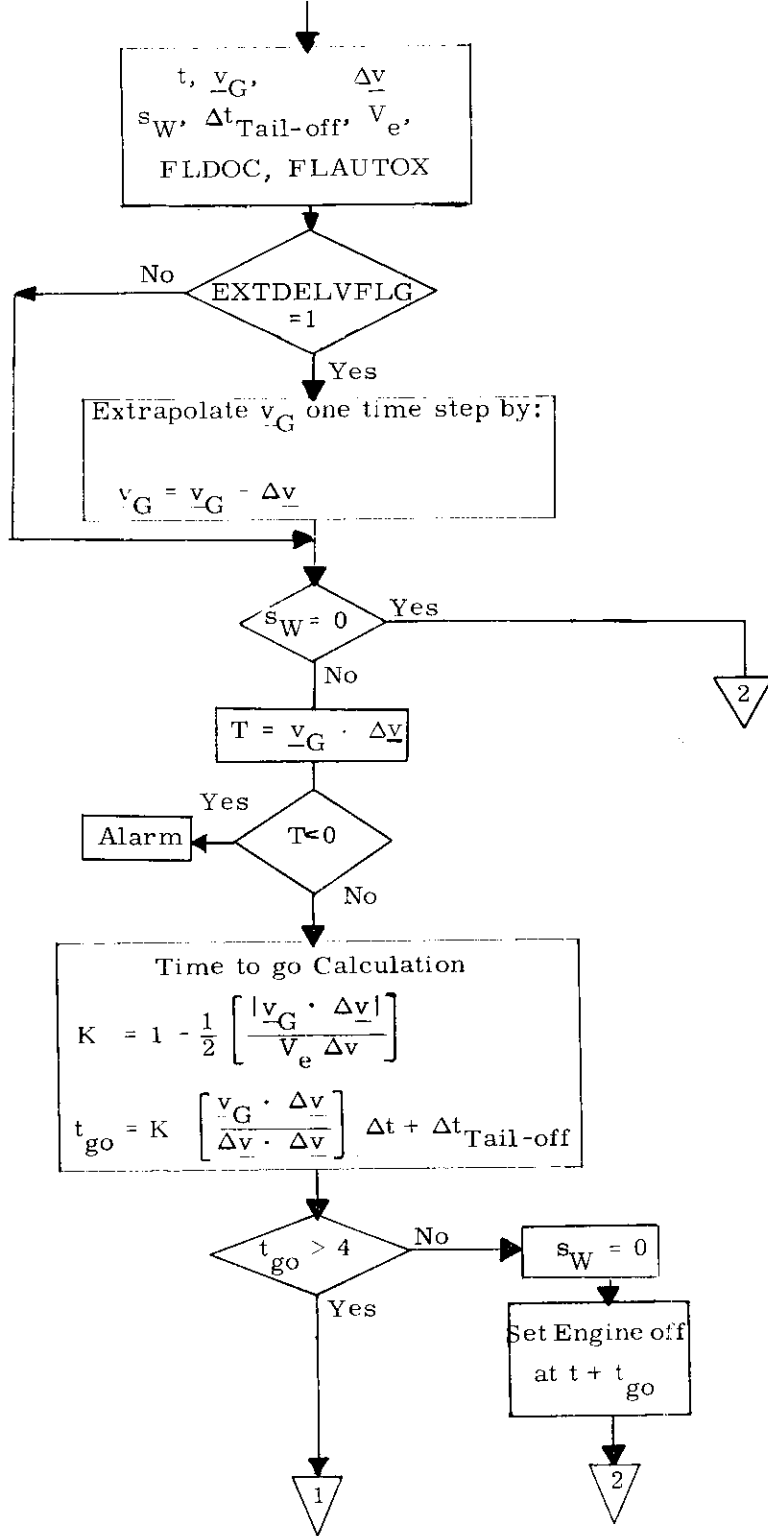


Figure 3.3-12 Cross Product Steering Subroutine
(page 1 of 2)

5.3-35

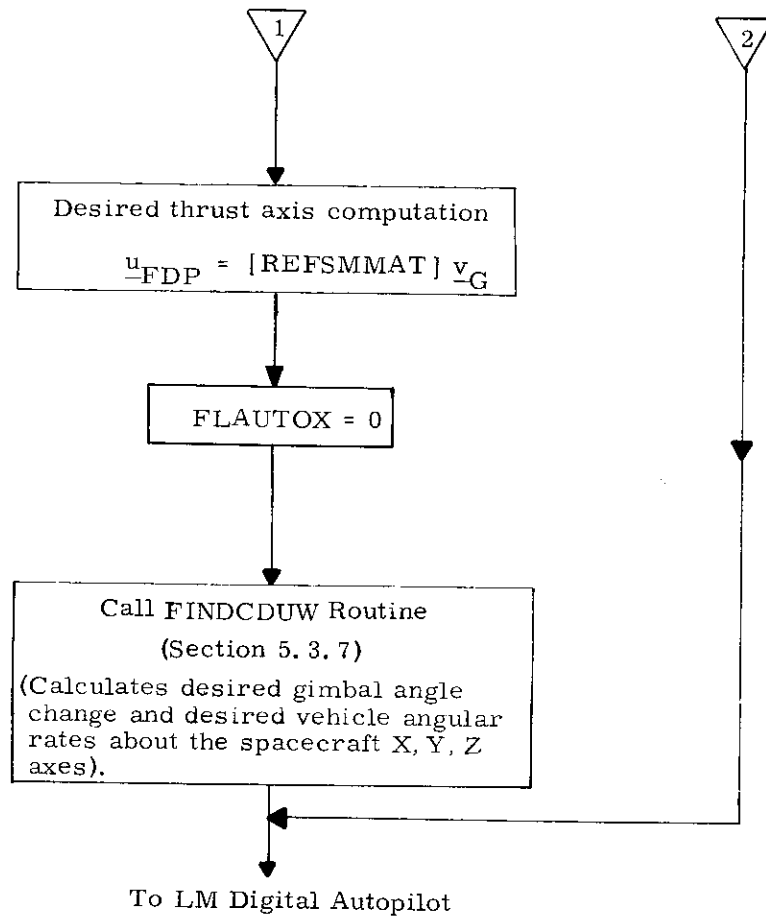


Figure 3.3-12 Cross Product Steering Subroutine
(page 2 of 2)

Switch s_W remains zero for short duration thrust periods or during the initiation of long duration thrust periods before the thrust has increased above the threshold level. In both these cases there is no active steering and the vehicle attitude is held at the pre-thrust alignment. When s_W is set to 1, active steering is performed. The time-to-go, t_{go} , computation and steering commands, \underline{u}_{FDP} , are performed as shown in Fig. 3.3-12. When the computed t_{go} becomes less than 4 seconds, then the engine-off signal is set and switch s_W is reset to zero. For the remainder of the maneuver, no further computations are made except for v_G updating.

The steering command generated prior to calling the FINDCDUW Routine is the desired unit thrust direction vector in platform coordinates, \underline{u}_{FDP} . In addition, the X axis override flag FLAUTOX is set to 0 (Fig. 3.3-12). This will result in bypassing the window vector pointing error section of the FINDCDUW Routine and will also permit X axis override. The FINDCDUW Routine of Sec. 5.3.7 is the interface routine between the guidance program and the LM Digital Autopilot (DAP) for all powered maneuvers. This interface routine generates three angle commands representing the desired change in IMU CDU gimbal angles ($\Delta CDUD$), the desired angular rate vector, $\underline{\omega}_{DV}$ expressed in vehicle coordinates, and commanded attitude lag angles.



5.3.3.5 Velocity-to-be-Gained Routine

The velocity-to-be-gained computations shown in Fig. 3.3-13 are those carried out during the Lambert Aim Point powered flight guidance. The velocity-to-be-gained computation for the External ΔV guidance mode is simpler than that for the Lambert Aim Point guidance mode. The External ΔV velocity-to-be-gained computation is that shown in the cross product steering routine of Fig. 3.3-12 and is

$$\underline{v}_G = \underline{v}_G - \underline{\Delta v}$$

The velocity-to-be-gained computations for the Lambert Aim Point guidance mode involve the determination of a new \underline{v}_G by processing the ASTEER Subroutine.

The following parameter definitions refer to Fig. 3.3-13.

$\left. \begin{array}{l} \underline{r}(t) \\ \underline{v}(t) \\ t \end{array} \right\}$ active vehicle state vector

$\underline{r}(t_2)$ Offset target intercept position vector at time t_2 . This parameter is determined by the preceding targeting program (P-34, P35).

5.3-38

Revised

Added

LUMINARY

GSOP # 567

PCR # 844

Rev. 7

Date 11/69

t_2 Intercept time of arrival associated with the offset target vector $\underline{r}(t_2)$. This is an input target parameter.

t_{IG} Nominal ignition time.

$\underline{v}_R(t)$ Required velocity vector at time t .

$\underline{v}_G(t)$ Velocity-to-be-gained vector at time t .

S_R Target rotation switch set in P-34, P-35 indicating that the target vector was rotated into the orbital plane due to proximity to 180° transfer.

S_R $\left\{ \begin{array}{l} 1 \text{ Target vector rotated} \\ 0 \text{ No rotation} \end{array} \right.$

\underline{g}_b Component of the earth gravity acceleration vector representing earth oblateness effects (Section 5.3.2).

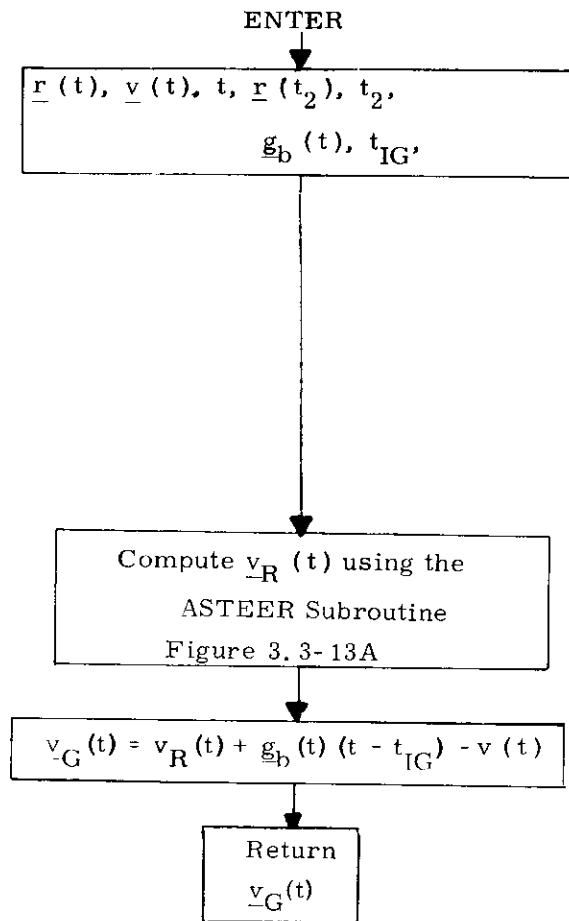


Figure 3.3-13 Velocity-to-be-Gained Subroutine

It should be noted that in Fig. 3.3-13 the velocity-to-be-gained, v_G , derived from the ASTEER solution using an offset target vector is modified by the term $g_b(t) [t - t_{IG}]$. This term is an approximation to the velocity change contributed by the earth oblateness effect. The compensation used in this subroutine is computed as the current oblateness acceleration, $g_b(t)$, multiplied by the time since nominal ignition $(t - t_{IG})$ where t_{IG} is the nominal ignition time. This correction is zero for lunar orbits. The objective of this correction is to reduce cut-off errors due to finite maneuver time effects, and to minimize commanded thrust attitude variations during the maneuver. These two effects occur during long maneuvers because in accounting for earth oblateness effects in the initial targeting programs (P-34 or P35), it is assumed that an impulsive maneuver will be applied at ignition time. Since a finite maneuver time is required, the precomputed target aim point becomes less accurate as the maneuver progresses. The $g_b(t - t_{IG})$ correction is an approximate substitute for a retargeting procedure which can not be performed during a powered maneuver

5.3-41

Revised
 Added

LUMINARY

GSOP # 567

PCR # 844

Rev. 7

Date 11/69

5.3.3.5.1 ASTEER Subroutine

The ASTEER Subroutine solves for the required velocity, (\underline{v}_R) given the semi-major axis parameter (a). Prior to ignition, the t_{IG} vehicle state vector is used in the Lambert Subroutine to obtain a semi-major axis (a) satisfying the time of flight and the conic aim vector. Then, starting at approximately $t_{IG} - 28$ and at each guidance computation cycle (2 sec), the ASTEER Subroutine is called to determine an updated required velocity to satisfy the parameter a. Figure 3.3.13A presents the flow diagram for the \underline{v}_R calculation. The following definitions refer to this figure.

$\underline{c}(t)$	chord vector $\underline{r}_2(t) - \underline{r}(t)$
$\underline{r}_2(t_2)$	conic aim vector
s	semi-perimeter $[\underline{r}(t) + \underline{r}_2(t_2) + \underline{c}(t)] / 2$
SGNA	$\begin{cases} 1 & \text{if transfer angle less than 180 degrees} \\ -1 & \text{if transfer angle greater than 180 degrees} \end{cases}$
t_m	minimum energy transfer time = $\sqrt{\left(\frac{s^3}{2\mu}\right)} \left\{ \frac{\pi}{2} - \text{SGNA} \left(\text{ARCSIN} \sqrt{\frac{s-c}{s}} - \frac{1}{s} \sqrt{c(s-c)} \right) \right\}$
a	semi-major axis obtained from Lambert Subroutine
UT	an erasable time quantity for updating the semi-major axis a
$\underline{1}_N$	Unit normal to the trajectory computed in the lambert subroutine

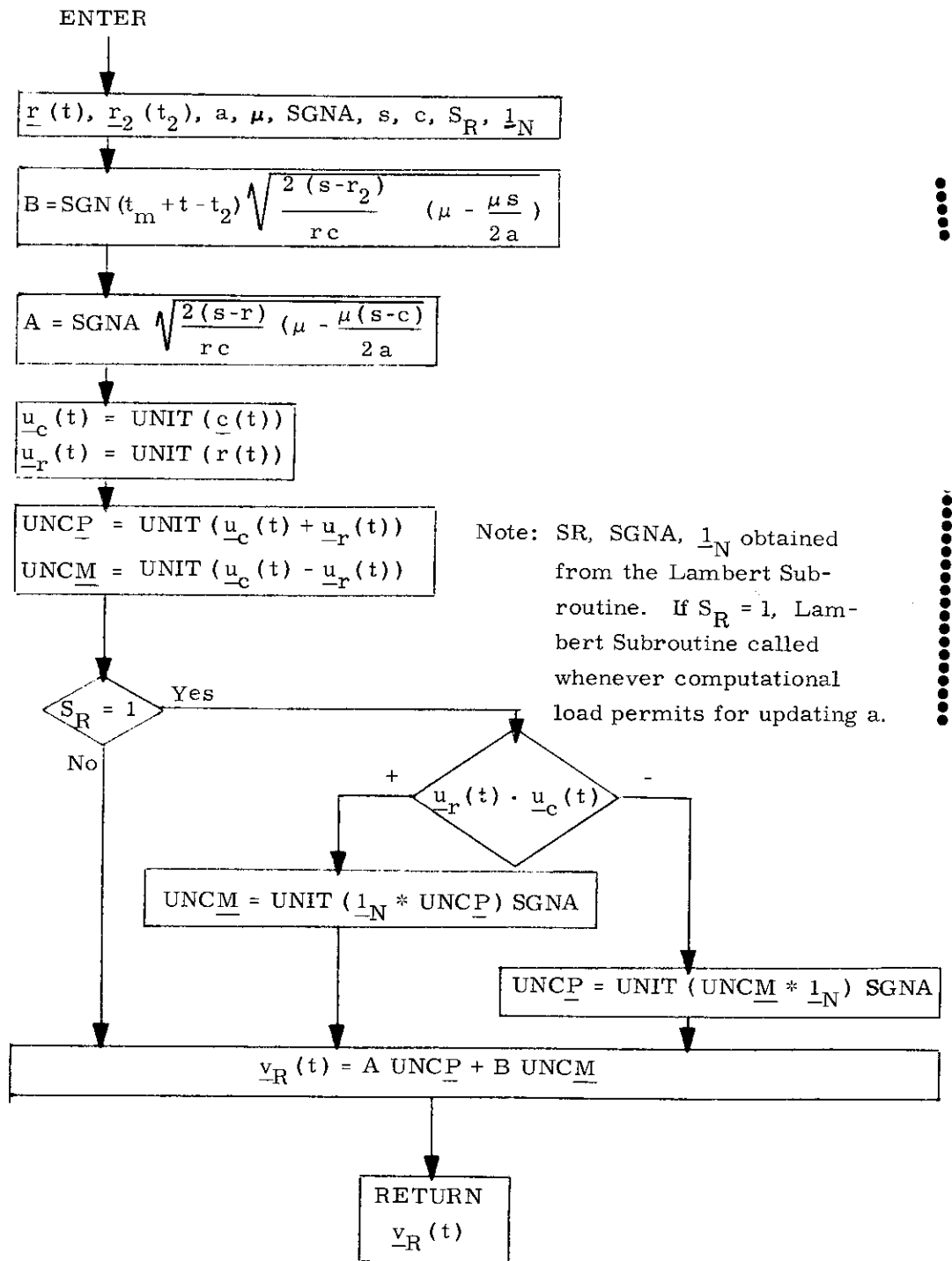


Figure 3.3-13A ASTEER Subroutine

5.3.3.6 ΔV Monitor Subroutine

The purpose of the ΔV Monitor Subroutine is to detect and monitor the presence of engine thrust every 2 seconds commencing at 2 to 4 seconds after the actual ignition time. The subroutine also turns on the cross product steering routine if the maneuver is not a short burn. If subsequent to 2 sec after the nominal ignition time the engine thrust is detected to be below a given level for 5 cycles of this routine, control is transferred to the DPS/APS Thrust Fail Routine R-40. If during the thrusting period the engine thrust is detected to be below a given level for 2 cycles of this routine, control is transferred to the DPS/APS Thrust Fail Routine R-40.

The ΔV Monitor Subroutine logic diagram is illustrated in Fig. 3.3-14. The following parameter definitions apply to this section (5.3.3.6):

- | | |
|--------------|--|
| Δv | PIPA measured velocity change over the last computation cycle. |
| N_C | A counter in the ΔV Monitor Subroutine which indicates an engine failure when ≤ 0 . This counter is initialized during pre-thrust computations to a value of 4. |
| S_I | Impulse Switch |
| S_I | $\left\{ \begin{array}{l} 0 \text{ Allows steering at the proper time via } \Delta V \text{ Monitor Subroutine} \\ 1 \text{ Indicates short maneuver and no steering is required} \end{array} \right.$ |
| ΔV_K | A constant set by the pre-thrust routine which establishes the ΔV which must be sensed in a 2 second computation interval if the engine is to be considered on. |

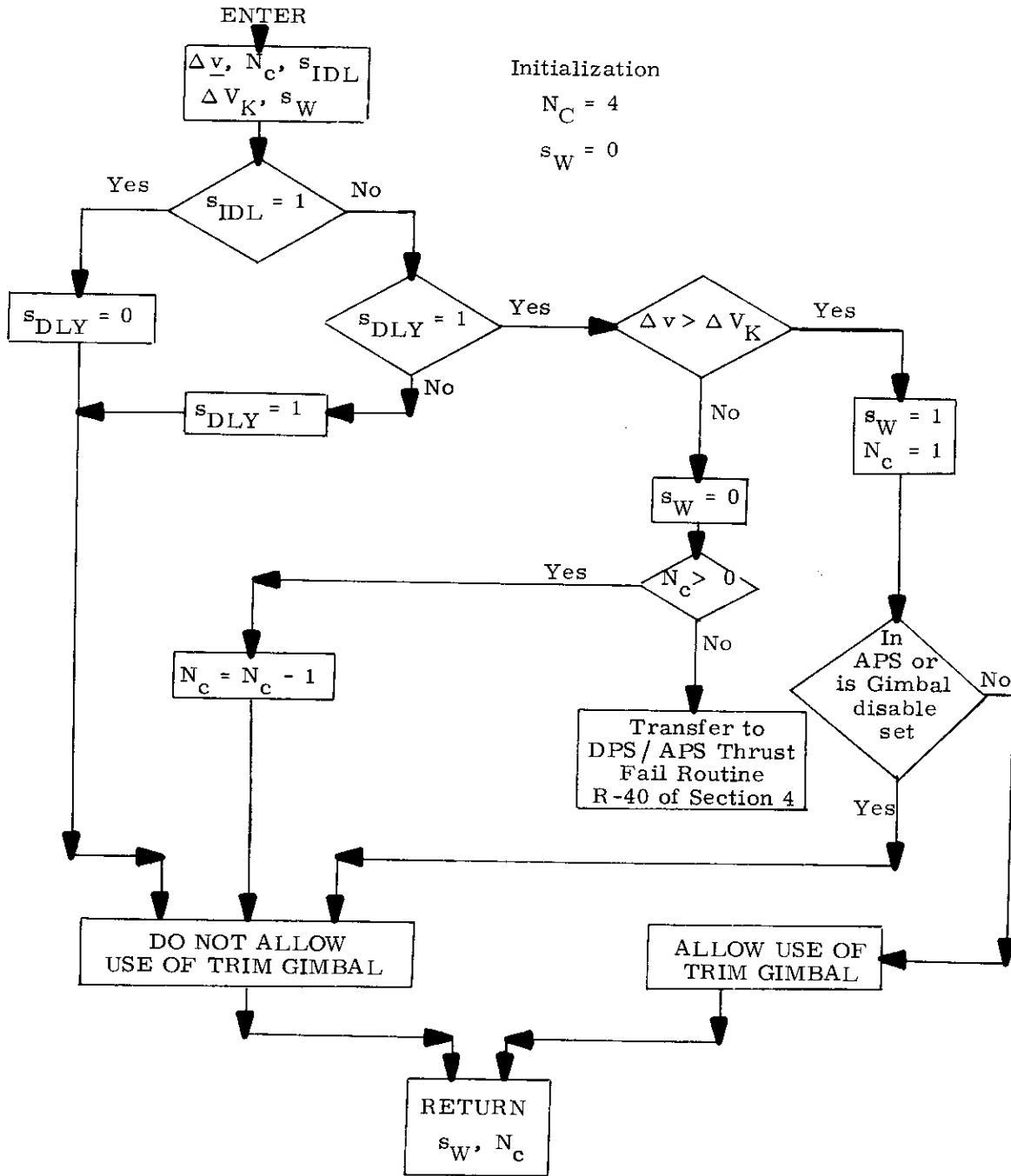


Figure 3.3-14 ΔV Monitor Subroutine

s_W A logic switch in the cross product steering routine which when set to 1 allows steering commands and t_{go} calculations to be made. When in the external ΔV mode when set to 0, the velocity-to-be-gained, v_G , is updated by Δv . (Section 5.3.3.4).

s_{IDL} ΔV Monitor Idle Switch

s_{IDL} $\left\{ \begin{array}{l} 1 \text{ By pass } \Delta V \text{ Monitor Operation} \\ 0 \text{ Activated } \Delta V \text{ Monitor Operation} \end{array} \right.$

s_{IDL} is set to 1 prior to ignition for all maneuvers. For all RCS maneuvers (P-41), APS maneuvers (P-42) and DPS maneuvers (P-40) of less than 6 seconds estimated duration ($s_I = 1$) s_{IDL} is maintained at 1. For DPS and APS maneuvers of greater than 6 seconds, s_{IDL} is set to zero at ignition. s_{IDL} is set to 1 when $t_{go} < 4$ seconds.

s_{DLY} ΔV Monitor Delay Switch

s_{DLY} $\left\{ \begin{array}{l} 0 \text{ By pass } \Delta v \text{ test in } \Delta V \text{ Monitor} \\ 1 \text{ Activate } \Delta v \text{ test in } \Delta V \text{ Monitor} \end{array} \right.$

The function of s_{DLY} is to prevent the ΔV Monitor from checking Δv until $(t_{IG} + 2)$, since s_{IDL} is set to 0 at t_{IG} .

It should be noted that the ΔV Monitor Subroutine is activated by the Servicer Routine during all PGNCs controlled thrust maneuvers.* The engine fail procedures activated by this subroutine differ for various mission phase maneuvers as described in R-40 of Section 4.

* The ΔV Monitor Subroutine is activated at $(t_{IG} + 2)$ for DPS and APS maneuvers exceeding 6 seconds.

5.3.4 LUNAR LANDING GUIDANCE

5.3.4.1 Introduction

Lunar Landing Guidance is divided into two distinct parts:

1. Pre-ignition calculations (also called the Ignition Algorithm)
2. A guidance cycle starting before ullage initiation and ending after touchdown.

The pre-ignition calculations use the guidance equations as a subroutine. Therefore, the description and flow graphs of the pre-ignition calculations and the guidance cycle are grouped together.

The guidance cycle produces, in order (except for Auto P66):

1. A time of state vector validity (called PIPTIME)
2. The corresponding state vector*
3. Command thrust acceleration and window-pointing vectors, based on the state vector, according to the current phase
4. Throttle commands, based on command thrust acceleration
5. DAP commands, based on the command thrust acceleration and window-pointing vectors.

(For Auto P66 items 4 and 5 are interchanged)

Table 3.4.1-1 relates these computations to the routines where the computations are made.

Figure 3.4.1-1 illustrates the phases of the lunar landing. These phases are described as follows.

PRE-IGNITION

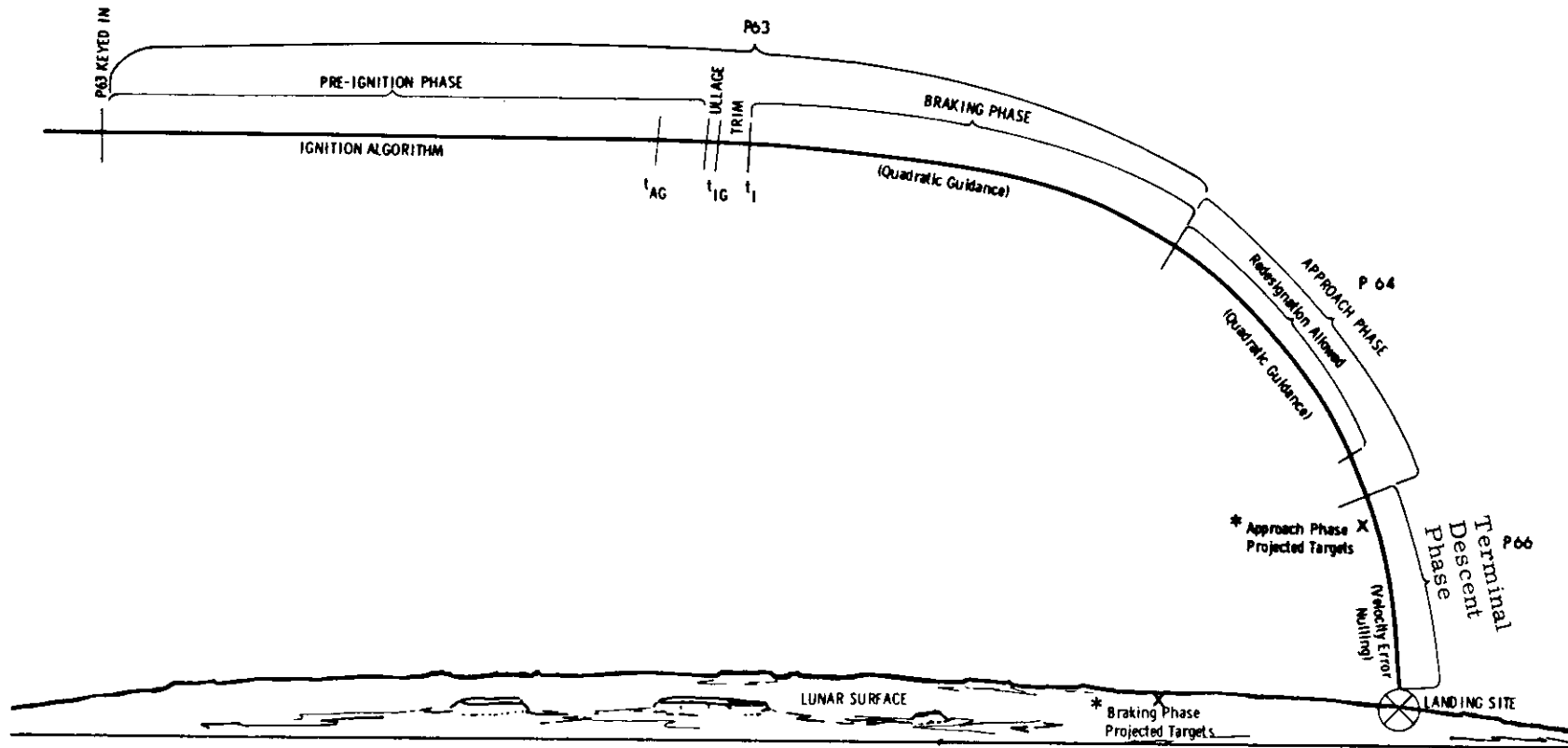
The astronaut starts the pre-ignition phase by selecting program P63 some time in the coasting descent orbit which follows the DOI burn. The purpose of the pre-ignition phase is to determine ignition time based on the LM state in guidance coordinates. P63 determines the foregoing iteratively, while simultaneously determining the unit vectors defining the guidance frame, and the thrust acceleration and

*The state vector is calculated using the following information: the previous state vector, readings from the PIPAs, estimated acceleration due to gravity, and readings from the radar (when radar readings are valid; c.f., LR Data Read Routine and LR Data Reasonableness Routine).

Table 3.4.1-1 Guidance Cycle Computations
and the Acting Routines

Calculation Of	Calculated By
State Vector a. from PIPA's b. from Radar	State Vector Update Routine LR Data Read Routine LR Velocity and Range Measurement Data LR Data Reasonableness Test Routine
Command Thrust Acceleration and Window-pointing Vectors	Ignition Algorithm and Guidance-and-Control Routine
Throttle commands DAP commands	Throttle-Command Routine FINDCDUW Routine

5.3-49



The above diagram, purposely drawn out of scale to fit on one page, illustrates the way the Lunar Landing Guidance Equations fit along the actual flight trajectory. The uncircled X's show approximate positions of projected targets for each phase. They are the targets that the spacecraft would achieve if the phase failed to end so that the same equations and targets were used until TTT reached zero.

*Targets achieved at these positions include position, velocity, acceleration, and one component of jerk. The X indicates the approximate position only; the braking phase targets may be located above or below the surface.

LUNAR LANDING PHASES

Figure 3.4.1-1

window-pointing vectors which will be commanded on the first guidance cycle of the braking phase. During the pre-ignition phase, the vehicle travels in free-fall (unpowered) flight. Under normal conditions, the proper ignition time will be determined several minutes before the nominal ignition time.

After the above computations are complete, the MIDTOAVE Routine (R41) is entered to extrapolate the LM state vector forward to the time at which the guidance cycle is initiated.

The Attitude Maneuver Routine (R60) orients the spacecraft for the succeeding ULLAGE AND TRIM phase such that the LM X-axis lies in the direction of the first thrust command to be issued during the succeeding braking phase, as predicted by the pre-ignition computations.

The guidance cycle is initiated shortly before the initiation of ullage so that the LM state vector can be maintained using accelerometer measurements (average-G equations). The routines processed each guidance cycle are illustrated in Fig. 3.4.1-2, except that the Guidance-and-Control Routine, the Throttle-Command Routine, and the FINDCDUW Routine are excluded from the guidance cycle until the start of the braking phase.

ULLAGE AND TRIM

This phase is a continuation of program P63.

The LGC Master Ignition Routine starts RCS ullage and shortly thereafter ignites the engine at minimum throttle setting for a period of time sufficient to trim the thrust vector to point through the center of mass. The Guidance-and-Control Routine, the Throttle-Command Routine, and the FINDCDUW Routine remain excluded from the guidance cycle; therefore, the engine maintains a constant (minimum) throttle setting, and the spacecraft maintains a constant attitude throughout the ULLAGE AND TRIM phase.

BRAKING PHASE

This phase concludes program P63.

The throttle is commanded to maximum at the precise instant computed by the pre-ignition computations for starting the braking phase. This is accomplished by a WAITLIST call by the LGC Master Ignition Routine. The throttle must be advanced on time. The Guidance-and-Control Routine, the Throttle-Command Routine, and the FINDCDUW Routine are added to the guidance cycle at this time. The first reading (during the braking phase) of the accelerometers occurs some time later.

To utilize propellant efficiently, it is necessary to operate the DPS at maximum thrust as long as possible. In order to provide landing-site visibility during the succeeding approach phase, it is necessary that the braking phase

A TIME CHART FOR LUNAR LANDING GUIDANCE

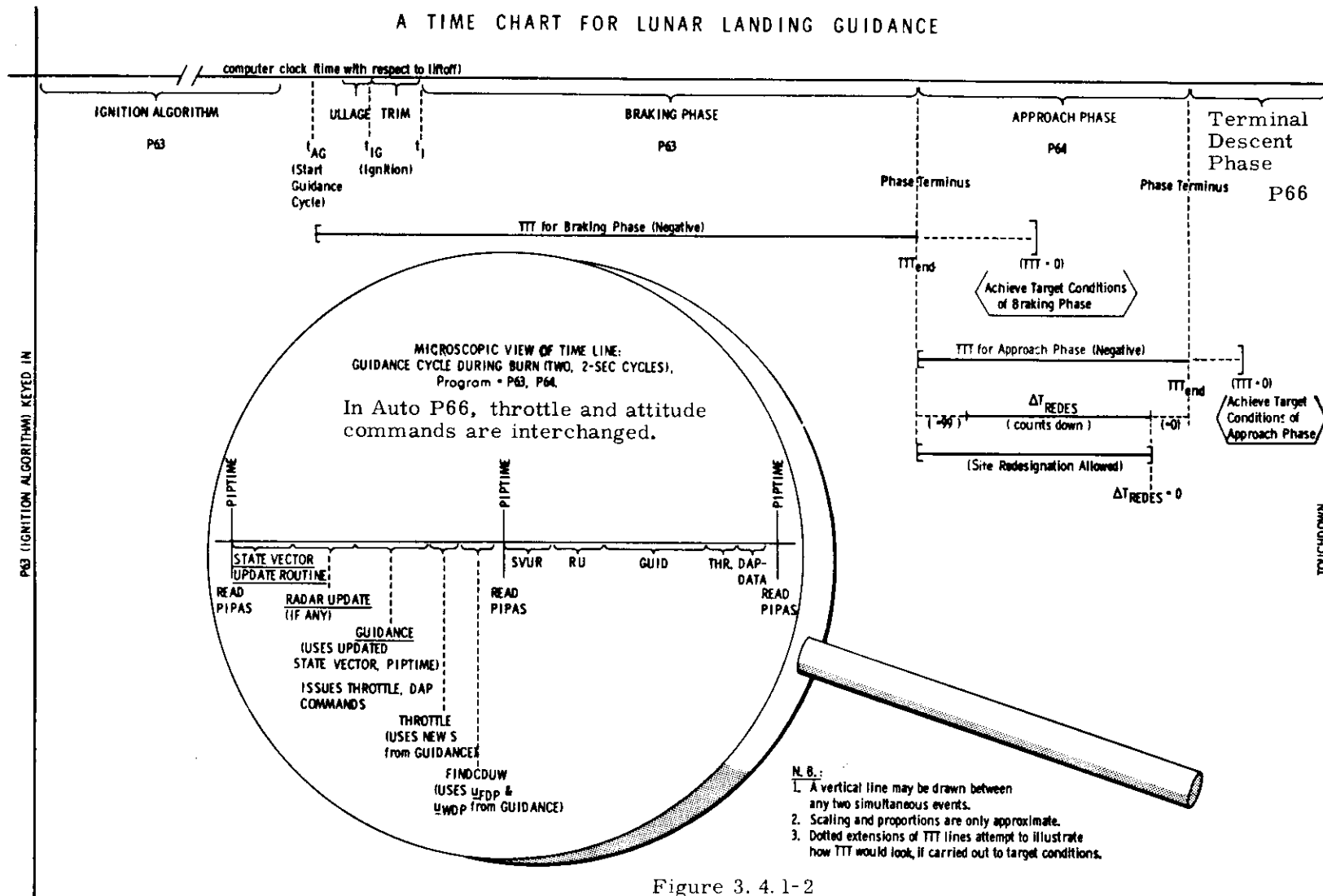


Figure 3. 4. 1-2

Revised

Added

LEARNARY TO

OSOP # 1R-567

PCR # 988

REV. B Date 3/70

5.3-51

P63 (IGNITION ALGORITHM) KEYS IN

TOUCHDOWN

targets (position and velocity) be achieved with reasonable accuracy. This requires that the DPS be operated in the continuously throttleable region during the final part of the braking phase.

To accomplish the above objectives, a throttle-command logic is employed which will hold the throttle at maximum until the command thrust acceleration is less than a pre-selected value (FLO). By properly selecting the braking-phase targets, this switching (or throttling-down) is made to occur at a specified time before the end of the phase on a nominal trajectory.

During the early part of the braking phase, the astronaut may yaw the vehicle about its X-axis to observe the lunar terrain. When the altitude drops below a pre-selected value, the guidance automatically commands a window-up yaw attitude to ensure proper landing radar operation.

The state-vector updatings during the braking phase are based solely on IMU data until the estimated altitude is below a pre-selected value. After this time, landing radar altitude measurements are used to update the vehicle's state guidance cycle if they pass certain tests. Velocity measurements are taken and used for state-vector updates when the estimated speed falls below a pre-selected value. There is no altitude check before reading LR Altitude information.

APPROACH PHASE

This phase is also called the VISIBILITY phase and is synonymous with program P64.

The initial state and the targets for the approach phase are chosen so that the vehicle's X-axis (i. e., the thrust vector) is elevated sufficiently high above the local horizontal to permit landing-site visibility for the bulk of the phase. The DPS is operated in the continuously-throttleable region throughout the approach phase.

The state vector of the LM is updated with both altitude and velocity-component LR measurements each guidance cycle throughout the phase. The updatings take place immediately after the accelerometer output data are processed. An altitude measurement and a velocity-component measurement are processed sequentially at each updating time, with the altitude measurement preceding the velocity-component measurement. The radar measures any given velocity component every third guidance cycle. Pre-stored weighting functions are used in the processing of the radar data.

The astronaut has the option of manually redesignating the landing site by means of landing point designator (LPD) switch commands. These commands are computer inputs from the rotational hand controller. Each guidance cycle, a

computation is made of the new landing site location. The depression angle of the line-of-sight to the current site below the vehicle's Z-axis is computed and displayed to the astronaut throughout the approach phase. Normally, the current site is visible in the window, and the attitude commands are generated to yaw the vehicle about its X-axis until the line-of-sight to the current site is in the plane of the vehicle Z - and X-axes; i. e., it is along the LPD index line. Using the computed site depression angle on the DSKY, the astronaut can look out the window and decide where to redesignate the landing site, if he so desires.

VERTICAL DESCENT

P66 [also called Rate of Descent (ROD)] is entered automatically at the conclusion of P64. P66 can also be entered at any time after initial throttle-up by switching to Attitude Hold and manipulating the Rate of Descent Controller.

There are two parallel sections of P66, allowing for four possible combinations of PGNCS control:

VERTICAL	{	Auto Throttle
		Manual Throttle
HORIZONTAL	{	Auto Mode
		Attitude Hold Mode

When in Auto Throttle, P66 issues commands to the DPS such as to maintain a reference altitude rate. When P66 is entered, or whenever the throttle mode is manual, this reference rate is set to the current altitude rate. When in Auto Throttle, the reference rate can be manually incremented by manipulating the Rate of Descent Controller. When not in Auto Throttle, of course, the throttle commands are controlled by the astronaut.

When in Auto Attitude Mode--independently of the setting of the Throttle switch--P66 controls the attitude such as to null horizontal velocity. When in Attitude Hold, the astronaut controls attitude. If a PGNCS velocity error exists, the only way to null the true velocity relative to the moon is to switch to Attitude Hold. This is because there is currently no mechanism whereby the astronaut can correct the PGNCS velocity error.

5.3.4.2 Lunar Landing Coordinate Systems

Several coordinate systems are used for navigation and guidance of the LM during the powered landing maneuver. Each is a right-hand, orthogonal system. Figure 3.4.2-1 lists the various names by which each frame is known; it lists the mnemonic identifying initial used in this section; and it identifies the matrices used in this section and their equivalents from other sections, which transform between the coordinate systems.

The subscripts of these matrices are chosen according to the usual mathematical conventions which allow a product of transformations to be constructed by cancellation of internal subscripts. That is,

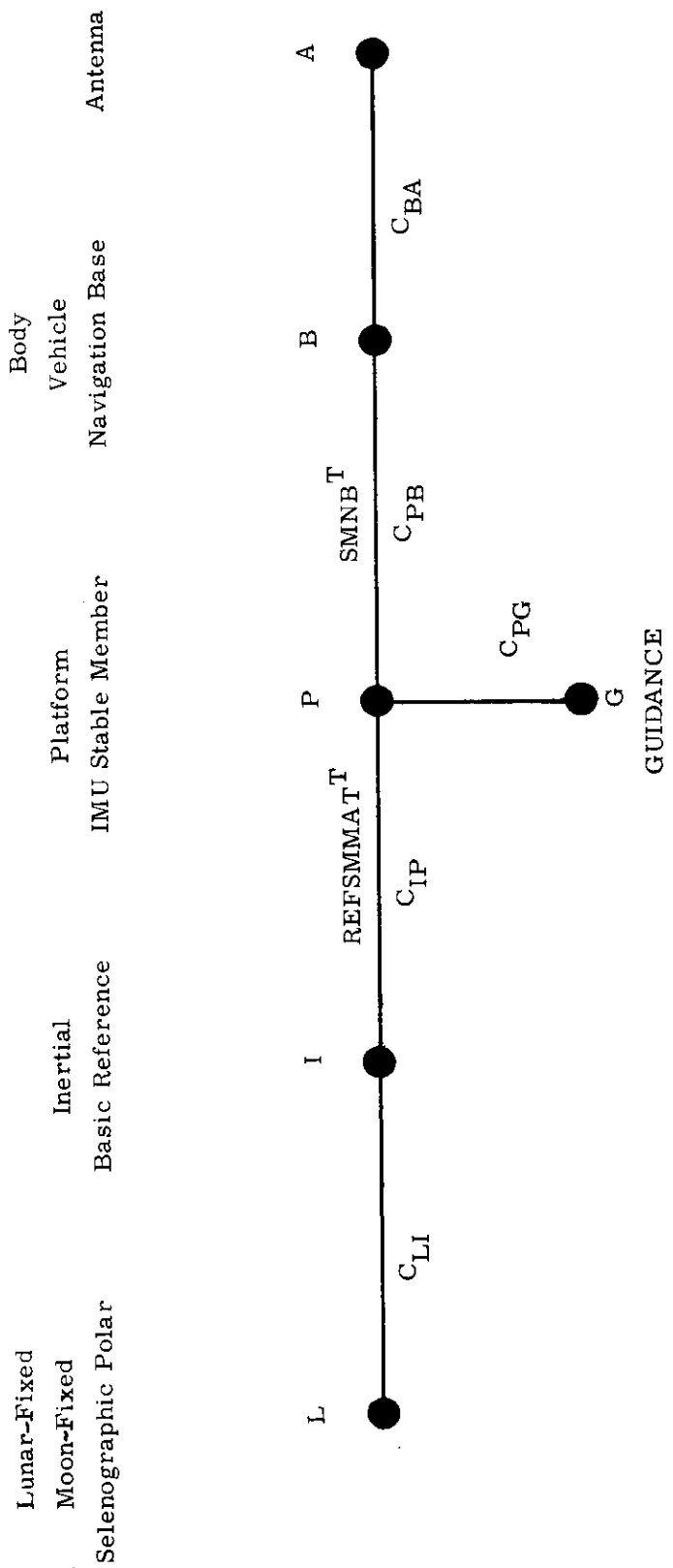
$$\underline{V}_L = C_{LA} \underline{V}_A = C_{LI} C_{IP} C_{PB} C_{BA} \underline{V}_A$$

where \underline{V}_L is a column vector whose components are expressed in lunar-fixed coordinates, and \underline{V}_A is the same vector in antenna coordinates. With this notation, the matrix C_{LA} is given by

$$C_{LA} = \begin{bmatrix} \underline{C}_{XLA} \\ \underline{C}_{YLA} \\ \underline{C}_{ZLA} \end{bmatrix}$$

where \underline{C}_{XLA} , \underline{C}_{YLA} , \underline{C}_{ZLA} are the unit row vectors of the lunar-fixed coordinate frame expressed in antenna coordinates. These subscript conventions are opposite to those used in the Lunar Landing Guidance section of Revision 3 of the GSOP, Section 5.

Only those coordinate systems peculiar to landing navigation and guidance will be described in this section. The Lunar-Fixed, Inertial, Platform, and Body coordinates are described in the Coordinate Systems subsection of the Introduction to the GSOP; the transformations between Inertial, Platform, and Body coordinates are defined in the IMU Routines subsection of the General Service Routines, and the transformation between Lunar-Fixed and Inertial coordinates is defined in the Planetary Inertial Orientation Subroutine subsection of the Basic Subroutines.



Coordinate Frames, Aliases, and Transformations Between Adjacent Frames

Figure 3.4.2-1

Guidance Coordinates

The origin coincides continuously with the current landing site (the frame rotates with the moon). The X-axis is vertical, the Z-axis lies in the plane of the trajectory relative to the moon at phase terminus and points forward, and the Y-axis completes a right-hand triad. Thus, the origin and orientation of the guidance frame are altered each time the landing site is redesignated.

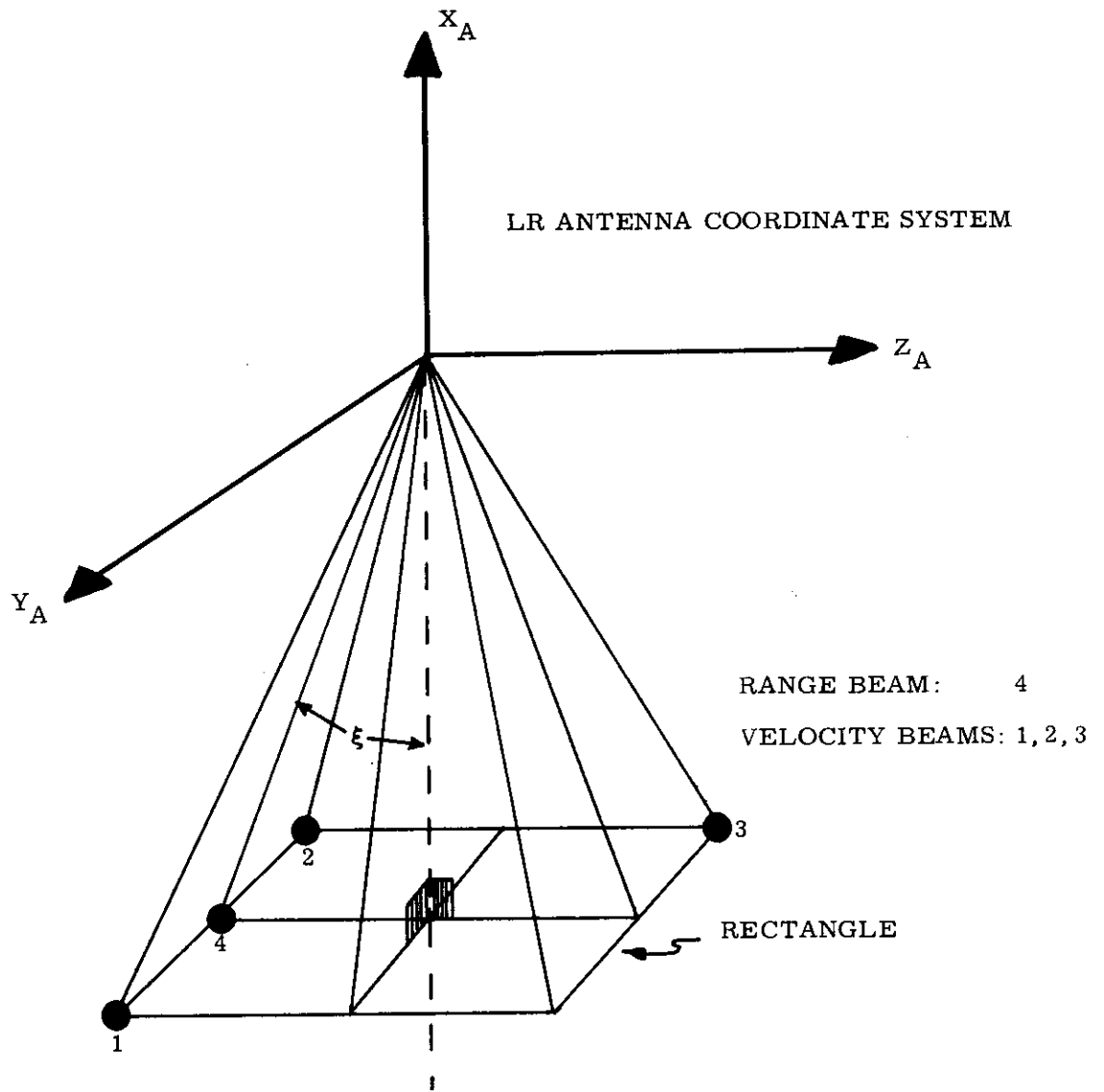
Landing-Radar Antenna Coordinates

The LR antenna coordinate system (Fig. 3.4.2-2) is the one in which the range and velocity-component data used to update the LM navigation system are obtained. The antenna axes are fixed with respect to the range and velocity beams. The X-axis is oriented along the axis of symmetry of the radar beams. The Z-axis is normal to the X-axis and directed forward, symmetrically oriented with respect to the rear velocity beams. The Y-axis is perpendicular to the X- and Z-axes, and directed so as to form a right-hand system. LR range data are referenced to the LR antenna coordinate system by the angle ξ as shown in Figure 3.4.2-2.

The following description helps to visualize the lunar landing coordinate frames. Assuming the plane of the orbiting command module is coplanar with the plane of the LM trajectory relative to the moon at the terminus of the landing, and assuming the LM lands at the nominal time at the nominal landing site in a normal erect attitude, then at the instant of landing -- and at that instant only -- the LM body coordinates (B), the descent guidance coordinates (G), and the platform coordinates (P) will have collinear X-axes, parallel Y-axes, and parallel Z-axes. The X-axes will be vertical pointing up, the Z-axes will be horizontal lying in the plane of the two trajectories pointing forward, and the Y-axes will complete right-hand triads pointing normal to the trajectory plane to the right. The origin of the platform coordinates is always at the center of the moon. At the landing instant in the nominal case, the origin of the guidance frame will be on the X-axis of the platform frame at the intersection with the surface, and the origin of the body frame will be on the X-axis of the platform frame a few feet above the surface at the height of the landing radar antenna. At any other time, the origin of the guidance frame is at the currently computed landing site with the X-axis along the local vertical so that the guidance and platform X-axes are neither collinear nor parallel.

5.3.4.3 State-Vector Update Routine

The State-Vector Update Routine computes the current state of the vehicle (\underline{r}_P , \underline{v}_P) using output data from the IMU and the landing radar (LR). The state-vector updatings occur at 2-second intervals during the landing maneuver at the times that the PIPA outputs are processed. The state-vector updatings are performed in the platform frame (subscript P).



- 1) Range Beam 4 is in the $X_A - Z_A$ plane at an angle ξ from the $-X_A$ axis
- 2) LR velocity data to LGC is in Antenna Coordinate System
- 3) LR Antenna Coordinate System is related to the PGNCS Navigation Base by a specified set of Euler angles for each of the two LR antenna positions (α, β)

Fig. 3.4.2-2 LR Antenna Coordinate System and Beam Configuration

If the Data-Good and Reasonableness tests are satisfied and the astronaut allows updating, range measurements (altitude) from the LR are used to update the state at 2-second intervals. When the estimated LM speed has dropped below VUP, the LR Velocity-Read flag (READVEL) is set. If the Data-Good and Reasonableness tests are satisfied and the astronaut allows updating, velocity-component measurements are used in addition to the range measurements. The three velocity-components are processed individually at 2-second intervals immediately after the altitude measurements. The time between consecutive processings of the same velocity component is 6 seconds. It is important to note that the LR velocity-component measurements, in general, will not be taken at precisely the times that the PIPA outputs are processed. This requires that the difference between the raw measurement (\tilde{q}) and its estimate (q') at the measurement time be carried over to the next PIPA-processing time for use in the state-vector updating procedure.

The information-flow diagram for the State-Vector Update Routine is given in Fig. 3.4.3-1. The sequence of operations involved in the procedure will next be described.

The first step in the updating cycle is to check the No-LR Read flag (NOLRREAD). This flag is set at unity during the period when the LR antenna is switched from Position-One to Position-Two. It is reset to zero after the Position-Two Discrete is obtained, or by a PROCEED response to a #523 alarm.

If NOLRREAD is unity, the PIPA data are next read as indicated in Fig. 3.4.3-1. If, on the other hand, NOLRREAD is zero, then the LR Range Read Subroutine is entered immediately to read and check LR range data at essentially the same time that the PIPA data are read. The range and PIPA read operations in this latter case proceed essentially in parallel, as shown in Fig. 3.4.3-1.

If the LR Range Read Subroutine is entered, LR range data are read and data-good checks are made. A detailed description of this subroutine is given in Section 5.3.4.4. Both LR range and IMU gimbal-angle data are stored at this time. If the range data satisfy the tests set up in the routine, then the range measurement flag (RANGEDATA) is set at a unity value.

* PRIORITY = 20

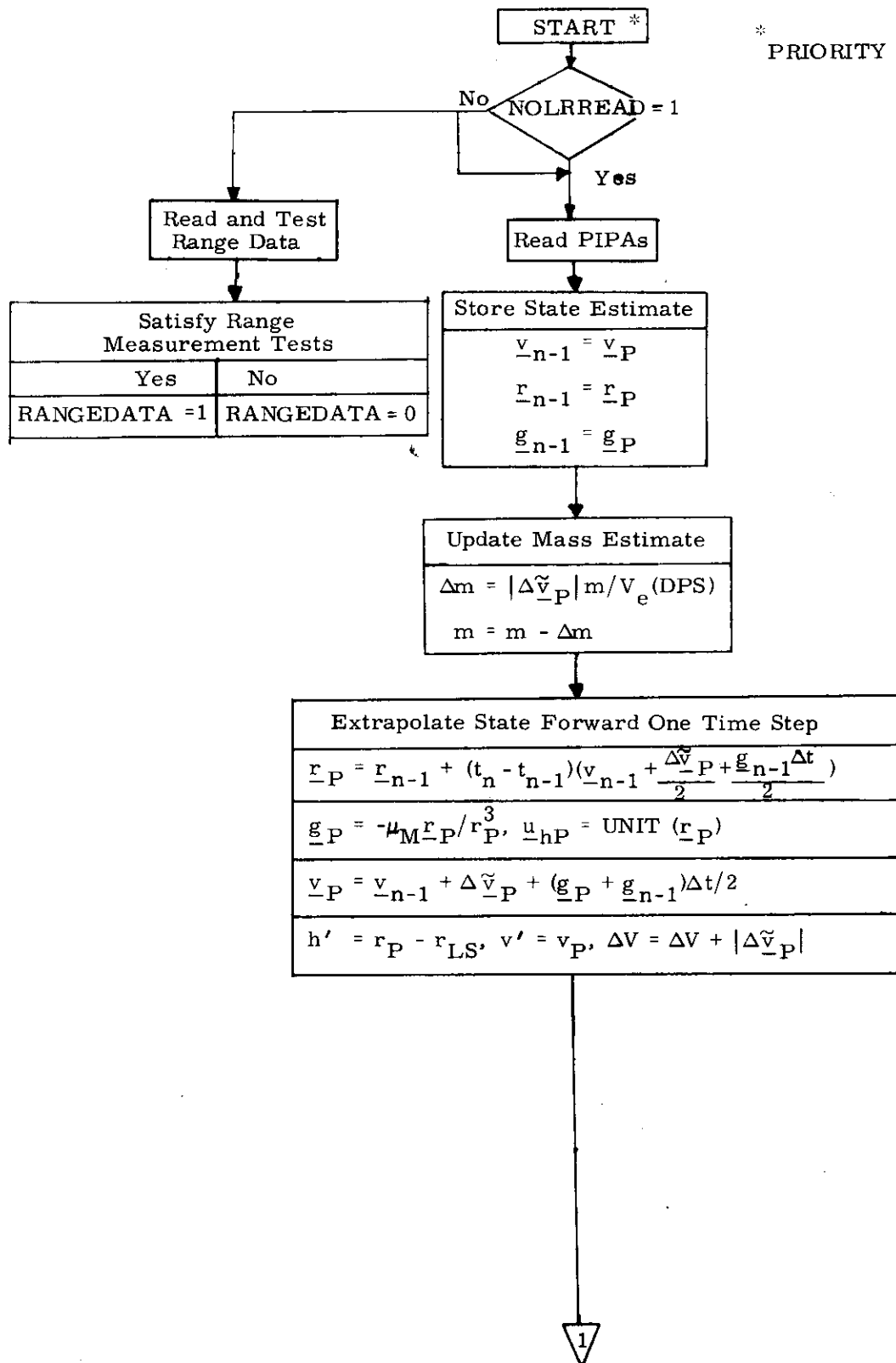


Fig. 3.4.3-1 State-Vector-Update Routine
(Page 1 of 4)

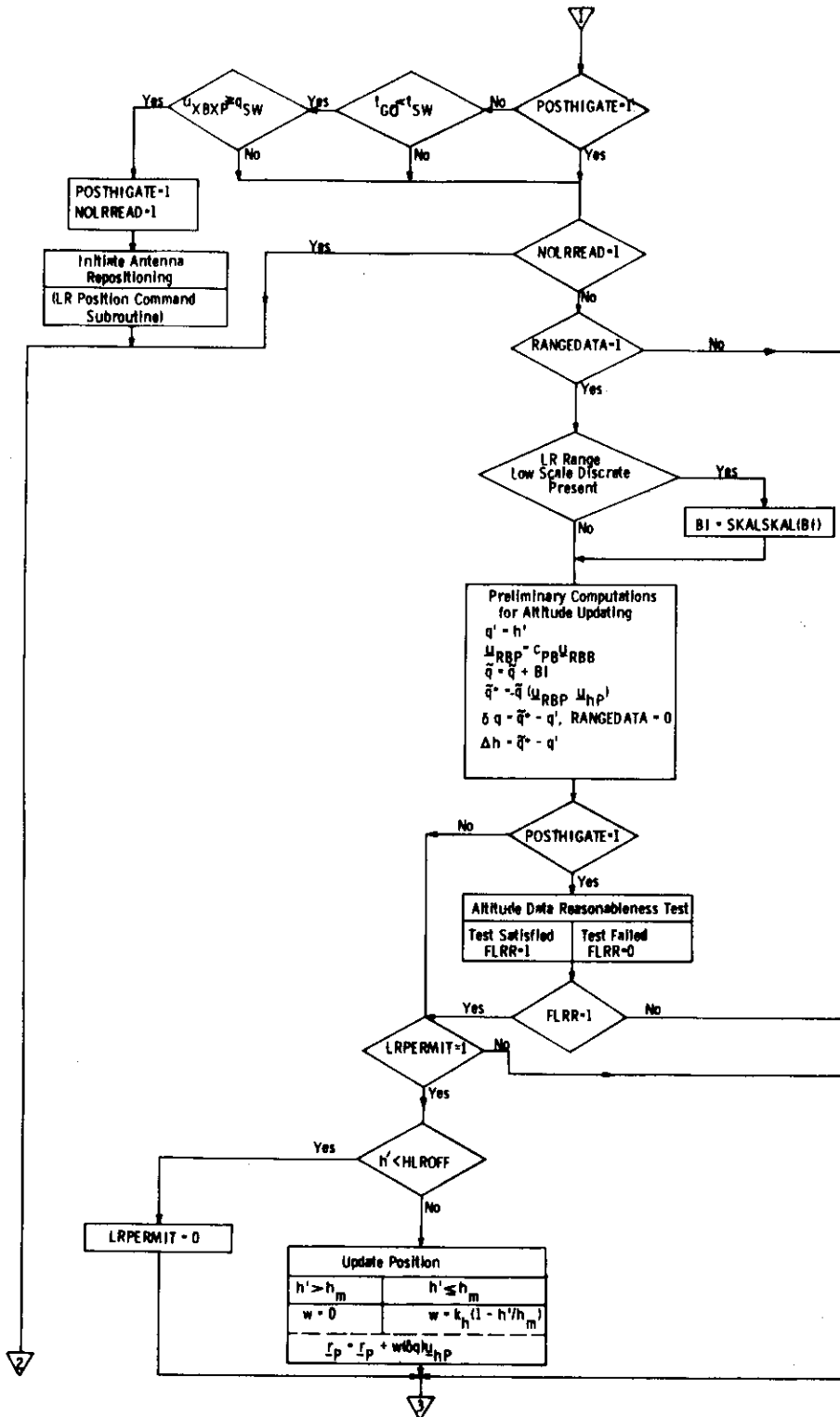


Fig. 3.4.3-1 State-Vector-Update Routine
(Page 2 of 4)

5.3-60

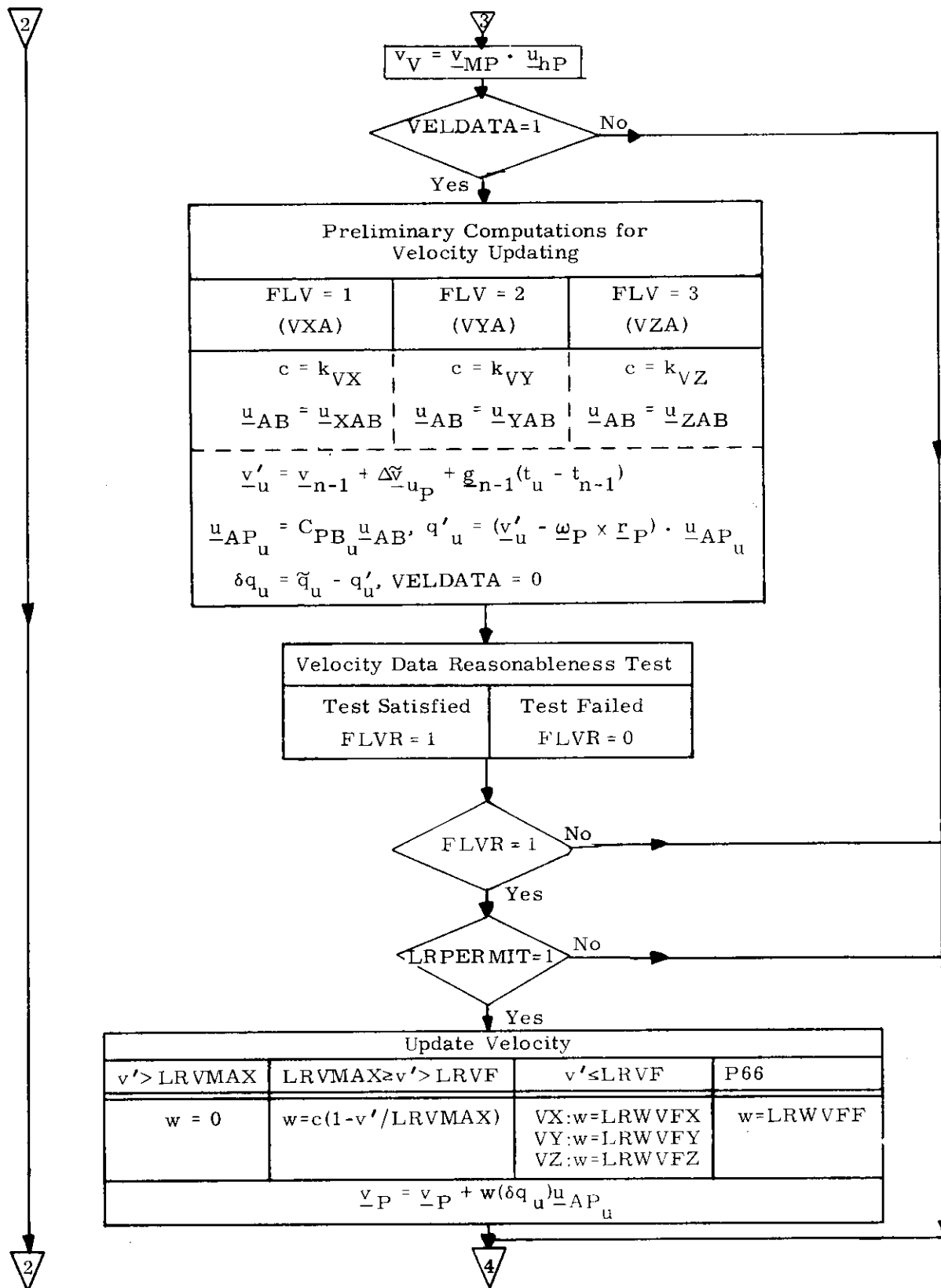
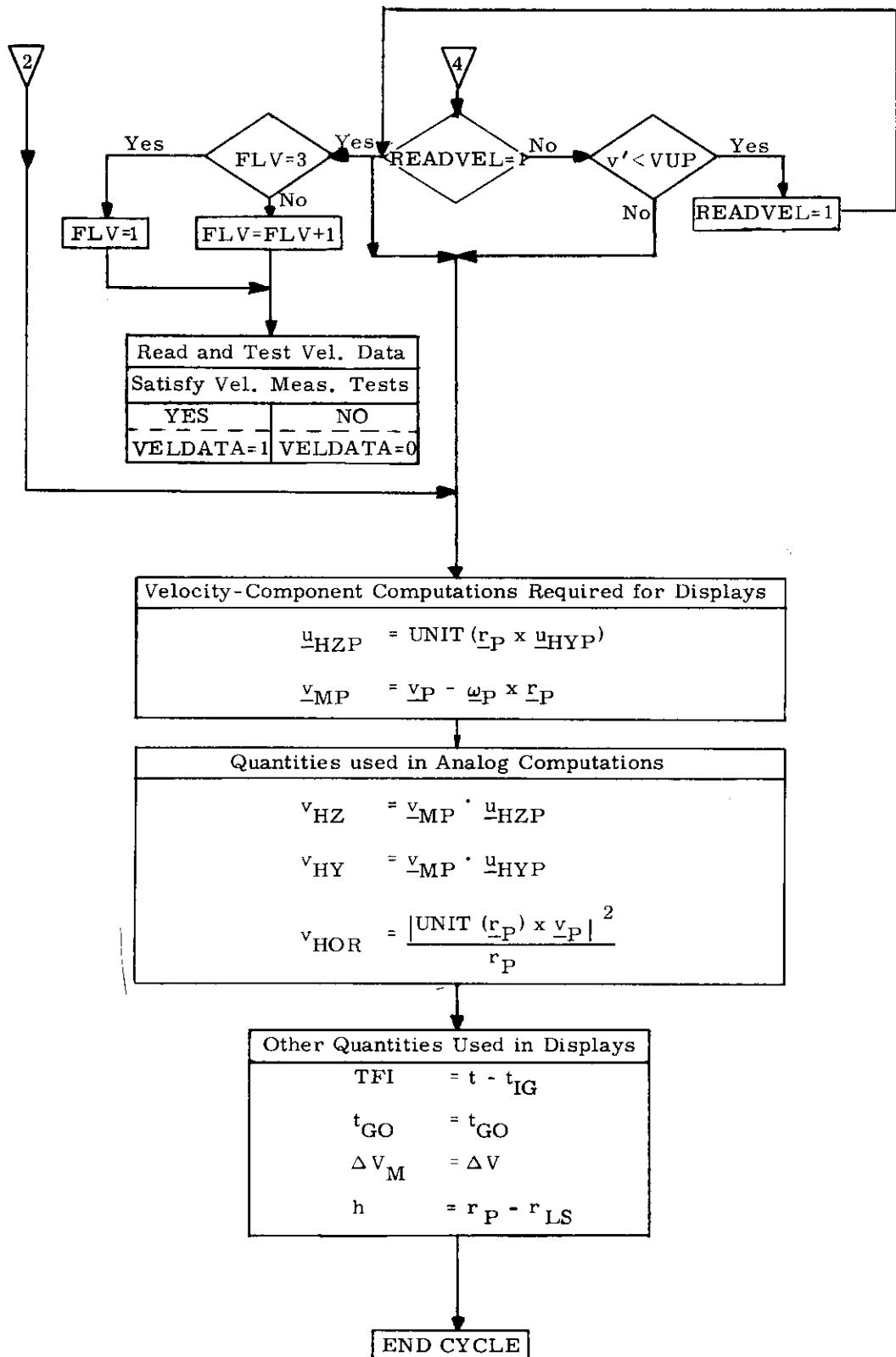


Fig. 3.4.3-1 State-Vector-Update Routine
(Page 3 of 4)

5.3-61



* Computed only until ignition authorized, or thrust fail.

Fig. 3.4.3-1 State-Vector-Update Routine (Page 4 of 4)

5.3-62

The next step in the updating cycle after the reading of PIPA data is to store the vehicle velocity and position estimates in platform coordinates for the previous cycle through the State Vector Update Routine (\underline{v}_{n-1} and \underline{r}_{n-1}).

Using the PIPA output data, an up-to-date estimate of LM mass (m) is next obtained as shown in Fig. 3.4.3-1. The change in estimated mass, as can be seen, is a function of the present vehicle mass (m), the measured PIPA velocity change $|\Delta \underline{\tilde{v}}_P|$, and a prestored constant value of exhaust velocity, V_e (DPS), for the DPS during the powered landing maneuver.

Next, the state of the LM from the previous cycle (\underline{r}_{n-1} and \underline{v}_{n-1}) is extrapolated forward to the present time using the PIPA output data ($\Delta \underline{\tilde{v}}_P$) and the previous-cycle estimate of the LM acceleration due to the lunar gravitational field (\underline{g}_{n-1}). The extrapolation interval (which is 2 seconds) is denoted by Δt . Extrapolated values of LM altitude (h') and speed (v') are next obtained, using the value of radius of the moon at the initially-selected site (r_{LS}) in the h' computation. The accumulated velocity increment for the powered landing maneuver (ΔV) is also computed at this time.

A test is next made to see if the Post-High-Gate flag (POSTHIGATE) has been set at unity, signifying the start of the visibility phase. At the same time that this flag is set at unity, the procedure for switching the LR antenna from Position-One to Position-Two is begun. The No-LR-Read flag (NOLRREAD) is also set at unity at this time to inhibit the reading of LR data during the interval that the antenna position is being switched. (The NOLRREAD flag is reset to zero in the LR Position Command Subroutine described in Section 5.3.4.4, when a Position-Two discrete is detected; or by a PROCEED response to a 523 alarm.)

The Post-High-Gate flag (POSTHIGATE) is set on the basis of the computed time-to-go (t_{GO}) and the projection of the vehicle's X-axis unit vector along the platform-frame X-axis (u_{XBXP}). The test thresholds t_{SW} and q_{SW} are stored in erasable memory. It should be noted that both the time-to-go and X-axis orientation tests must be satisfied before the Post-High-Gate flag can be set.

Proceeding forward in the State-Vector Update Routine, a check is next made to see if the Range-Measurement flag (RANGEDATA) is set at unity. This flag is set in the LR Data-Read Routine (Section 5.3.4.4.) after Range Data-Good discrettes are observed on three consecutive computation cycles. As soon as the check indicates that no Range Data-Good discrete is present, then the range measurement flag (RANGEDATA) is reset to zero. If RANGEDATA is unity, the altitude updating procedure is begun; if RANGEDATA is zero, the updating is bypassed.

The first step in the altitude-updating procedure, as shown in Fig. 3.4.3-1, is to obtain an estimate of LM altitude at the measurement time. For a range (altitude) measurement, the effective measurement time is made to occur essentially at PIPA-output processing time by proper choice of LR interrogation time. Hence, the estimated altitude (q') is simply obtained by subtracting the lunar radius at the initial site (r_{LS}) from the magnitude of the LM position vector $|\underline{r}_P|$.

The actual raw measurement from the radar (\tilde{q}) is the slant range to the ground along the direction of the range beam (\underline{u}_{RBB}). In order to obtain a meaningful comparison between \tilde{q} and q' (as previously defined), it is necessary to use the component of \tilde{q} projected along the local vertical (i. e. measured altitude) rather than the measured slant range along the beam.[†] This first requires that the orientation of the range beam at the measurement time in the platform frame (\underline{u}_{RBP}) be determined from the known range-beam orientation in vehicle (Nav. Base.) coordinates (\underline{u}_{RBB}), using the vehicle-to-platform transformation matrix (C_{PB}). It should be noted that the orientation of the range beam (\underline{u}_{RBB}) is changed prior to the start of the visibility phase. The altitude measurement used in the updating procedure (\tilde{q}^*) is obtained by scaling-down the raw range measurement by the projection of the range beam (\underline{u}_{RBP}) along the computed local vertical at the measurement time (\underline{u}_{hP}). At this point the range-measurement flag (RANGEDATA) is reset to zero.

If the vehicle is past the High-Gate point (POSTHIGATE=1), then the LR-Data Reasonableness-Test Routine is next entered to determine if raw altitude measurement data appear to be satisfactory. Prior to High Gate (POSTHIGATE=0) the Reasonableness-Test Routine is bypassed. The tests, which are described in detail in Section 5.3.4.5, are based on the measurement difference (δq) and the estimated altitude (q').

[†]It should be noted that the LR is located below the nav. base by about 4 meters. In order to obtain compatible LR and State-Vector Update Routine displays of altitude during the latter part of the visibility phase, this displacement is not incorporated into the state-vector update.

At this time, the LR-Updating Permit flag (LRPERMIT) is checked. If this flag has not been set to unity by the astronaut, the altitude updating-procedure is bypassed. If LRPERMIT = 1, calculated altitude is compared with an altitude constant. If altitude is below the constant, there is no position update, and LRPERMIT is set to zero.

The difference between the altitude measurement at the measurement time (\tilde{q}^*) and its extrapolated estimate (q') is used to update the LM position vector at the PIPA-output processing time. As can be seen in Fig. 3.4.3-1, the weighted measurement difference is used to update the position component of the state vector (\underline{r}_P) along the direction estimated for the altitude measurement (\underline{u}_{hP}). The altitude-measurement weighting function (w), as can be seen, is stored as a linear function of the estimated LM altitude (h'). The coefficients h_m and k_h , will be stored in erasable memory.

A check is next made to see if the Velocity Measurement flag (VELDATA) is set at unity. This flag is set at unity in the LR Velocity-Data Read Subroutine (Section 5.3.4.4.) after Velocity Data-Good discretes are observed on three consecutive computation cycles through the subroutine. As soon as the test indicates that no Velocity Data-Good discrete is present, then the measurement flag (VELDATA) is reset to zero. If VELDATA is unity, then the preliminary computations for velocity updating are begun; if the flag is zero, the entire velocity updating process is by-passed.

The preliminary computations for velocity updating involve first the selection of the coefficient (c) to be used in the linear weighting function for the velocity component of interest. This depends on the particular component used on the current cycle, which is identified by the velocity-component flag (FLV) from the LR Velocity-Data Read Subroutine (Section 5.3.4.4.). The numerical values for the coefficients k_{VX} , k_{VY} , and k_{VZ} , will be stored in erasable memory. Next, the unit vector for the velocity component being processed (\underline{u}_{AB}) is selected. The unit vectors for the different velocity components as shown in Fig. 3.4.3-1, are referred to as \underline{u}_{XAB} , \underline{u}_{YAB} , and \underline{u}_{ZAB} . (See Fig. 3.4.4-5.)

An estimate of vehicle velocity at the measurement time (\underline{v}'_u) is next obtained. Because of the fact that the velocity measurements are not taken at the normal PIPA-processing times, it is necessary to read the PIPA-output data also at the measurement time (t_u) and store the data for use on the next computation cycle through the State-Vector Update Routine. The velocity-measurement estimate (\underline{v}'_u), as indicated in Fig. 3.4.3-1, is obtained by adding the velocity change measured by the PIPA-output data from the preceding updating time ($\Delta \tilde{\underline{v}}_{uP}$) to the preceding-cycle updated velocity (\underline{v}_{n-1}). The effect of the lunar gravitational acceleration is included in the computation of \underline{v}'_u by using the previous-cycle value (\underline{g}_{n-1}) over the interval from the preceding updating time (t_{n-1}) to the measurement time (t_u).

5.3-65

In addition to the PIPA-output data ($\Delta \tilde{v}_{uP}$), the unit vectors defining the attitude of the vehicle at the measurement time (C_{PB_u}) are stored for use in the updating of the state vector after the velocity measurement. The orientation of the antenna axis for the velocity component being processed (u_{AP_u}) is obtained in platform coordinates from u_{AB} and the transformation matrix C_{PB_u} . Using v'_u and u_{AP_u} , an estimate is now obtained of the magnitude of the velocity component being processed (q'_u). The difference between the raw measurement and the estimate (δq_u) is computed at this time and the velocity measurement flag (VELDATA) is reset to zero.

The LR-Data Reasonableness-Test Routine is next entered to check the velocity measurement data. If the LR fails this test (FLVR = 0) or LR velocity updating is inhibited by the astronaut (LRPERMIT = 0), then the remainder of the velocity-updating procedure is bypassed. If, on the other hand, the data appear to be satisfactory (FLVR = 1) and the astronaut has allowed updating (LRPERMIT = 1), the velocity-updating procedure is continued. In effect, both the Reasonableness-Test flag (FLVR) and the LR-Updating Permit flag (LRPERMIT) must be set to unity to permit velocity updating by the LR. The flag LRPERMIT is set by the astronaut.

The updated estimate of vehicle velocity (v_p) is obtained, as shown in Fig. 3.4.3-1, by first computing the weighting function for the difference between q'_u and the raw measurement \tilde{q}_u . When the estimated vehicle speed (v') is smaller than a preselected threshold LRVMAX (erasable memory), then the velocity-component weighting function (w) is computed as a linear function of the estimated vehicle speed (v'), as shown in Fig. 3.4.3-1. When the estimated speed is below a second threshold LRVF (erasable memory), whose numerical value is smaller than LRVMAX, then a constant weighting function is used for each of the velocity components. These constant weighting functions LRWVFX, LRWVFY, and LRWVFZ are stored in erasable memory. Finally, when P66, the final descent program is entered a constant weighting function LRWVFF (erasable memory) is used for each velocity component. The weighted difference $w(\tilde{q}_u - q'_u)$ is then used to update the LM velocity vector at the PIPA output processing time along the direction of the measured velocity component (u_{AP_u}).

At this point in the routine a check is made to see if the LR Velocity-Read flag (READVEL) has been set at unity, indicating that velocity-data can be read. If the flag has not been set, then a test is made on estimated vehicle speed (v') to determine whether READVEL should be set to unity on the current cycle. The test threshold (VUP), which is in fixed memory, is currently set at 6000 ft/sec.

5. 3-66

Revised LUMINARY 1C
 Added GSOP # R-567 PCR # 988 Rev. 8 Date 3/70

If the Velocity-Read flag (READVEL) is unity, on the other hand, then the velocity-component selector flag (FLV) is set, as shown on Fig. 3.4.3-1. The procedures for reading and testing velocity data are next initiated. The details of the LR Velocity-Data Read Subroutine are described in Section 5.3.4.4. It should be noted that if the velocity measurement is satisfactory, it is stored for use on the next computation cycle along with the IMU gimbal-angle and PIPA-output data at the mean measurement time (t_u). This measurement time (t_u) is also stored. It should also be noted that the velocity-component reading and testing takes place at the same time that the subsequent State-Vector Update Routine computations are being accomplished.

The altitude and velocity updating sections of the State-Vector Update Routine are now completed. The remainder of the routine is primarily concerned with display computations.

Certain components of LM velocity are required during the landing maneuver to provide digital displays on the DSKY and to drive analog-type vehicle meters. The required computations for the digital displays are shown next on the State-Vector Update Routine information-flow diagram of Fig. 3.4.3-1.

The first display quantity computed is the vertical velocity (v_v), which is obtained by taking the component of LM velocity relative to the moon (v_{MP}) along the local vertical (u_{hP}). The components of LM velocity are now computed along a pair of essentially local-horizontal axes (u_{HYP} and u_{HZP}). The azimuth orientation of the Y and Z axes about r_P (or u_{hP}) is such that the Y-axis is normal to the CSM orbital plane. The Y-axis unit vector (u_{HYP}) is computed as a part of average-G initialization.

Other quantities required for the displays are next computed, as indicated in Fig. 3.4.3-1. These quantities are the current-phase time-to-go (t_{GO}), the accumulated landing maneuver velocity increment (ΔV_M), and the current estimate of vehicle altitude relative to that of the initial site (h).

It should be noted that the display computations described here are essentially for the PIPA-processing times. The procedure for computing display quantities at times in between the normal PIPA-processing time is described in Section 5.3.4.8 on Landing Maneuver Display Computations. It should also be mentioned that the computation for cross-range CR is presented in the ignition computations routine; for slant range RANGE in the Guidance and Control Routine (Fig. 3.4.6-2 page 3 of 12), and for the horizontal velocity digital v_H , and LPD angle θ_{ef} in the Guidance and Control Routine.

5.3.4.4. LR Data Read Routine . -- The LR Data Read Routine actually consists of a number of smaller subroutines which are used at various points in the State Vector Update Routine to read LR data or to perform various other functions required when using the LR. These smaller subroutines are known as the:

1. LR Range Data Read Subroutine
2. LR Position Command Subroutine
3. LR Velocity Data Read Subroutine
4. Radar Beams Orientation Subroutine

The logic associated with these subroutines is given in Figs. 3.4.4-1, 2, 3, 4, and 5. The manner in which these subroutines are used with the State Vector Update Routine is shown in Fig. 3.4.4-1. The dotted-line portions of this figure represent the parts of the State Vector Update Routine which are given in greater detail in Fig. 3.4.3-1. It should be noted that Fig. 3.4.4-1 shows essentially the same operational flow as that given in Fig. 3.4.3-1 for the State Vector Update Routine, except that greater emphasis is given in Fig. 3.4.4-1 on the logic used to process the LR, while Fig. 3.4.3-1 is actually an information flow diagram. Since there is a complicated interface between the LR Data Read Routine and the State Vector Update Routine, it is difficult to separate the two in any convenient manner in Section 4. Consequently, all of the functions performed by both of the above routines are considered to be the functions of just one routine in Section 4, which is arbitrarily denoted as the Descent State Vector Update Routine (R-12).

In Fig. 3.4.4-1, use is made of the following flags when processing the LR data. The first eight flags are reset at the beginning of the lunar landing maneuver.

- 1) LRBYPASS Set when it is desired to update the state vector only with PIPA data such as during P-12, P-70, and P-71.

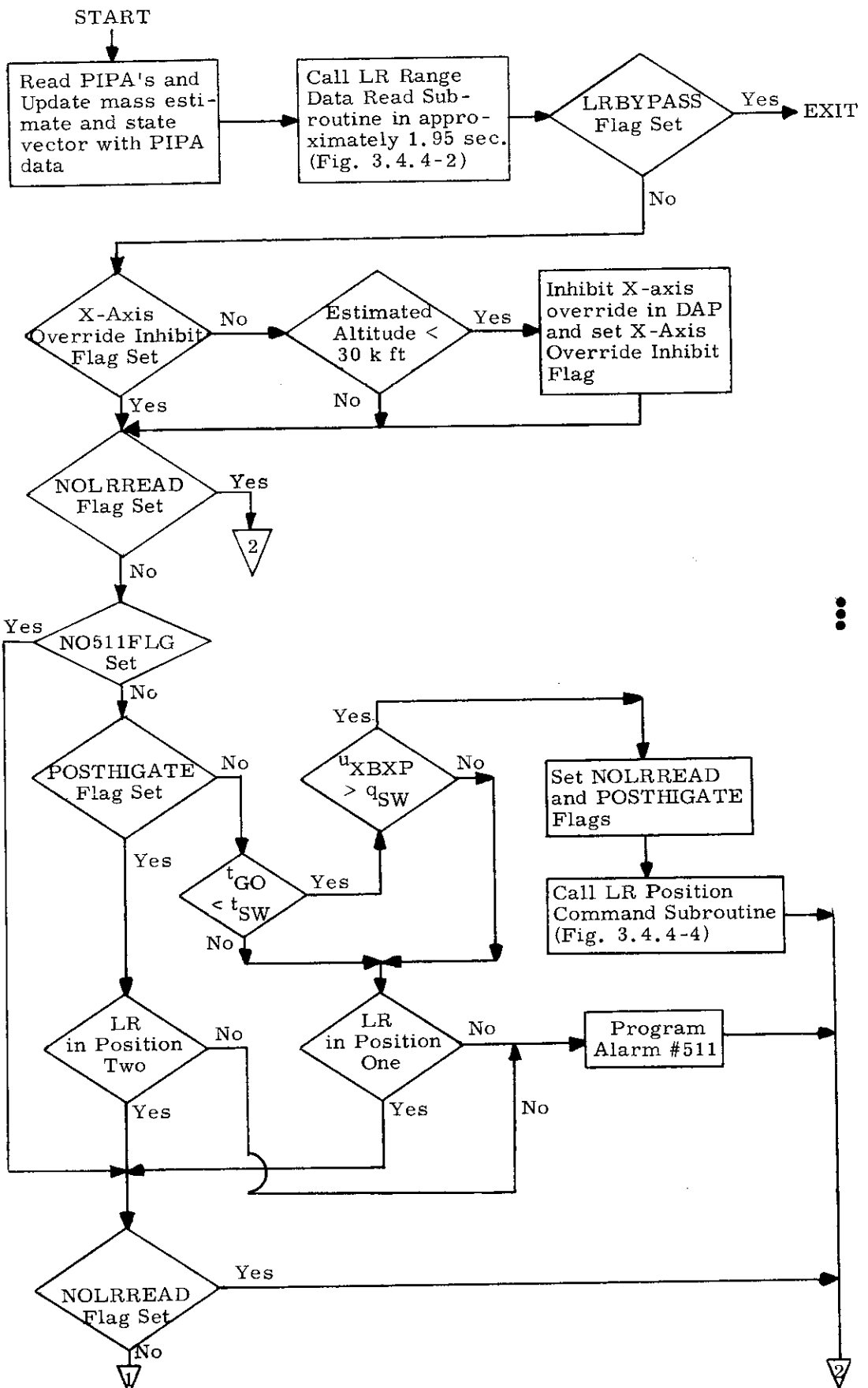


Fig. 3.4.4-1 LR Data Read Routine (general flow diagram)
(Page 1 of 2)

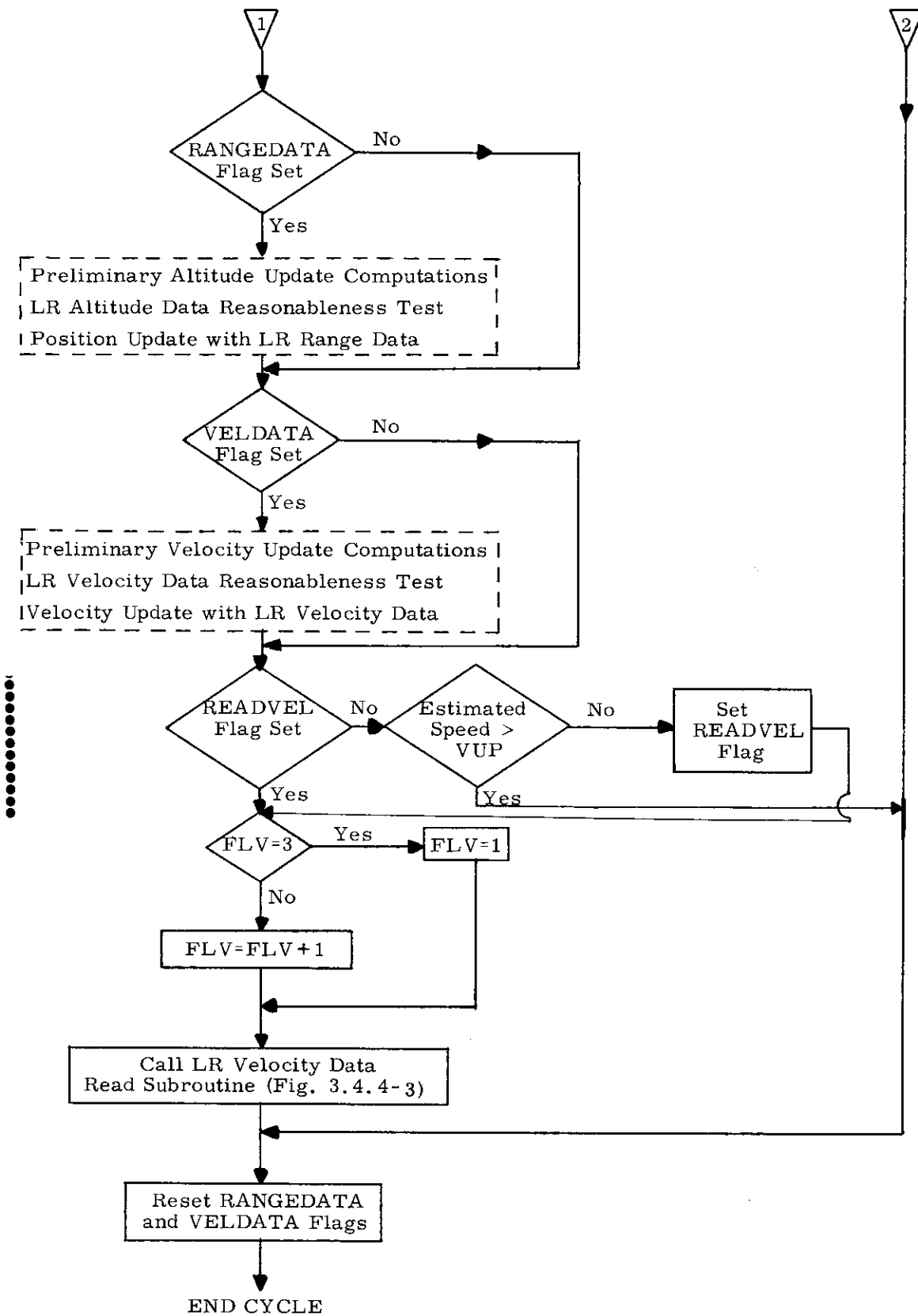


Fig. 3.4.4-1 LR Data Read Routine (general flow diagram)
(Page 2 of 2)

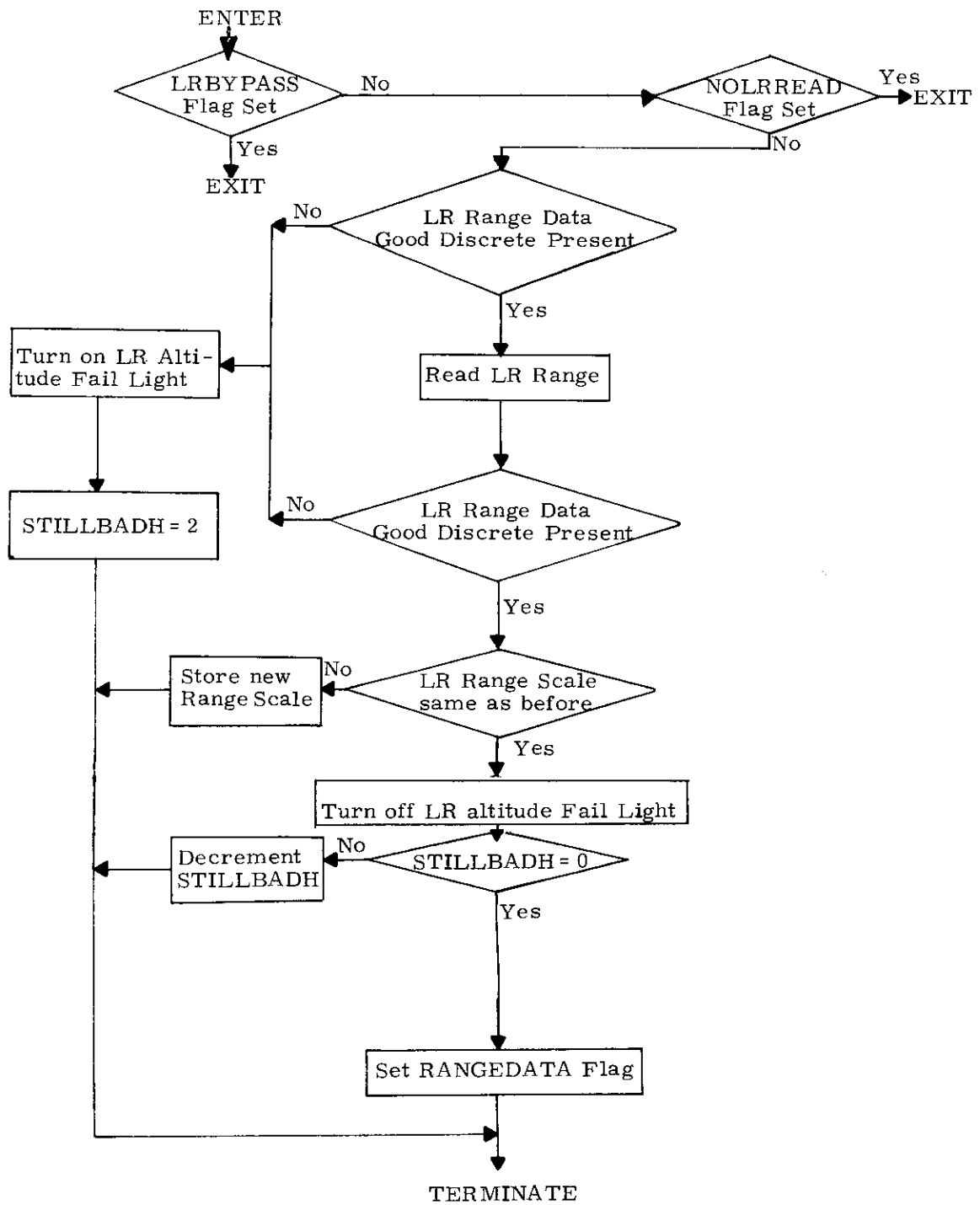


Figure 3.4.4-2 LR Data Read Routine
(LR Range Data Read Subroutine)

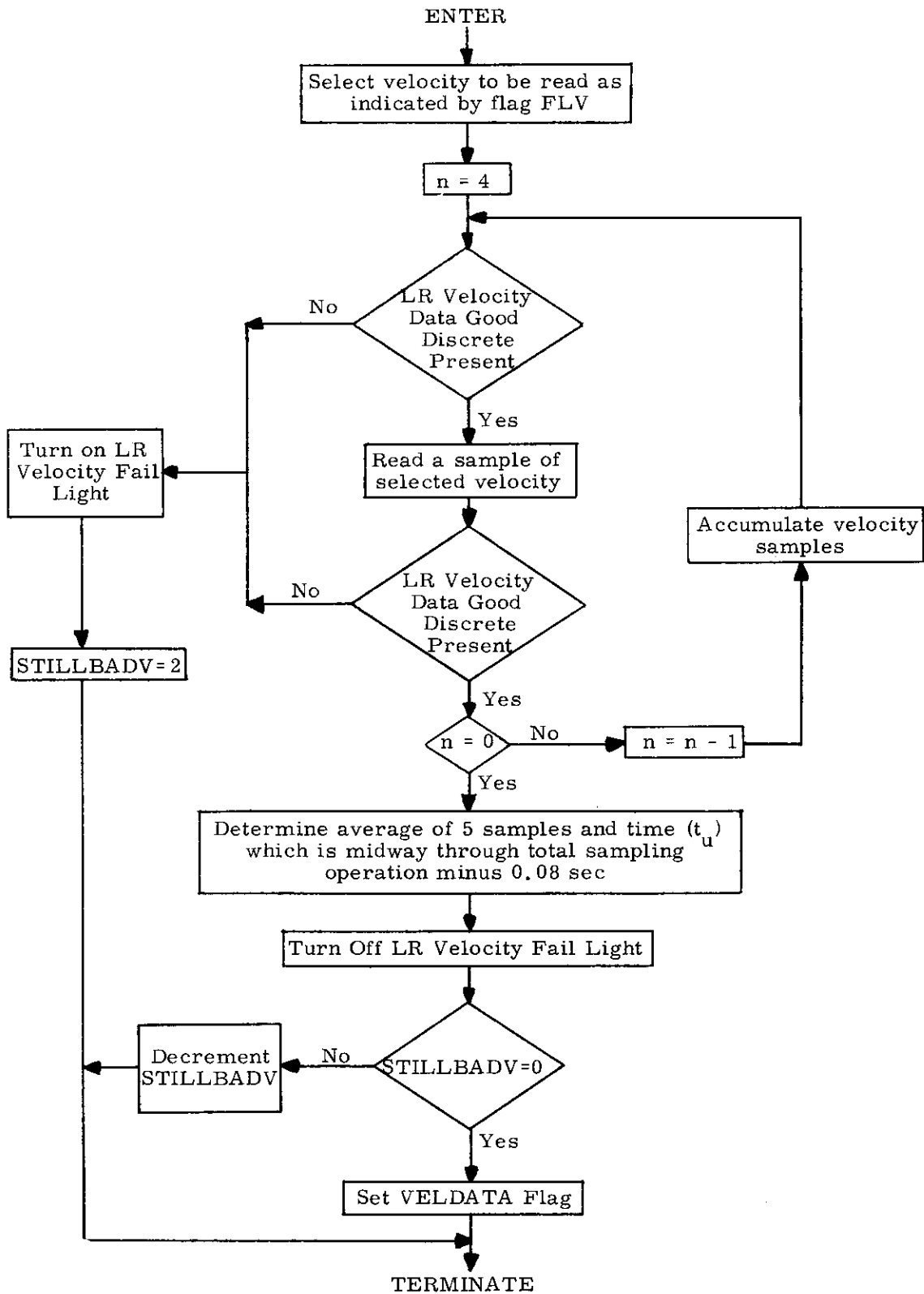


Fig. 3.4.4-3 LR Data Read Routine
(LR Velocity Data Read Subroutine)

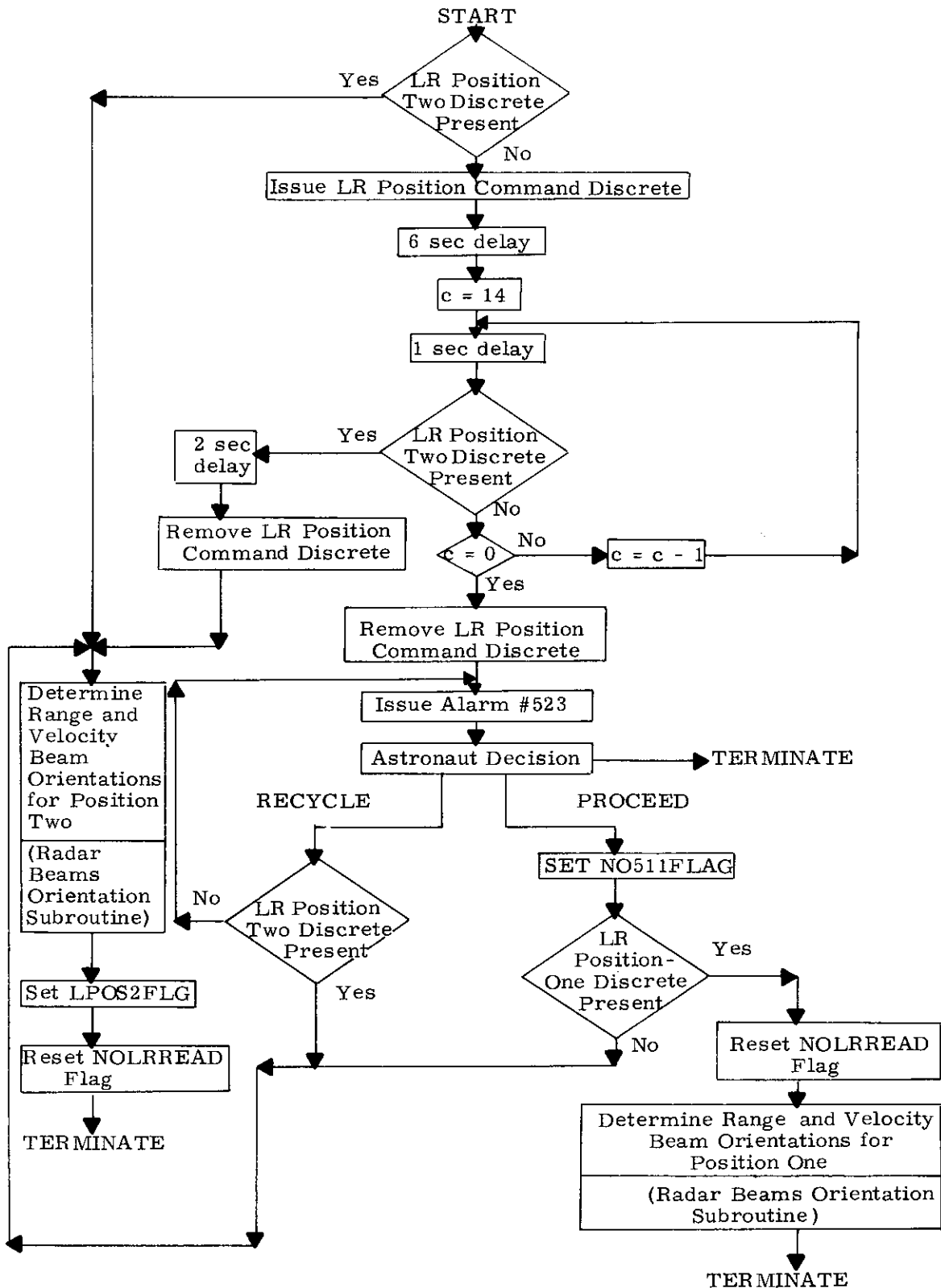


Fig. 3.4.4-4 LR Data Read Routine
(LR Position Command Subroutine)

5.3-73

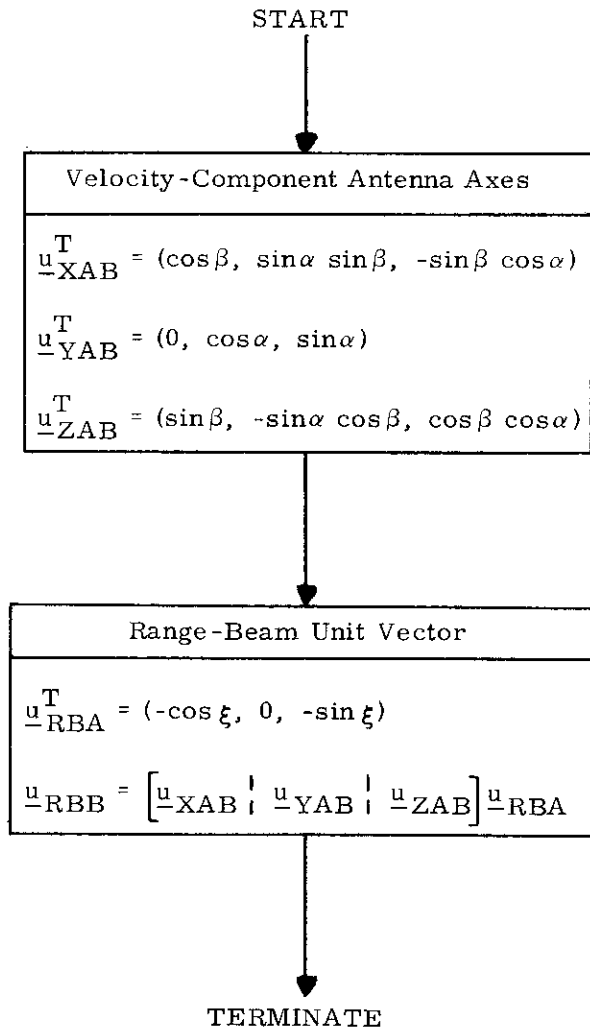


Fig. 3.4.4-5 LR Data Read Routine
(Radar Beams Orientation Subroutine)

- 2) NO511FLG Set when PROCEED is keyed in by astronaut after alarm #523. Eliminates antenna position checks thereafter in R12.

- 3) NOLRREAD Set during the time when the LR Position Command Subroutine is driving the LR antenna from Position One to Two and reset when Position Two has been obtained, or when astronaut keys in a PROCEED to alarm #523. This flag prevents LR data from being read when set.

- 4) POSTHIGATE Set when $t_{GO} < t_{SW}$ and $u_{XBXP} > q_{SW}$. When not set the computer checks to see if the LR antenna is in Position One. When set, the computer checks to see if the LR antenna is in Position Two.

- 5) READVEL Set when the estimated speed drops below VUP which is presently 6000 ft/sec. This flag permits LR velocity data to be read when set.

- 6) X-Axis Override Inhibit Set when the estimated altitude drops below 30,000 feet. When set this flag denotes that X-axis override has been inhibited by R12.

- 7) RANGEDATA Set when the LR Range Data Read Subroutine successfully obtains a range measurement from the LR.

- 8) VELDATA Set when the LR Velocity Data Read Subroutine successfully obtains a velocity measurement from the LR.

- 9) FLV Indicates which LR velocity component is to be read from the LR. This flag is set to 1, 2, and 3 for the v_X , v_Y , and v_Z velocity components, respectively.

- 10) LPOS2 FLG When set LR is in position 2; use beam orientations for position 2. When reset LR is in position 1; use position one data.

When the Average-G Routine is initiated at $t_{IG} - 30$ seconds, the logic shown in Fig. 3.4.4-1 is entered once every two seconds. It should be noted that the LR Range Data Read, the LR Velocity Data Read, and the LR Position Command Subroutines are merely called by the mainline logic flow of Fig. 3.4.4-1 and that this logic flow does not wait for completion of these subroutines.

The LR Range Data Read Subroutine starts reading the LR range each time it is called (i. e. every two seconds). However the range data is not used to update the state vector until the LRPERMIT flag has been set by the astronaut as described in Section 5.3.4.5. Each range measurement is the result of a single 80.001 millisecond duration count of a frequency in the LR. The time of the LR range measurement is made to be the same as that for the next PIPA reading. In Fig. 3.4.4-2 it is seen that the LR Range Data Good discrete is checked before and after the range reading. This discrete is sent to the LGC by the LR when the radar tracking loops associated with the LR range beam and the two rear LR velocity beams are locked-on and tracking satisfactorily. Absence of this discrete causes the LR Altitude Fail Light to be turned on. If the range scale status, as indicated by the LR Range Low Scale discrete from the LR, has changed since the last range reading the present range reading is not used, since a delay of at least one second is required after the LR scale change to allow settling of the LR range tracker. Note that the control exercised by the parameter STILLBADH is such as to delay use of LR range data for four seconds after detection of the LR Range Data Good discrete since large errors may exist in the LR data during this period of time after lock-on of the radar tracking loops. The parameter STILLBADH is initially set to 2 in the landing program. If a successful range reading is obtained, it is seen in Fig. 3.4.4-2 that the RANGEDATA flag is set to indicate that a successful range reading was obtained.

Proceeding through the LR Data Read Routine (Fig. 3.4.4-1), a check is made to see if the X-axis override should be inhibited in the DAP. The NOLRREAD flag is next checked and, if it has been set, the radar updating operations in the routine are bypassed. At this point, the NO511FLG is checked. If NO511FLG has been set, then the antenna position checks in the routine are bypassed as shown in Fig. 3.4.4-1.

If NO511FLG has not been set, a check is made to see if the time-to-go (t_{GO}) is $< t_{SW}$ and $u_{XBXP} > q_{SW}$. In Section 5.3.4.3, it is seen that u_{XBXP} is the component of the vehicle X-axis along the stable-member X-axis and is approximately the cosine of the pitch angle. The quantities t_{SW} and q_{SW} are part of the pre-launch erasable data load. If t_{GO} is not $< t_{SW}$ or u_{XBXP} is not $> q_{SW}$, the routine will next check to see if the LR antenna is in Position One. This position is indicated to the LGC by the LR with the LR Position One discrete. If the antenna is found to be in Position One, it seen in Fig. 3.4.4-1 that checks are made to insure that the NOLRREAD is reset and the RANGEDATA flag is set before performing the position update with LR range data. Absence of the Position One discrete, however, will cause an alarm to

be issued to the astronaut, indicating that he should place the antenna in Position One by using the separate antenna position control provided with the LR. If the radar is stuck in Position Two the astronaut can force the computer to accept Position Two data by keying in V59E, which insures that the radar is in Position Two, and sets $t_{sw} > t_{go}$ and $q_{sw} < u_{XBXP}$ to insure that certain steps will be performed by the LR position command subroutine before Position Two data is processed by the LGC.

When the estimated speed drops below VUP in the braking phase, it is seen at the bottom of Fig. 3.4.4-1 that the READVEL flag is set causing the LR Velocity Data Read Subroutine of Fig. 3.4.4-3 to be called on successive operation cycles. Note that the LR velocity reading obtained by the LR Velocity Data Read Subroutine is actually the average of 5 successive readings (samples) of a given component of LR velocity. Each sample represents an 80,001 millisecond duration count in the LR of certain doppler frequencies. The total time required to obtain one sample from the LR is about 100 milliseconds. In reading out each velocity sample from the LR the routine checks before and after each sample to insure that the LR Velocity Data Good discrete is being received from the LR. This discrete signifies that all three velocity tracking loops are locked-on and tracking satisfactorily. The time tag t_u assigned to the average velocity reading corresponds to the midpoint of the 5 samples minus 0.08 seconds to account for lag in the LR velocity tracking loops. Note that the control exercised by the parameter STILLBADV is such as to delay use of LR velocity data for four seconds after detection of the LR Velocity Data Good discrete since large errors may exist in the LR data during this period of time after lock-on of the radar tracking loops. (The parameter STILLBADV is initially set to 2 in the landing program.) If a successful velocity reading is obtained, it is seen in Fig. 3.4.4-3 that the LR Velocity Fail Light is turned off, if previously on, and the VELDATA flag is set to indicate this fact.

It should be noted that the PIPA-output data which are used to obtain the estimated velocity at the center of sampling interval are read at τ_{VMES} seconds after the LR Velocity Data Read Subroutine is initiated. The quantity τ_{VMES} , which is stored in fixed memory, is chosen so that the estimated velocity (v'_u on Page 3 of Fig. 3.4.3-1) corresponds to the time t_u at the midpoint of the velocity-measurement samples.

When $t_{GO} < t_{SW}$ and $u_{XBXP} > q_{SW}$ it is seen in Fig. 3.4.4-1 that the NOLRREAD and POSTHIGATE flags are set, and the LR Position Command Subroutine is used to drive the LR antenna from Position One to Two. The purpose of the NOLRREAD flag is to prevent reading LR data and updating the state vector with this data when the antenna is being driven from Position One to Two. The POSTHIGATE flag indicates which antenna position should be checked by the LGC.

5.3-77

Revised LUMINARY
 Added GSOP # 567 PCR # 895 Rev. 7 Date 11/69

The logic associated with the LR Position Command Subroutine is shown in Fig. 3.4.4-4. Initially, the routine checks to see if the LR Position Two discrete is being received from the LR, signifying that the antenna is already in Position Two. If the discrete is not present, the LR Position Command discrete is issued to the LR, causing its antenna to be driven to Position Two. When Position Two has been successfully obtained, it is seen that the NOLRREAD flag is reset and the unit vectors defining the orientations of the range and velocity beams for Position Two are determined. Seven seconds after the start of the routine, and each second thereafter, the routine checks to see if the LR Position Two discrete is being received, until 21 seconds have expired, at which time, an alarm is issued to the astronaut.

At this point, the astronaut has three options in response to the #523 alarm. He can terminate LR updating operation in R12 by entering V34E, which will set the NOLRREAD flag. He can try to place the antenna in Position Two with the separate antenna position control provided with the LR. Entering V32E in this case will cause checks to be made for the Position-Two discrete. Finally, the astronaut may decide to use the LR after High-Gate, even if the Position-Two discrete input has not appeared at the LGC input channel. In this case, he will key in PROCEED, which will set the N0511FLG flag. If the Position-One discrete is still present, the LR data will be used assuming the LR is still in Position One. If the Position-One discrete is not present, it will be assumed that the LR is in Position Two.

LR Velocity and Range Measurement Data . -- The velocity data obtained from the LR by the LR Data Read Subroutine are with respect to the LR antenna coordinate system of Fig. 3.4.2-2 and are in a form which is described as follows, along with the various data processing steps the LGC performs to transform the data into the Navigation Base Coordinate System.

The velocity data furnished at the LGC interface by the LR comprise three binary data words of the following form:

$$\begin{aligned}
 S_{XA} &= [(f_1 + f_3) / 2 + f_B] \tau_{LR} \\
 S_{YA} &= [(f_1 - f_2) + f_B] \tau_{LR} \\
 S_{ZA} &= [(f_3 - f_2) + f_B] \tau_{LR}
 \end{aligned}
 \tag{3.4.1}$$

where S_{XA} , S_{YA} , and S_{ZA} correspond, respectively, to the velocity components along the $-X_A$, $+Y_A$, and $+Z_A$ antenna axes of Fig. 3.4.2-2. The quantities f_1 , f_2 , and f_3 are the beam doppler frequencies, f_B is the bias frequency used in the LR, and τ_{LR} is the time interval used by the LR when counting the cycles of the above frequencies so as to produce the data words S_{XA} , S_{YA} , and S_{ZA} . The time interval τ_{LR} is 80.001 milliseconds.

In the LGC the velocity along each antenna coordinate axis is computed from the above data words as follows:

$$\begin{aligned}
 v_{XA} &= k_{XA} (S_{XA} - f_B \tau_{LR}) \\
 v_{YA} &= k_{YA} (S_{YA} - f_B \tau_{LR}) \\
 v_{ZA} &= k_{ZA} (S_{ZA} - f_B \tau_{LR})
 \end{aligned}
 \tag{3.4.2}$$

where v_{XA} , v_{YA} , and v_{ZA} are the LR measured velocities along the positive antenna coordinate axes, and k_{XA} , k_{YA} , and k_{ZA} are the corresponding scale factors used to obtain the above velocities in feet per second.

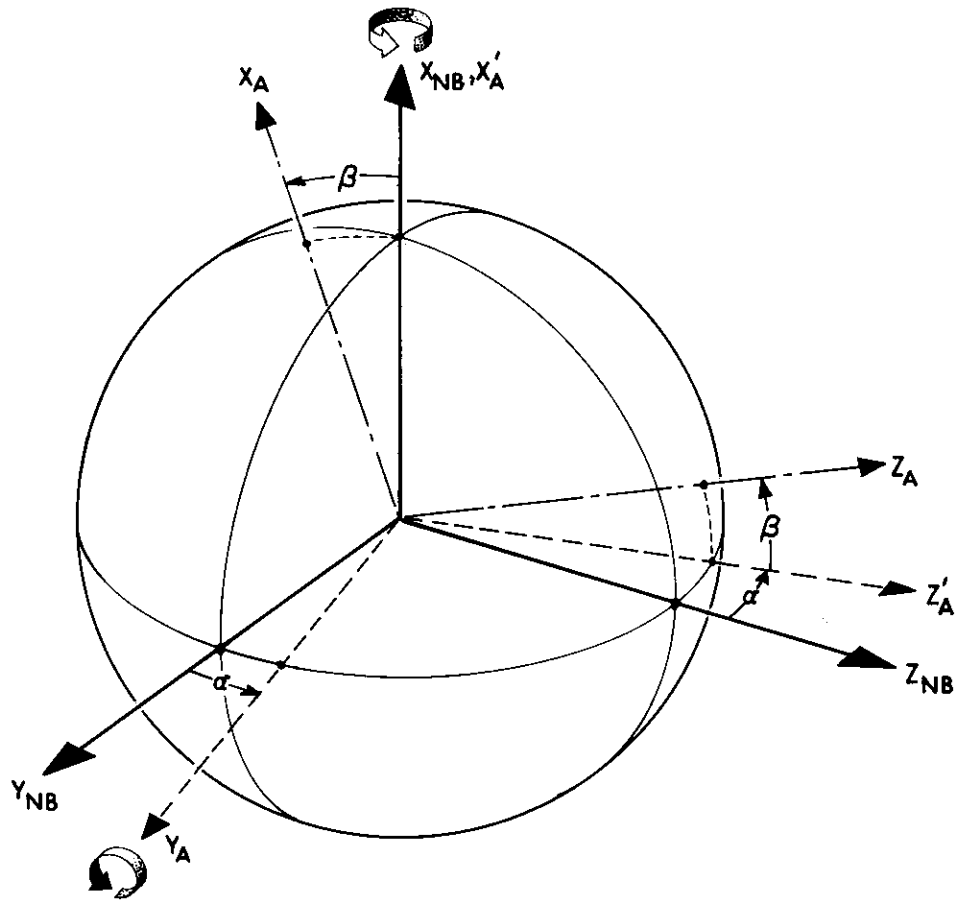
The LR velocity information expressed by v_{XA} , v_{YA} , and v_{ZA} must first be transformed from the antenna coordinate system (subscript A) to the Navigation-Base coordinate frame (subscript B). This requires that the antenna-axes unit vectors be determined in the Navigation-Base coordinate frame.

The required antenna-axes unit vectors (\underline{u}_{XAB} , \underline{u}_{YAB} , \underline{u}_{ZAB}) are computed in the Radar Beams Orientation Subroutine shown in Fig. 3.4.4-5 as functions of the Euler angles α and β . The angles α and β , which are used to define the orientation of the LR antenna axes with respect to the Navigation Base, are shown in Fig. 3.4.4-6. Two sets of Euler angles are stored in the LGC: one set for antenna-coordinate system Position One (α_1, β_1), and the other set for Position Two (α_2, β_2).

In order to properly utilize LR range information it is necessary to determine the range-beam unit vector in Navigation-Base coordinates (\underline{u}_{RBB}). This is also accomplished by the Radar Beams Orientation Routine of Fig. 3.4.4-5 as a function of the previously-mentioned Euler angles (α and β) and the displacement angle ξ which is stored in the LGC. The range beam is in the $X_A - Z_A$ plane of the antenna coordinate system, and is at an angle ξ from the $-X_A$ axis, as shown in Fig. 3.4.2-2.

It should be noted here that the Radar Beams Orientation Subroutine is entered at two different times during the powered landing maneuver. It is entered first prior to the braking phase when the initial conditions and flags for the landing maneuver are set (α_1, β_1 , and ξ are used here). It is entered again near the end of the braking phase when called by the LR Position Command Subroutine (Fig. 3.4.4-4) to determine the Position Two beam orientations (using α_2, β_2 , and ξ).

The range data obtained from the LR by the LR Data Read Routine is that measured by the LR along the range beam shown in Fig. 3.4.2-2. The range data is sent to the LGC from the LR as a binary data word R_{LR} , which represents the count of a certain frequency in the LR during the time interval τ_{LR} . Within the LGC, the range r_{LR} along the range beam is computed as follows:



<u>ANGLE ORDER</u>	<u>AXIS OF ROTATION</u>
α	X_{NB}
β	Y_A

Fig. 3.4.4-6 Angles Defining Orientation of LR Antenna Axes with Respect to the Navigation Base Coordinate System

$$r_{LR} = \begin{cases} k_{LR1} R_{LR} \\ k_{LR2} R_{LR} \end{cases} \quad (3.4.3)$$

where k_{LR1} and k_{LR2} are the bit weights respectively for the long and short range scales in order to obtain r_{LR} in feet. When the LR Range Low Scale discrete is being received from the LR by the LGC, k_{LR2} is used.

A summary of the processing constants required by the LGC for LR operation is given as follows:

f_B	Velocity bias frequency.
τ_{LR}	Counting interval of the landing radar.
k_{XA}	Scale factor to convert $(S_{XA} - f_B \tau_{LR})$ to velocity along the LR antenna coordinate X_A (Fig. 3.4.2-2) in feet per second for the counting interval τ_{LR} .
k_{YA}	Scale factor to convert $(S_{YA} - f_B \tau_{LR})$ to velocity along the LR antenna coordinate Y_A in feet per second for the counting interval τ_{LR} .
k_{ZA}	Scale factor to convert $(S_{ZA} - f_B \tau_{LR})$ to velocity along the LR antenna coordinate Z_A in feet per second for the counting interval τ_{LR} .
$\left. \begin{matrix} \alpha_1 \\ \beta_1 \end{matrix} \right\}$	Respective angles between the LR antenna coordinate system in position One and the navigation base coordinate system (Fig. 3.4.4-6).
$\left. \begin{matrix} \alpha_2 \\ \beta_2 \end{matrix} \right\}$	Respective angles between the LR antenna coordinate system in position Two and the navigation base coordinate system (Fig. 3.4.4-6).
k_{LR1}	Bit weight in feet for long range scale
k_{LR2}	Bit weight in feet for short range scale
ξ	Angle of range beam with respect to the $-X_A$ axis of the LR antenna.

5.3.4.5 LR Data Reasonableness Test Routine. -- The objective of the LR data reasonableness test is to detect and reject degraded LR data caused by cross-coupled side lobe or vibrating structure frequency-tracker lock-up. In these situations the LR Data Good discretely to the PGNCSS will be present, and the LGC would normally process the LR data to update the estimated state vector. The LR data reasonableness test must be applied to both range and velocity LR data with appropriate external control by the astronaut to bypass or permit the LR updating process.

The logic diagram for the LR Data Reasonableness Test Routine is presented in Fig. 3.4.5-1. This routine performs separate tests on derived altitude and velocity data from the LR, as controlled by the LR Data Read Routine of Section 5.3.4.4. The dotted line portions of Fig. 3.4.5-1 are those parts of the State Vector Update Routine (Fig. 3.4.3-1) which are associated with the LR Data Reasonableness Test Routine.

With reference to Fig. 3.4.5-1, the altitude difference δq as determined by the preliminary computation for altitude updating by the State Vector Update Routine, is used as the basic parameter for the LR altitude reasonableness test. The parameter δq is the difference between the altitude as determined from the LR range data (\tilde{q}^* of Fig. 3.4.3-1) and that derived from the estimated state vector (q'). The LR altitude reasonableness test shown in Fig. 3.4.5-1 is a linear function of the estimated current altitude q' . The constant DELQFIX is stored in erasable memory. If the altitude data test is passed, the state vector will be updated with the LR range data if the LRPERMIT flag controlled by the astronaut is set. Note that the altitude reasonableness test is not made until after the POSTHIGATE flag is set in Fig. 3.4.4-1. It should be noted that the LRPERMIT flag is initially reset automatically in the landing program so that no LR updating can start until commanded to do so by the astronaut.

If the altitude difference parameter δq exceeds the limits set by the altitude data test, the alarm logic shown in Fig. 3.4.5-1 is designed to flash the LR Altitude Fail Light if two or more of the last four consecutive LR altitude reasonableness tests have failed. It should be noted that R and ℓ are initially set to zero in the landing program.

If the LR Data Read Routine calls for a velocity update (FLV = 1, 2, or 3), the velocity difference δq_u as determined by the preliminary computation for the velocity updating in the State Vector Update Routine (Fig. 3.4.3-1) is used as the basic parameter in the LR velocity reasonableness test as was done with the altitude difference in the altitude test. In the velocity test case, δq_u is the difference between the LR indicated velocity component (\tilde{q}'_u of Fig. 3.4.3-1) and that estimated by the LGC with respect to the lunar surface (q'_u of Fig. 3.4.3-1). The LR velocity data reasonableness test is of similar form to the altitude test as shown in Fig. 3.4.5-1. The constant VELBIAS is stored in erasable memory. All three LR velocity components use the same velocity data reasonableness test. If the velocity difference passes the LR velocity reasonableness test and the velocity update is permitted by the astronaut, the state vector is automatically updated by the State Vector Update Routine.

5.3-82

Revised LUMINARY
 Added GSOP # 567 PCR # 943 Rev. 7 Date 11/69

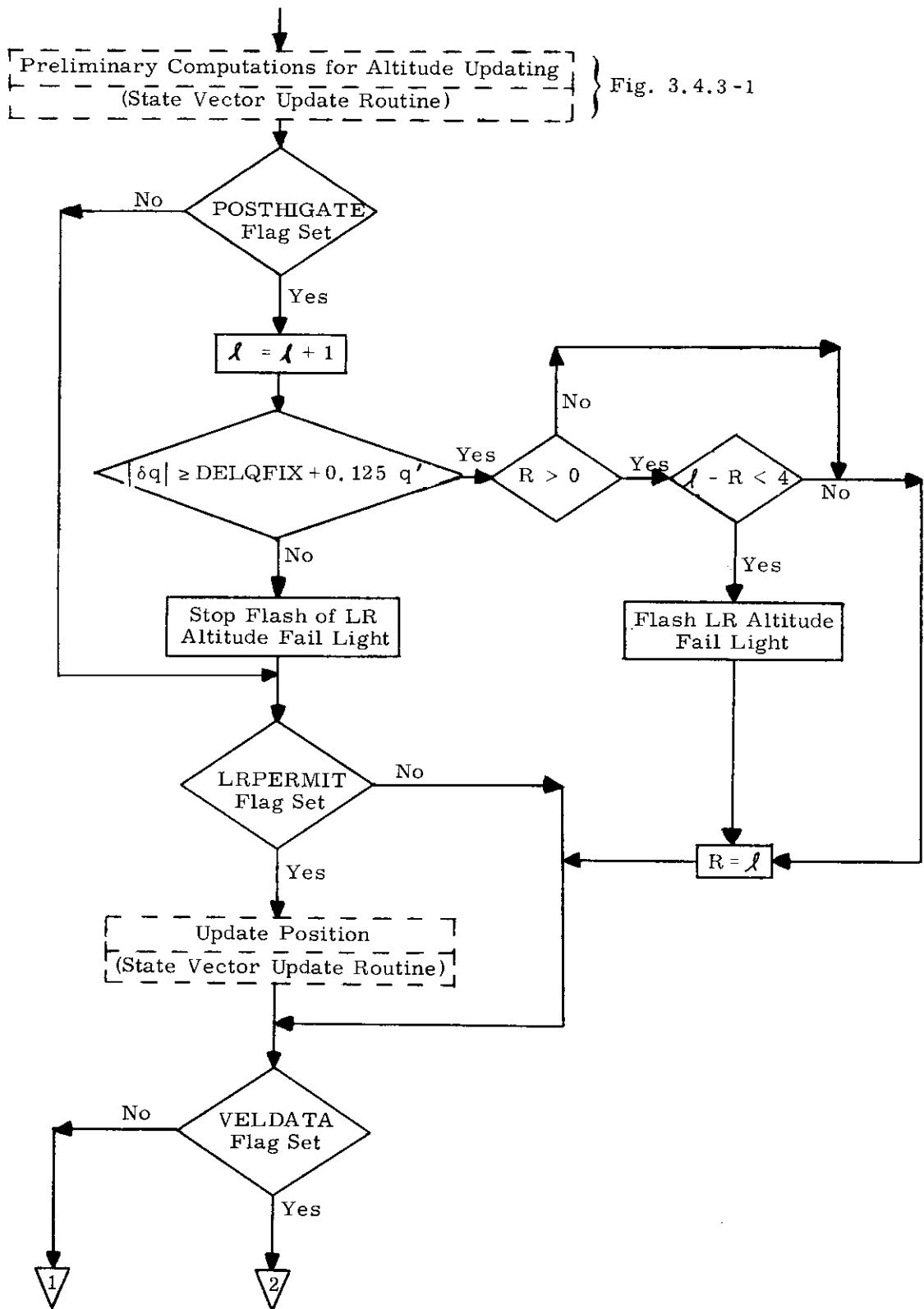


Fig. 3.4.5-1 LR Data Reasonableness Test Routine
(Page 1 of 2)

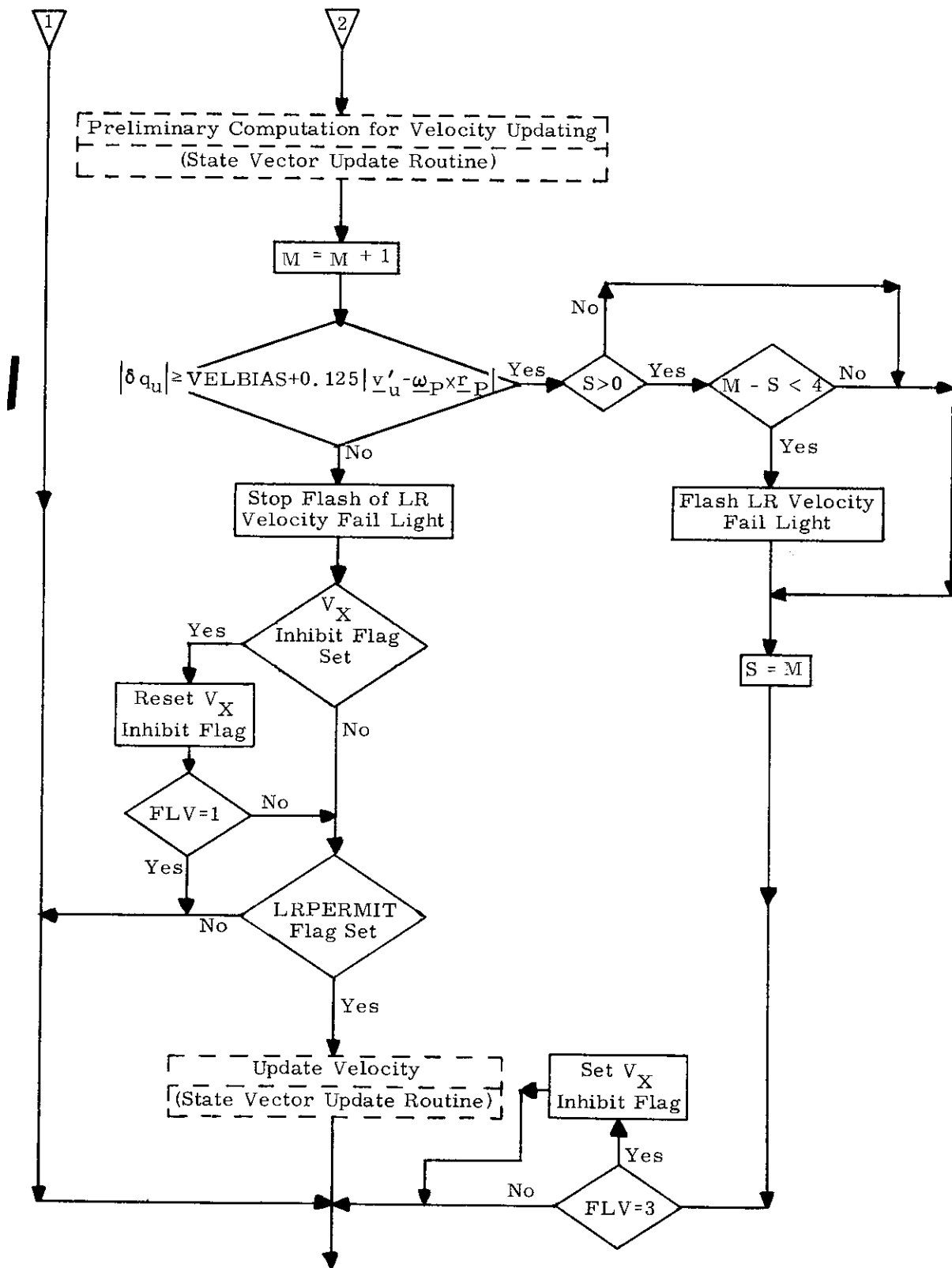


Fig. 3.4.5-1 LR Data Reasonableness Test Routine
(Page 2 of 2)

5.3-84

Revised LUMINARY
 Added GSOP # 567 PCR # 943 Rev. 7 Date 11/69

If the LR velocity data exceed the limits of the velocity reasonableness test, it is seen in Figure 3.4.5-1 that the same type of alarm criterion is used as with the altitude data test. If two or more of the last four LR velocity readings obtained by the Data Read Routine fail the velocity reasonableness test, the LR Velocity Fail Light is flashed. It should be noted that S and M are initially set to zero in the landing program.

Under certain conditions it is possible for the routine not to detect cross lobe lock-up when checking the v_{XA} component of LR velocity, but it is able to do so with the v_{ZA} component. To avoid updating the state vector with the v_{XA} component when the v_{ZA} component fails to pass the test, it is seen at the bottom of Figure 3.4.5-1 that a test is made on the velocity selection flag (FLV) to see if the v_{ZA} component had failed (i.e., FLV = 3) and, if so, the V_X Inhibit flag is set so as to prevent an update with the next v_{XA} component.

It should also be noted that since the LR data reasonableness tests follow the LR Data Good discrete checks in the LR Data Read Routine, the data reasonableness alarm criterion is based on the last four LR readings which had passed the Data Good discrete check. The LR data reasonableness tests are independent of intermittent tracking and do not account for any data rejected because of failure to pass the LR data good tests.

5.3.4.6 Guidance-and-Control Routine

The Guidance-and-Control Routine uses the data provided by the State-Vector-Update Routine to produce a thrust acceleration command for the Throttle-Command Routine and desired thrust direction and window direction vectors for the FINDCDUW routine.

The Guidance-and-Control Routine is described by an overall flow chart, Figure 3.4.6-1, and by a detailed flow chart, Figure 3.4.6-2. Notes on interpreting the charts follow the detailed flow chart. The nomenclature used in the Guidance-and-Control section is defined at the end of this section.

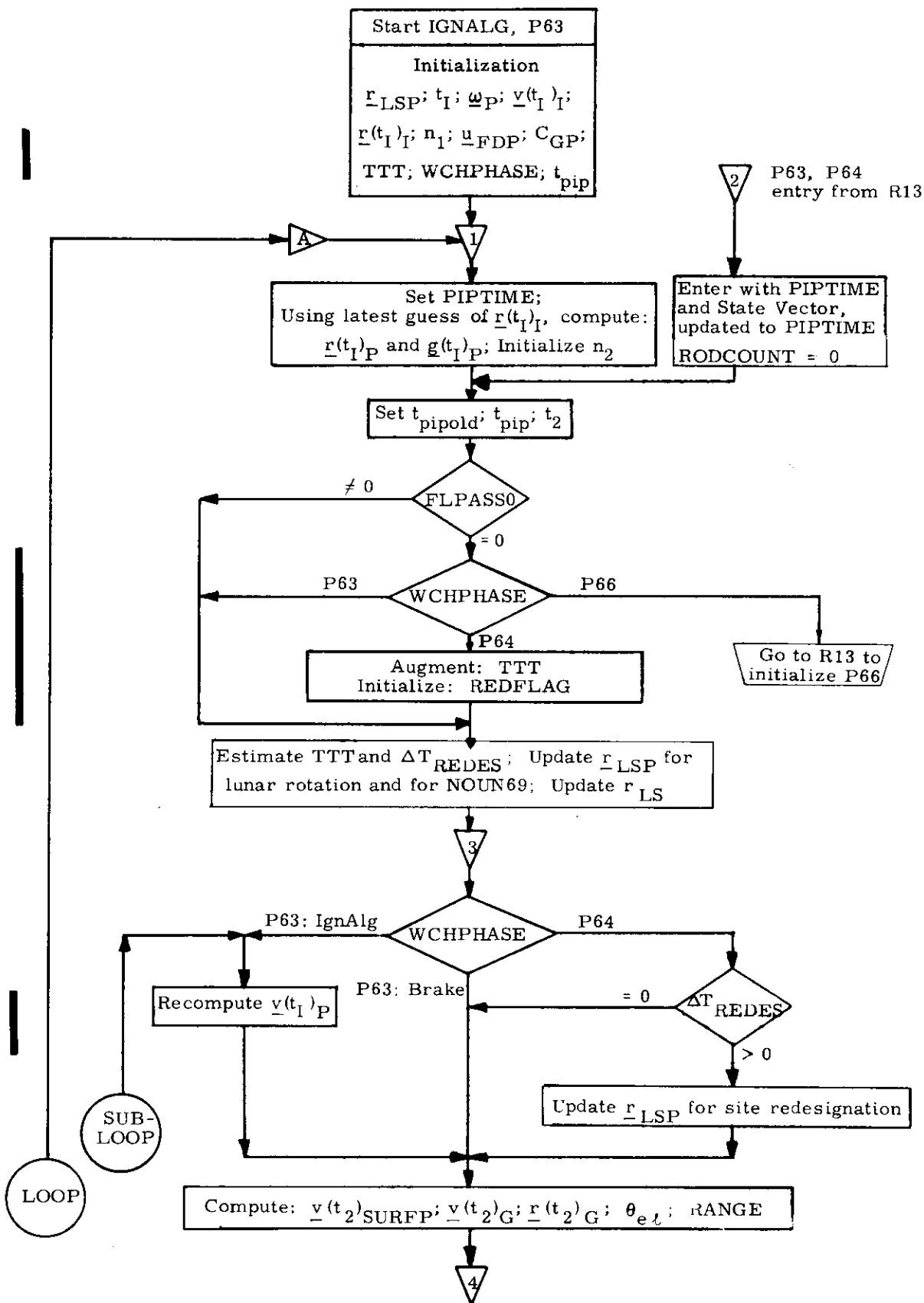


Fig. 3.4.6-1 Guidance-and-Control Routine - Overall Flow

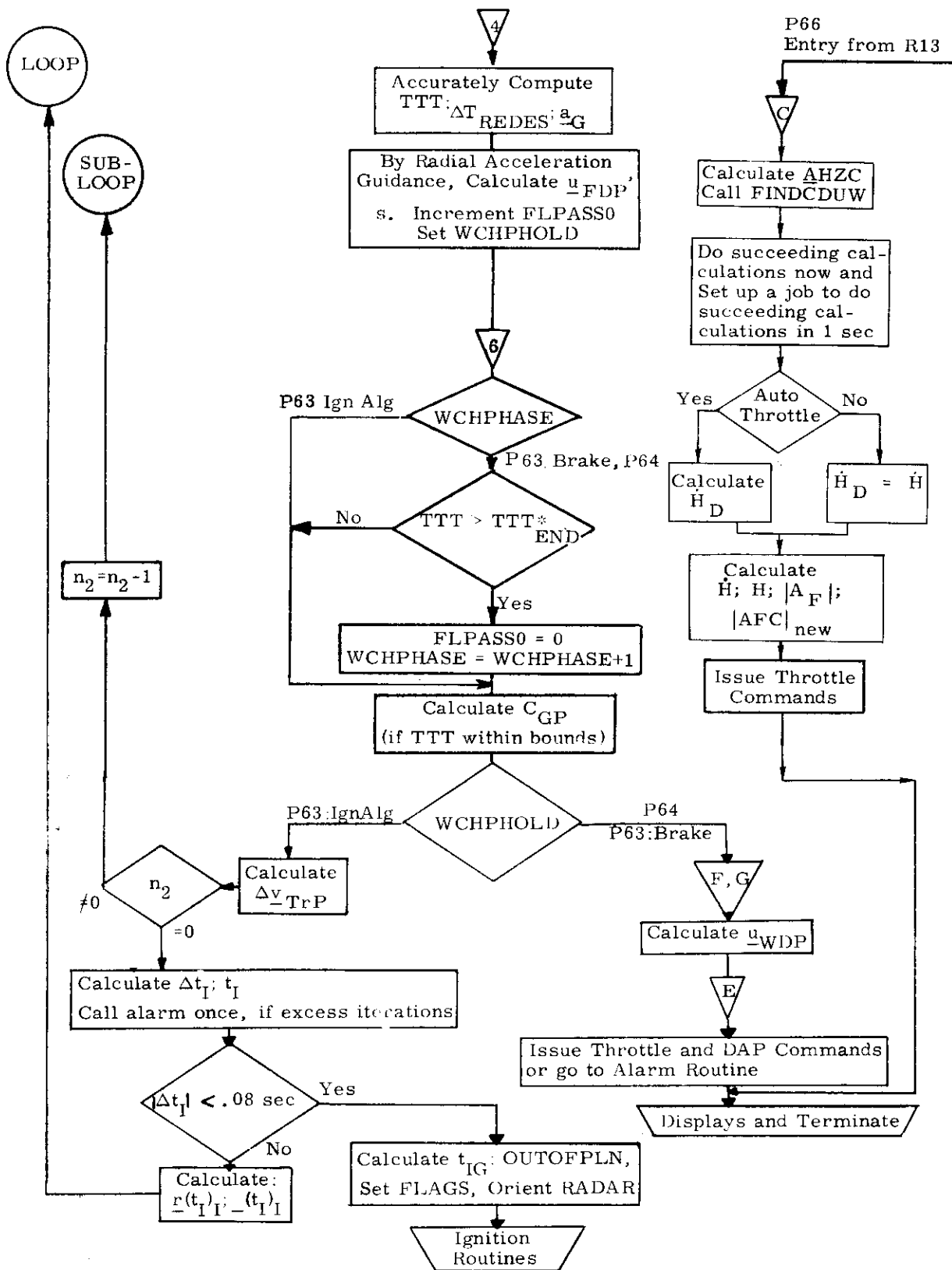


Fig. 3.4.6-1 Guidance-and-Control Routine - Overall Flow
Page 2 of 2

5.3-87

Revised LUMINARY 1C

Added GSOP # R 567

PCR # 988

Rev. 3

Date 3/70

For Conventions used in this diagram, see: Notes on Interpreting Lunar Landing Guidance Information Flow, this Section. For Nomenclature, see Lunar Landing Guidance Nomenclature, at end of diagram.

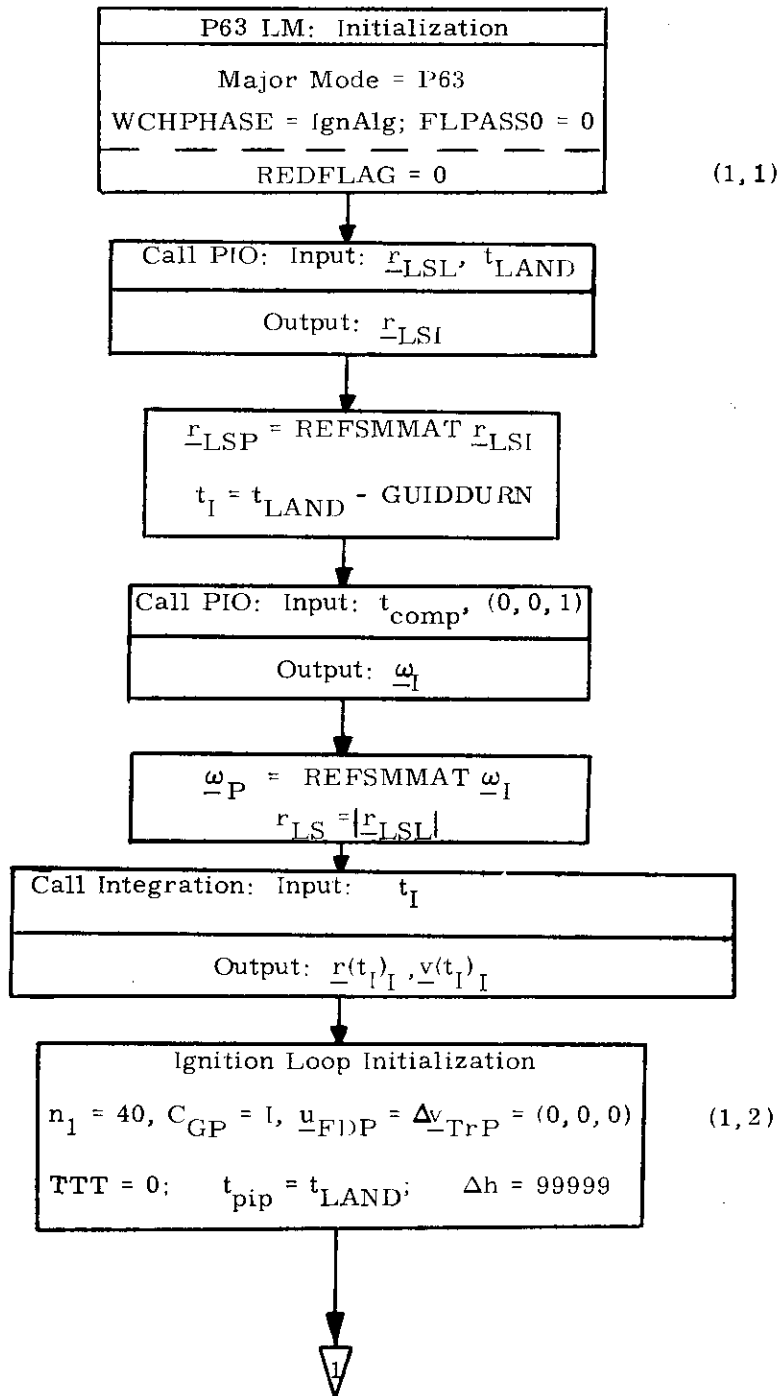


Fig. 3.4.6-2 Guidance-and-Control Routine
Page 1

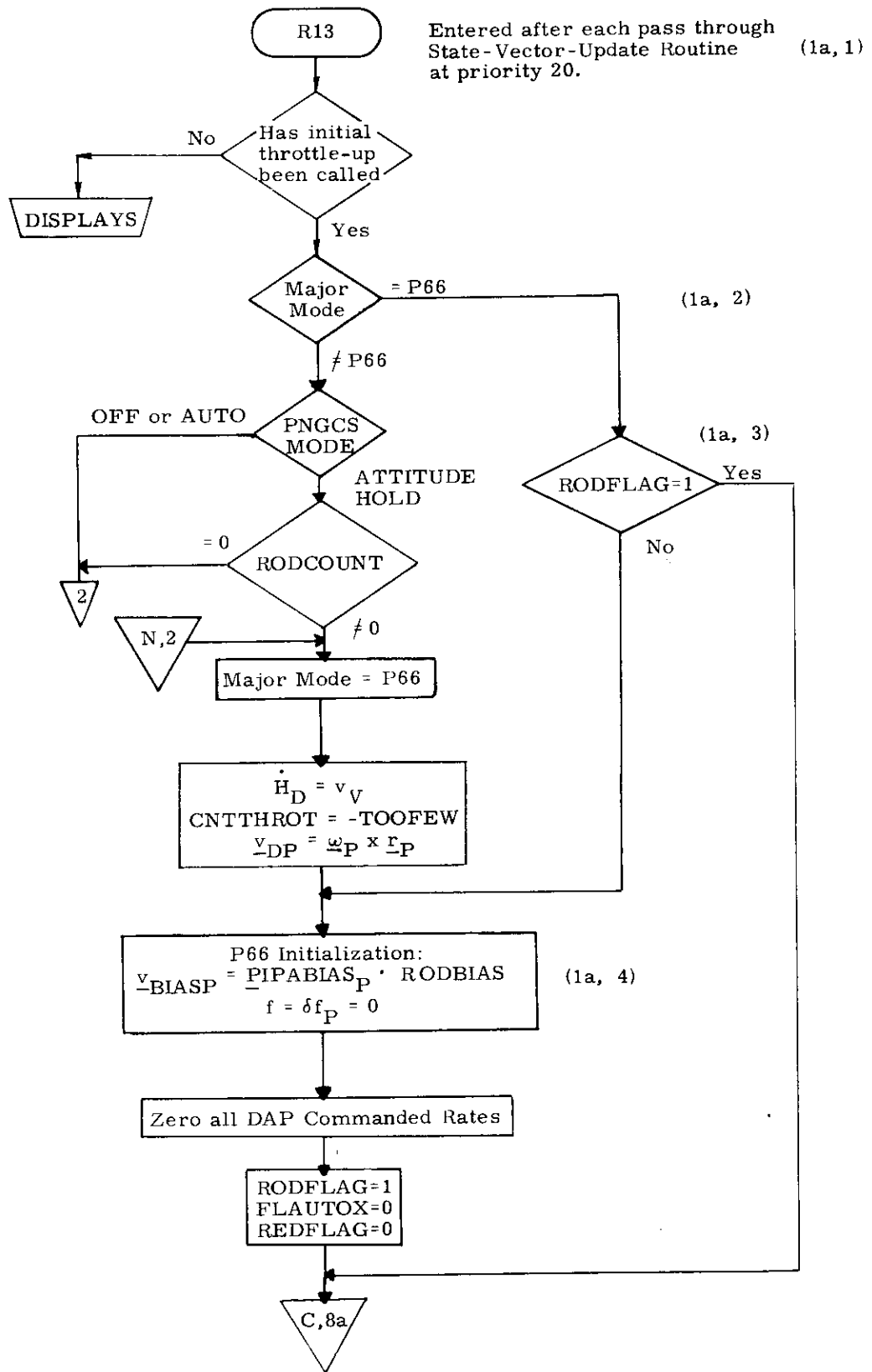


Figure 3.4.6-2. Guidance and Control Routine
Page 1a

5.3-88a

Revised

LUMINARY 1C

Added

GSOP # R-567

PCR # 988, 1013

Rev. 8

Date 3/70



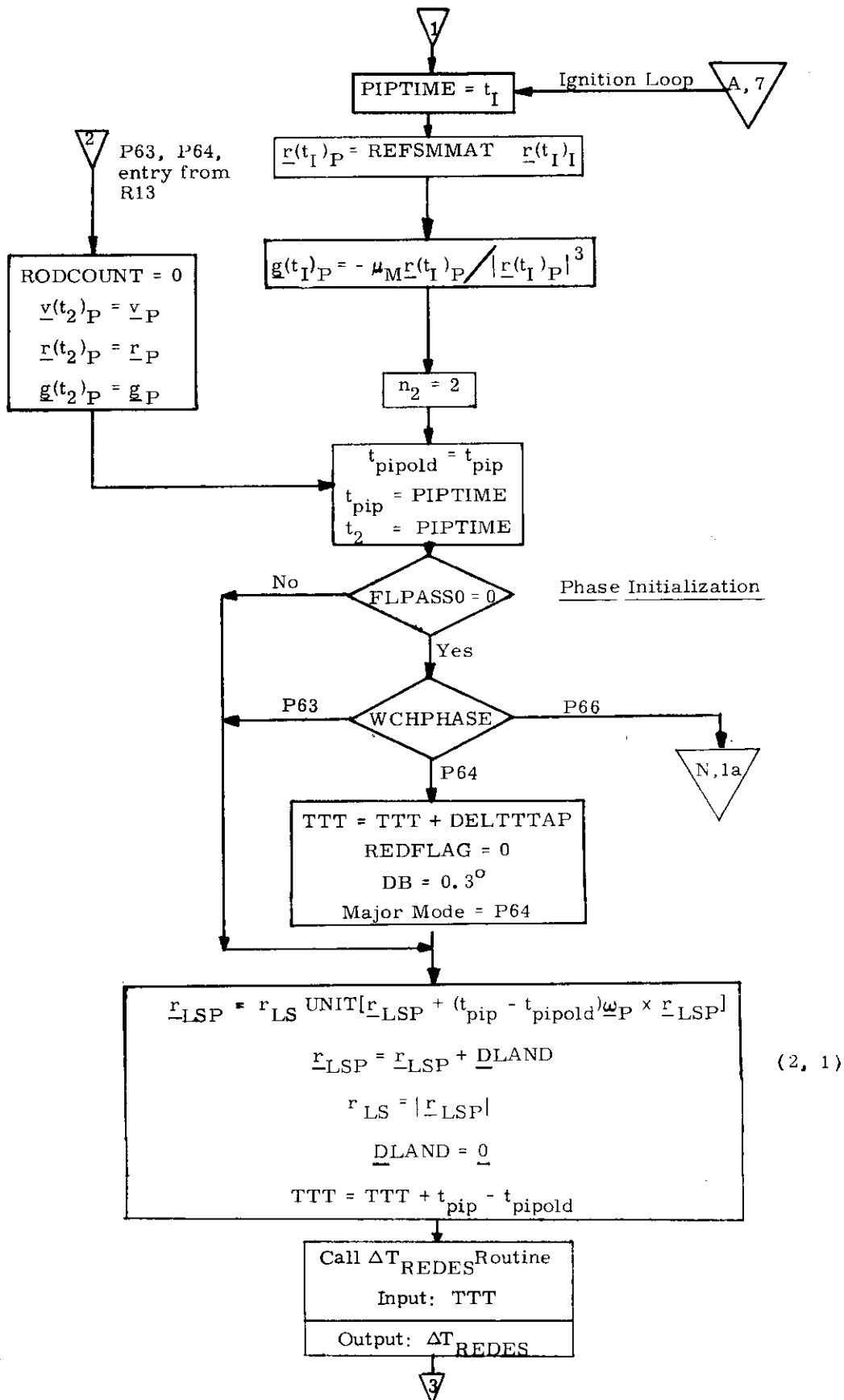


Fig. 3.4.6-2 Guidance-and-Control Routine
Page 2

5.3-89

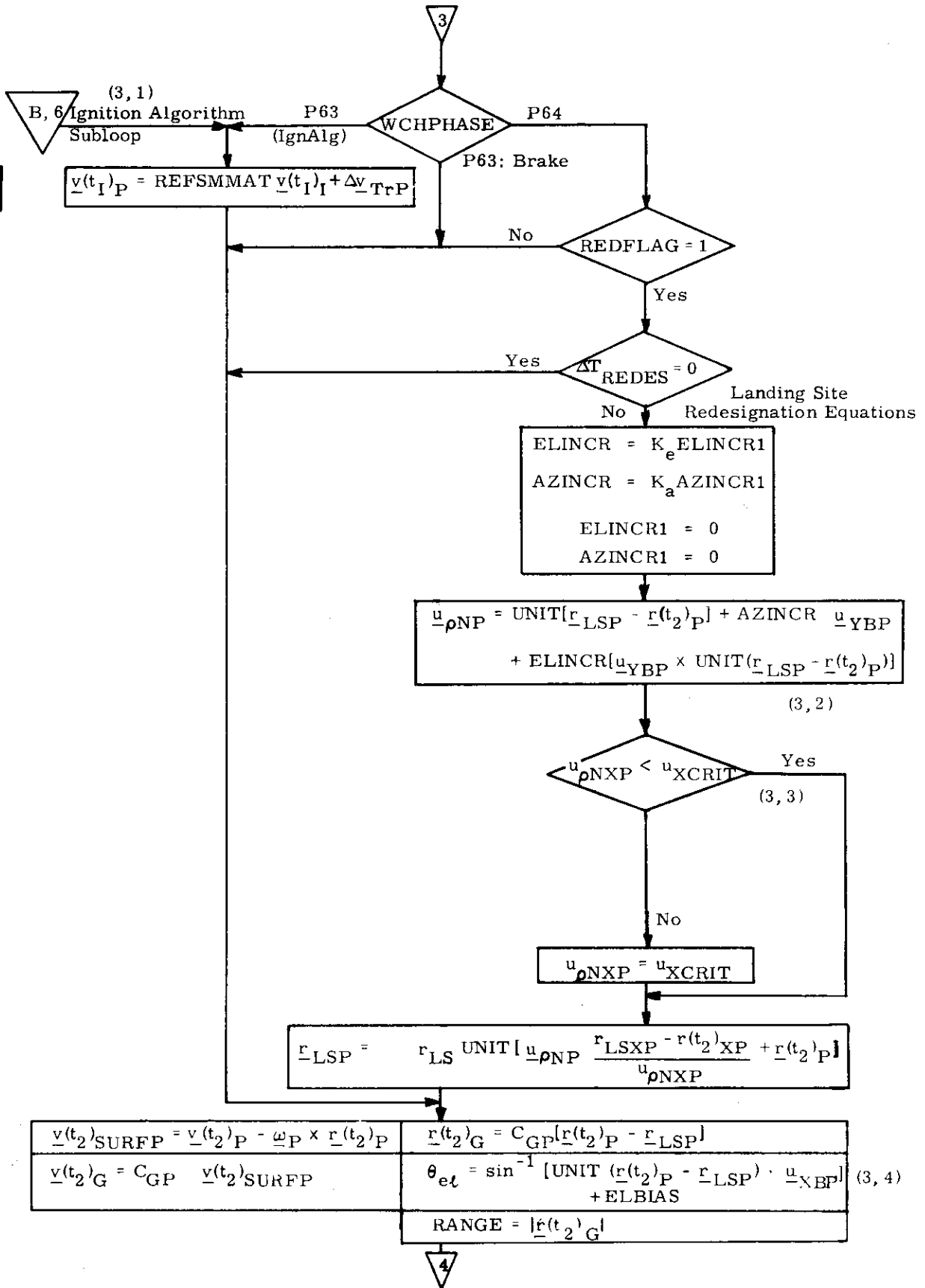


Fig. 3.4.6-2 Guidance-and-Control Routine

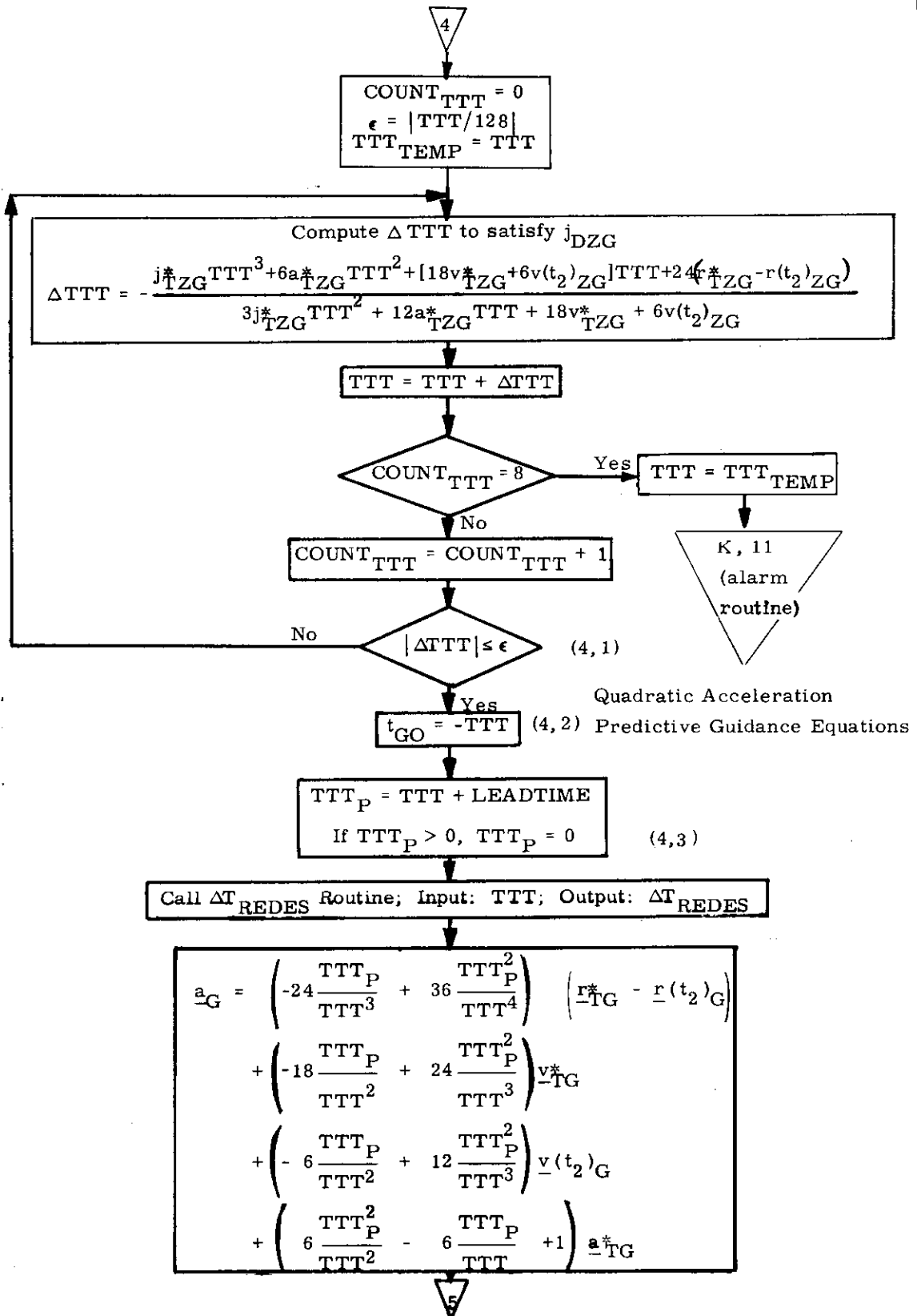


Fig. 3.4.6-2 Guidance-and-Control Routine

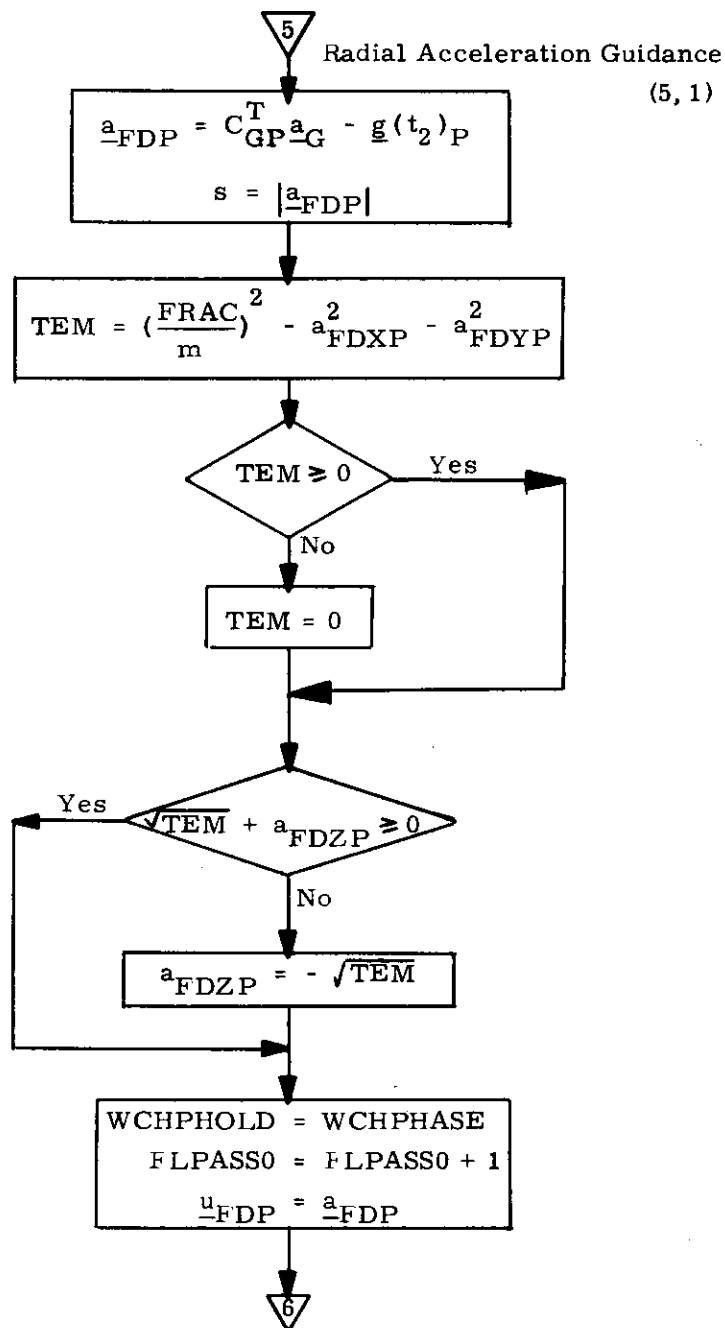


Fig. 3.4.6-2 Guidance-and-Control Routine

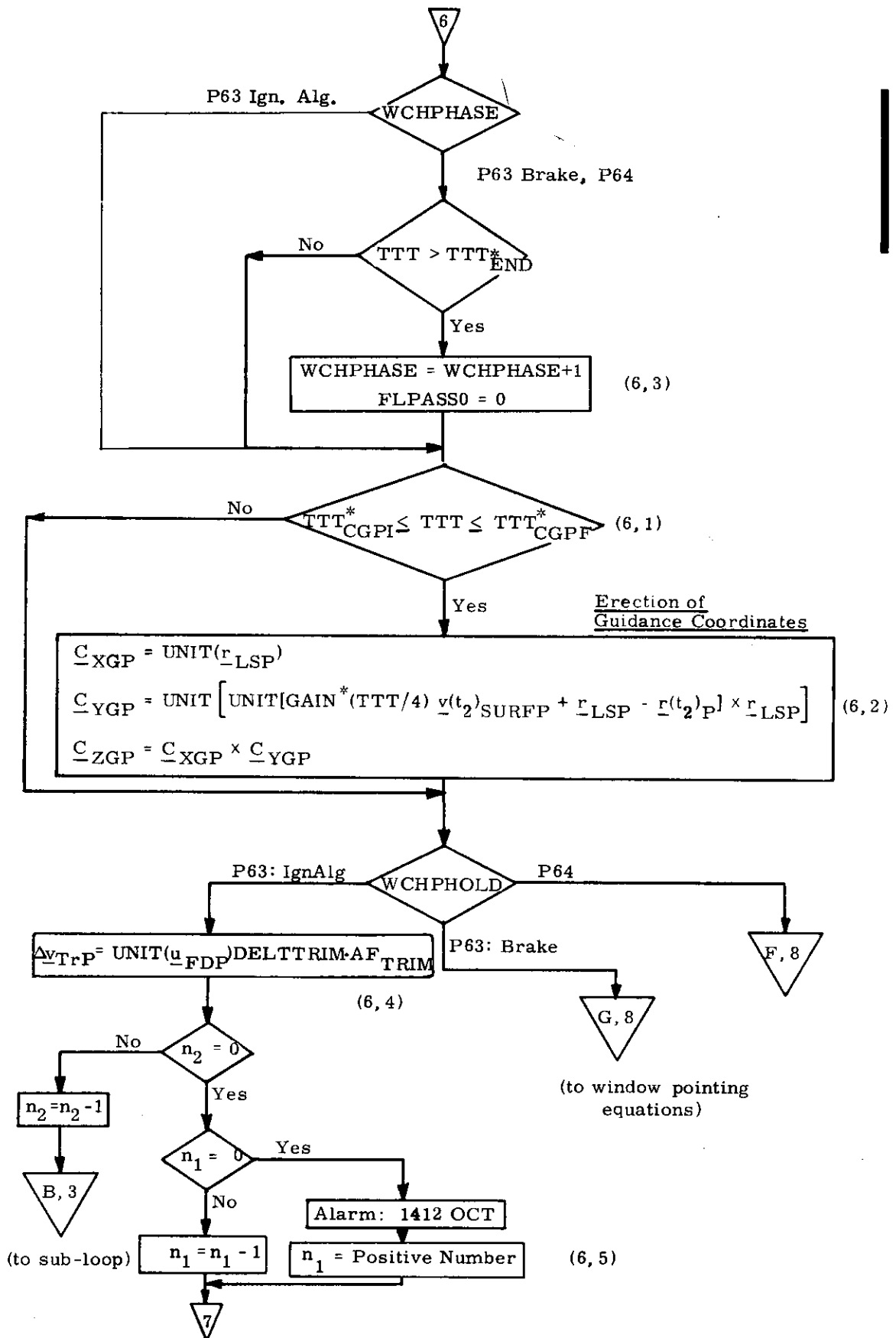


Fig. 3.4.6-2 Guidance-and-Control Routine

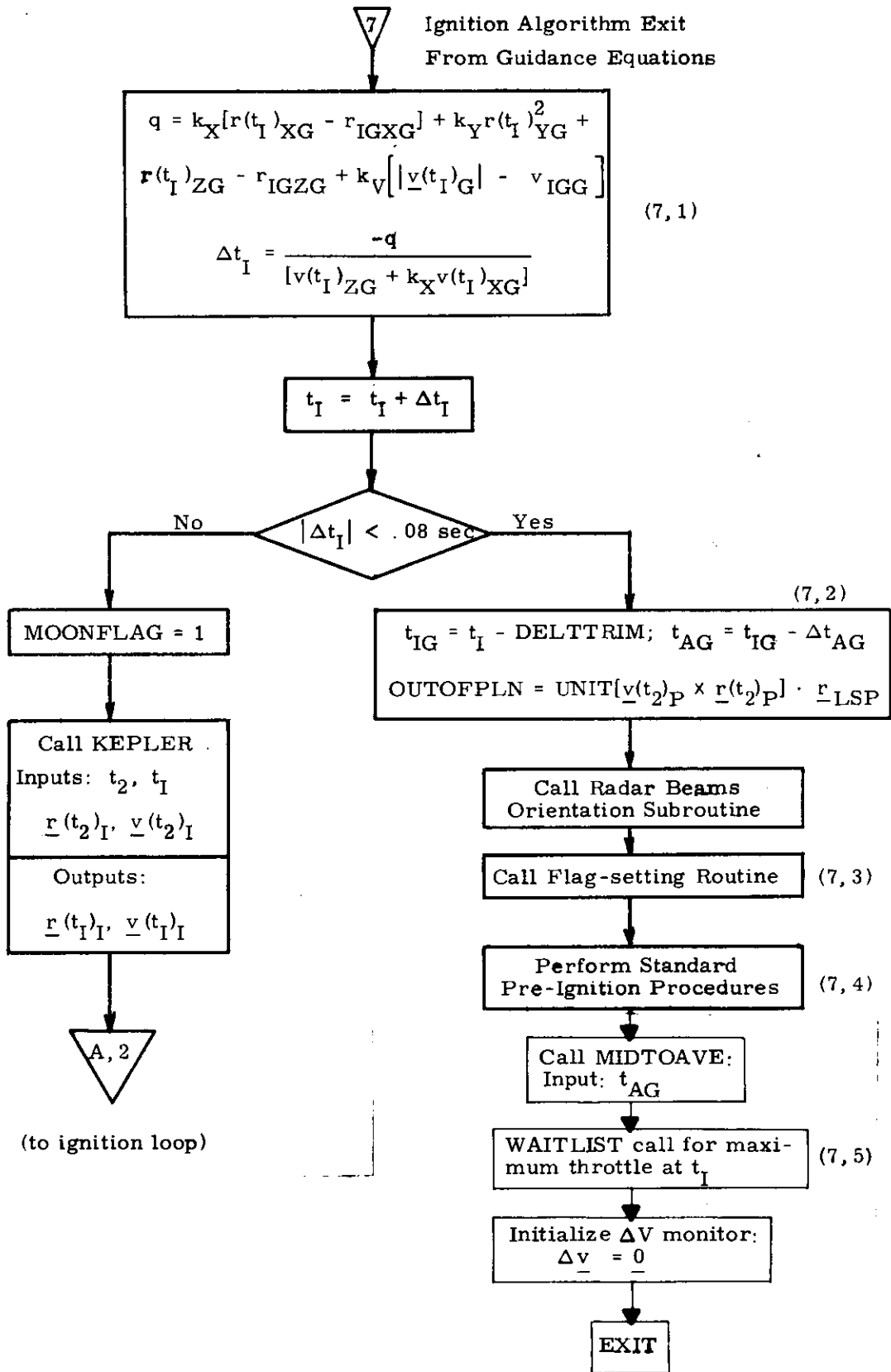


Fig. 3.4.6-2 Guidance-and-Control Routine

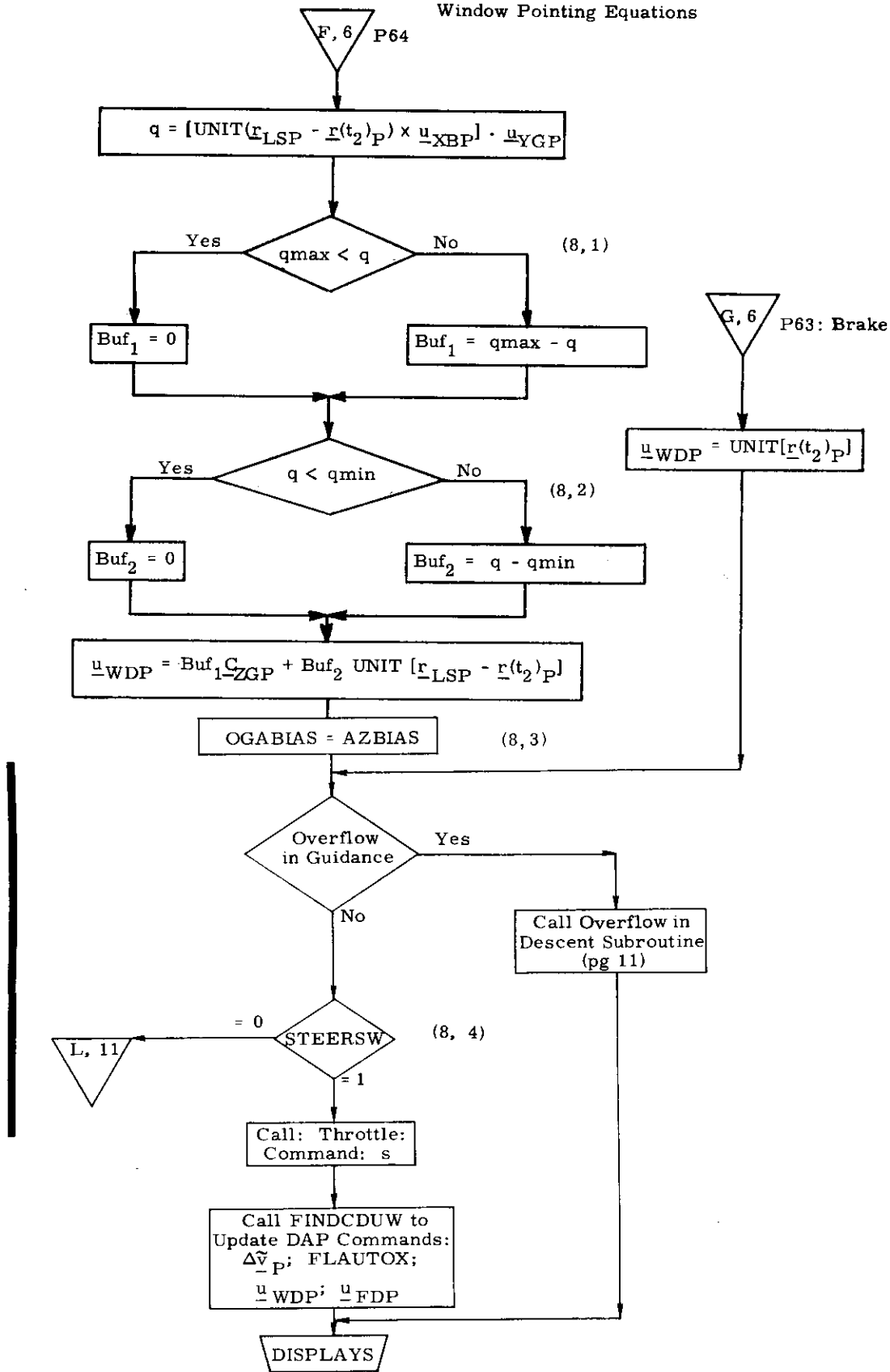


Fig. 3.4.6-2 Guidance-and-Control Routine
Page 8

5.3-95

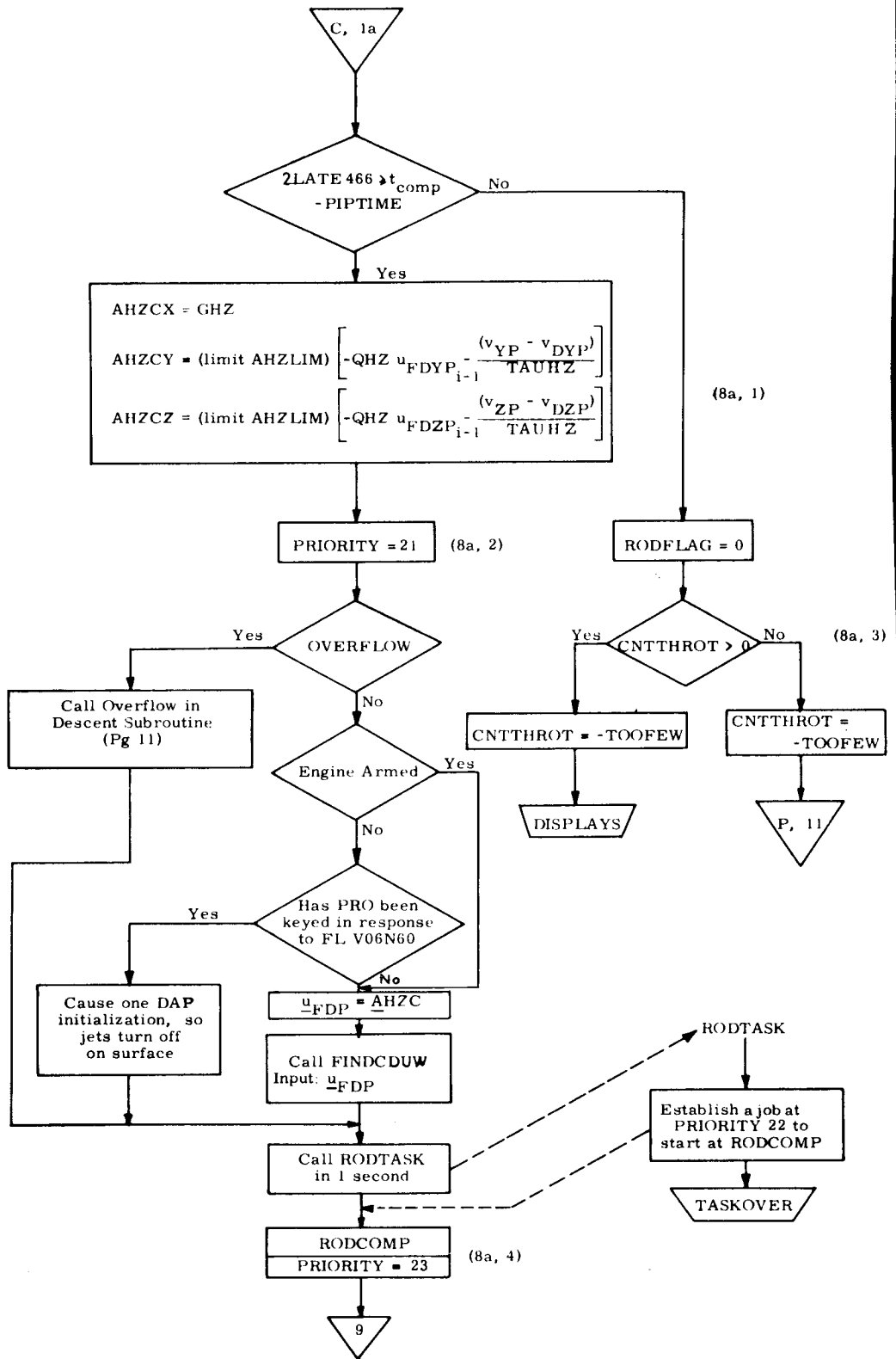


Fig. 3.4.6-2 Guidance-and-Control Routine (Page 8a)

9.3-95a

Revised

LUMINARY 1C

Added

GSOP # R-567

PCR # 988, 1013 Rev. 8

Date 3/70

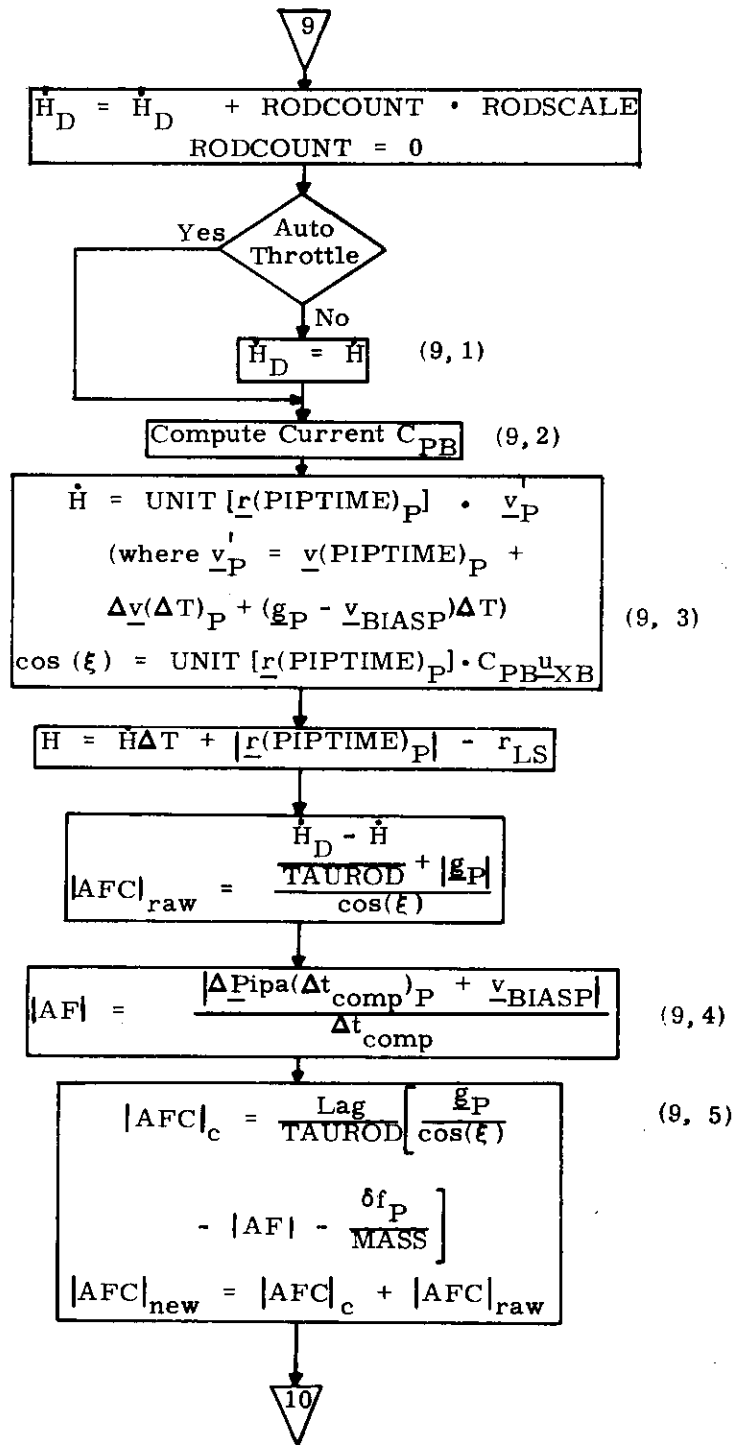
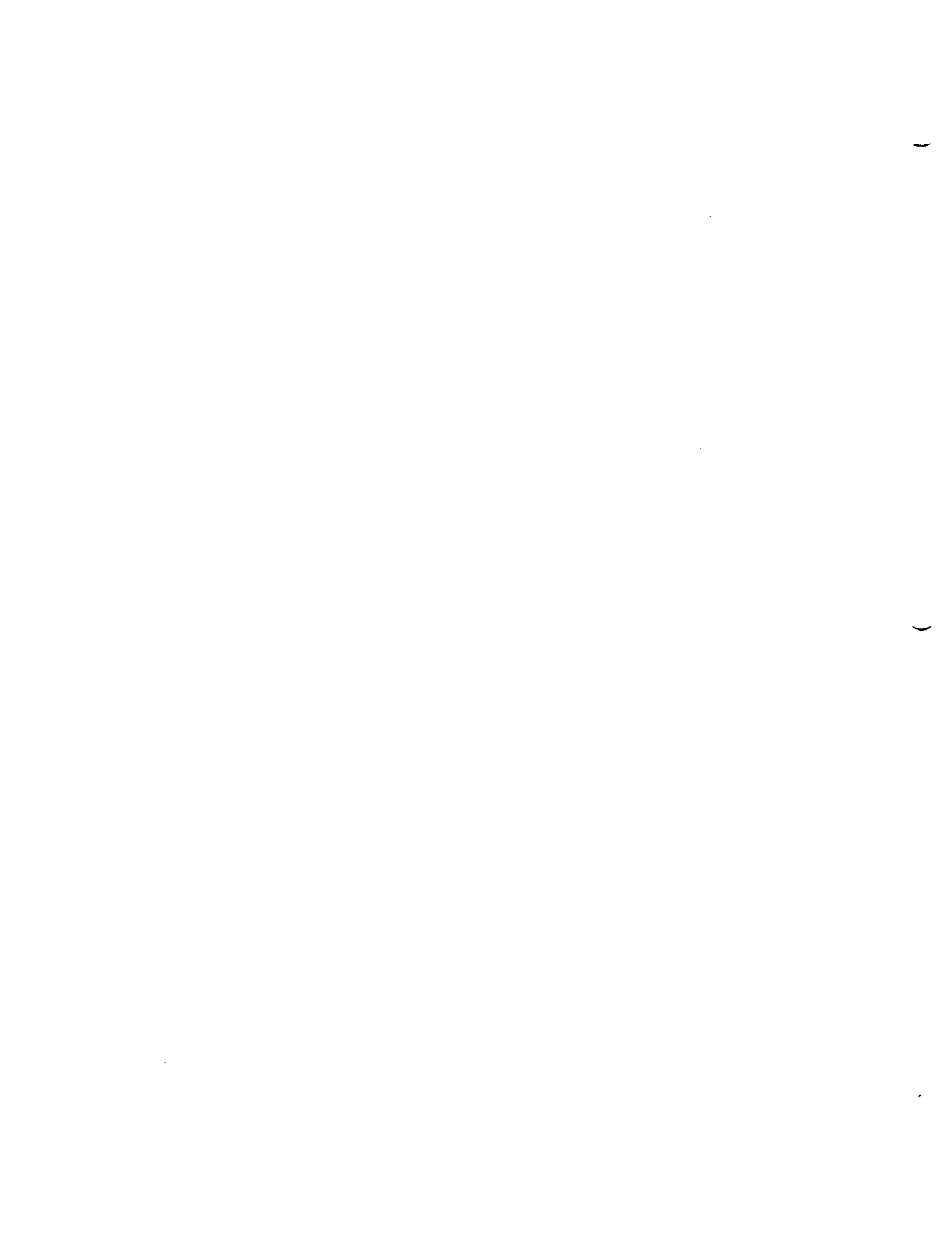


Fig. 3.4.6-2 Guidance and Control Routine



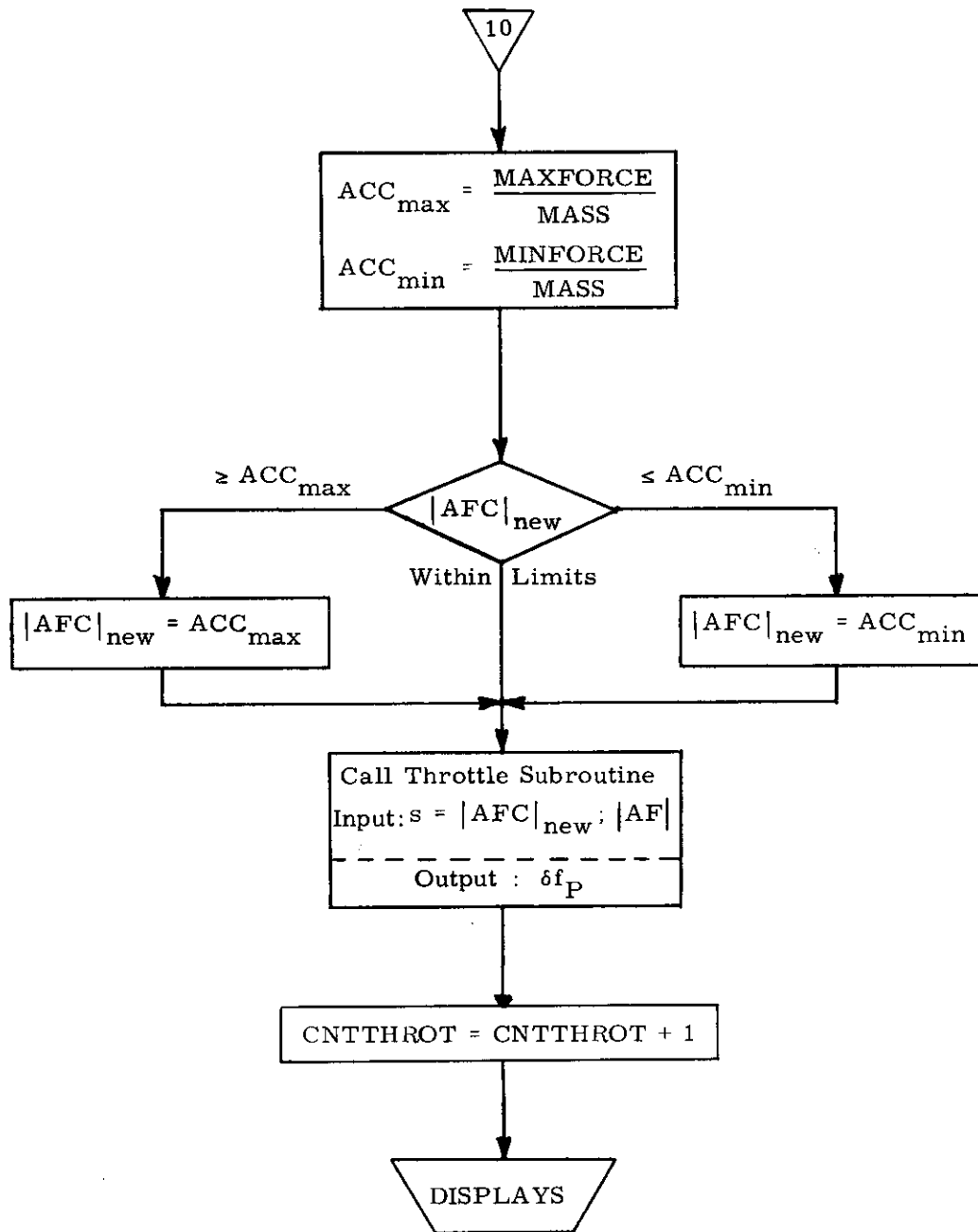
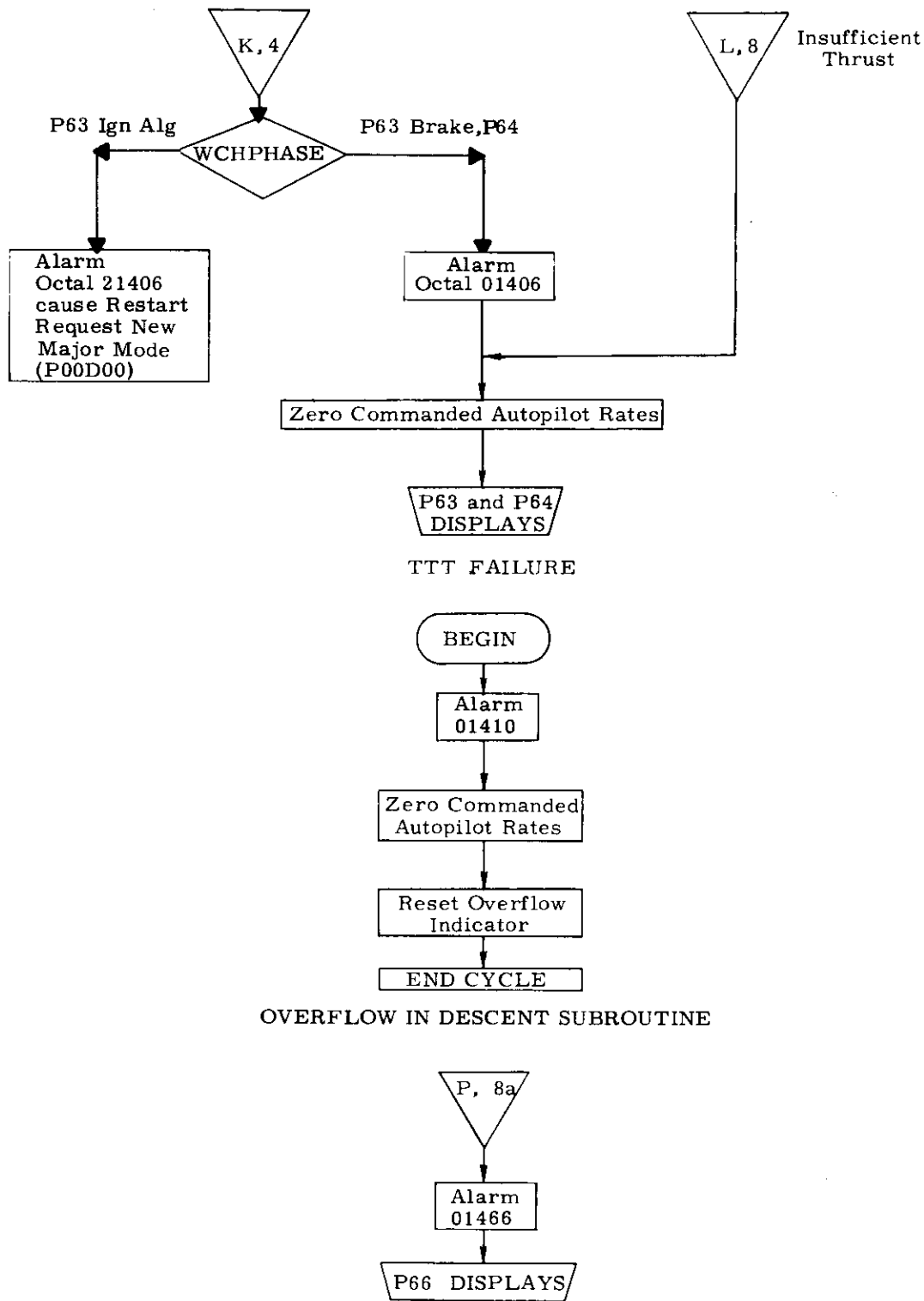


Fig. 3.4.6-2 Guidance-and-Control Routine



OVERFLOW IN DESCENT SUBROUTINE

INSUFFICIENT THROTTLING IN P66

Lunar Landing Guidance Alarms

Figure 3.4.6-2 Guidance-and-Control Routine

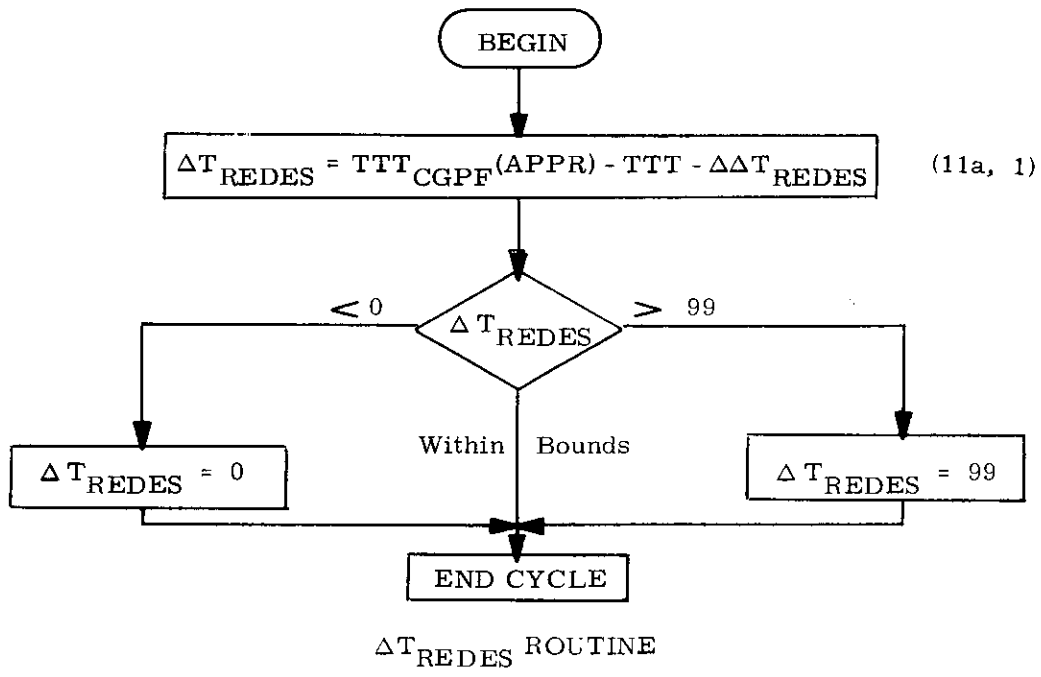
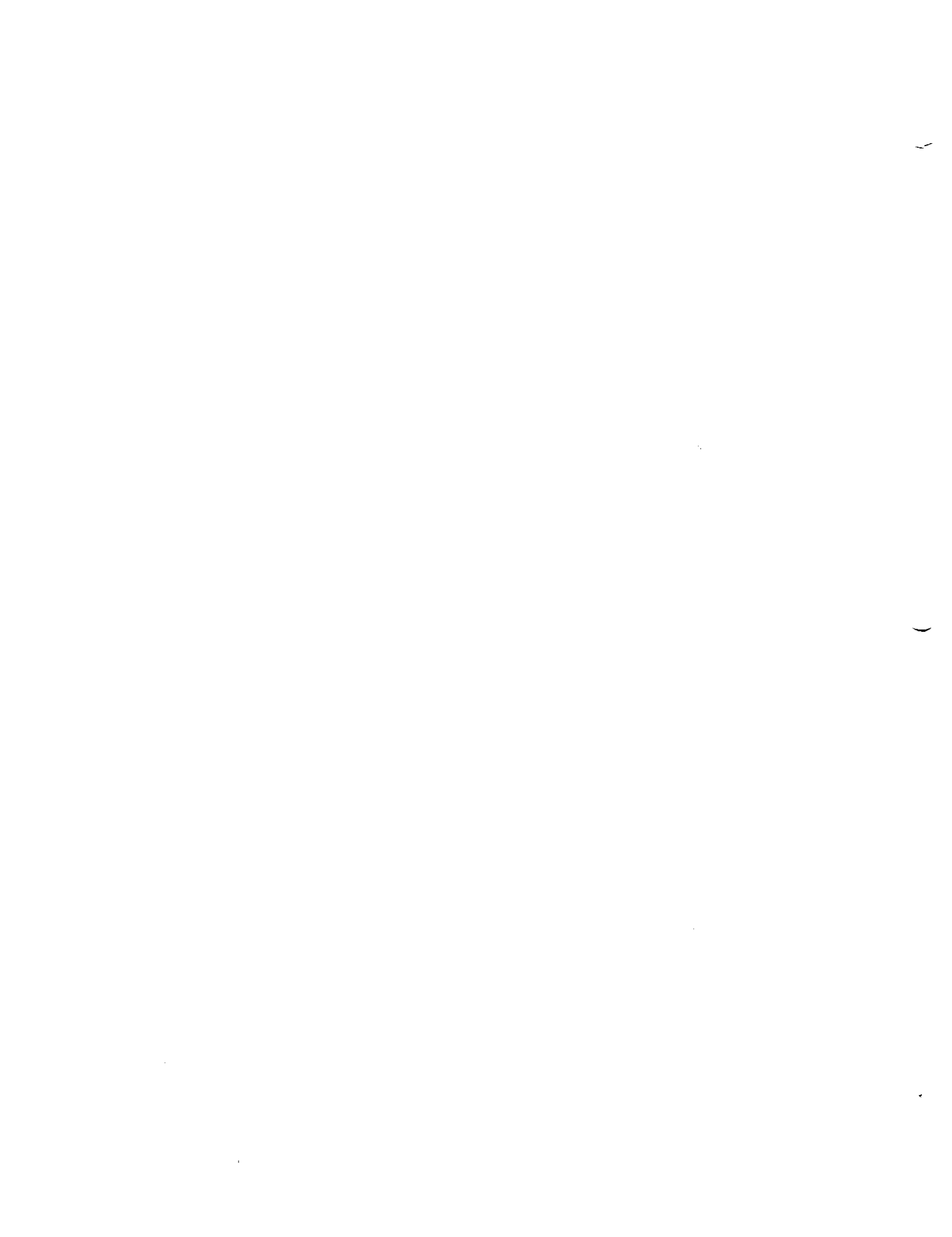
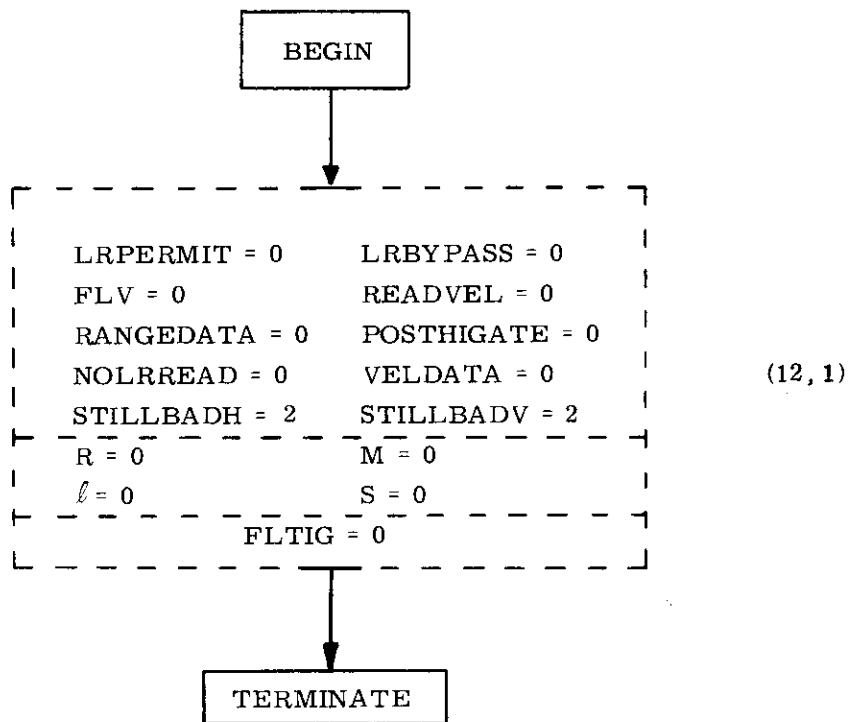


Figure 3.4.6-2 Guidance and Control Routine (page 11a)





Flag-Setting Routine
of
Lunar Landing Guidance

Fig. 3.4.6-2 Guidance-and-Control Routine
Page 12

NOTES ON INTERPRETATION OF LUNAR LANDING GUIDANCE INFORMATION FLOW

General Conventions

In this section, there are short paragraphs relevant to specific parts of the information flow diagram. These are identified by the notation (m,n), where m refers to the page of the information flow on which the relevant section may be found, and n refers to the particular note on that page. A corresponding identifier will be found to the right of the flow.

Small triangles at the bottom of the page containing integers only indicate that flow continues to the succeeding page at the triangle with the identical integer. Large triangles containing a letter followed by an integer indicate that control is passed to or from the large triangle with the identical letter on the page indicated by the integer.

Specific Notes

(1, 1) a. REDFLAG, here initialized to zero (and again zeroed at the start of P64) to forbid redesignations, will be set to one only upon receipt of a "PROCEED" response to flashing display V06N64, while ΔT_{REDES} is non-zero. AZINCR1 and ELINCR1 are initialized to zero in response to this "PROCEED".

b. FLAUTOX is not shown initialized to zero because the selection of any new Major Mode causes it to be set to zero. (The State-Vector-Update Routine sets it to one when the proper altitude is passed.)

(1, 2) The ignition algorithm loop counter, n_1 , prevents performing the pre-ignition computations endlessly without notice. It is difficult to imagine circumstances in which alarm code 1412, which n_1 controls, will be called.

(1a, 1) During each guidance cycle after the radar updates, if any, have been made, the State-Vector-Update Routine calls R13 which calls the Guidance-and-Control Routine. This continues until touchdown or abort.

(1a, 2) Because of this branch, once the ROD mode (P66) has been selected, the astronaut cannot return to the LGC automatic landing modes P63 or P64.

(1a, 3) This test determines whether a P66 deletion or a restart has occurred since the last pass through R13.

(1a, 4) See (9, 3)

(2, 1) The components of \underline{DLAND} are the components of NOUN 69 as follows: $DLAND_x = R_3$, $DLAND_y = R_2$, $DLAND_z = R_1$. The NOUN 69 registers must be initialized to zero (most probably by pad loading) before P63 is called, unless some change in the landing site is desired and loaded.

(3, 1) For a particular t_p , it has been empirically determined that three passes through the ignition algorithm subloop are necessary and sufficient to define accurately the guidance coordinate frame, and the thrust-pointing vector, \underline{u}_{FDP} .

(3, 2) Moving the Rotational Hand Controller forward (or left) produces a redesignated site forward (or left) of the previous landing site.

Thus, a + roll deflection creates a positive AZINCR,
a - roll deflection creates a negative AZINCR,
a + pitch deflection creates a negative ELINCR,
and a - pitch deflection creates a positive ELINCR.

- (3, 3) The X-component of the LOS direction is checked at this point to make sure that the site has not been redesignated past the horizon.
- (3, 4) θ_{el} is computed by a polynomial which produces accurate results only in the region $\pm 70^\circ$. Because the window outline limits visibility short of this range, the error when $|\theta_{el}| > 70^\circ$ is immaterial.
- (4, 1) This test on ΔTTT is usually satisfied on the first iteration cycle, except for the first guidance cycle of a given phase.
- (4, 2) Loading of t_{GO} necessary only to interface with other sections of this document.
- (4, 3) TTT_P is used in the calculation of \underline{a}_G to account for the lag in the guidance system between the time when the PIPAs are read (for a particular cycle) and the time when the corresponding commands to the throttle and DAP are actually implemented, thereby avoiding oscillations due to that lag.
- (5, 1) Radial acceleration guidance greatly reduces trajectory altitude variations due to engine dispersions.
- (6, 1) During the Ignition Algorithm, this test on TTT ensures that the guidance frame will not be erected if TTT has a value such that the frame might be reversed. This test should not fail after the first pass through the ignition algorithm. See also Note (11a, 1).
- (6, 2) The internal unitization in the computation of C_{YGP} is done to keep significance in the fixed-point arithmetic of the LGC, even though it is analytically superfluous.
- (6, 3) $FLPASS0$ is set to zero to ensure that phase initialization, shown on page 2 of the figure, will take place on the succeeding pass.
- (6, 4) $\Delta \underline{v}_{TrP}$ is computed because, while the trim phase does not alter $\underline{r}(t_I)_P$ appreciably, it does change $\underline{v}(t_I)_P$ significantly.
- (6, 5) If the ignition algorithm alarm is issued, a peculiarity of the LGC coding causes n_1 to be reset to the integer whose octal value is the GENADR* of the LGC location in which the alarm code is stored. This number depends upon the LGC assembly revision number, so it cannot be predicted prior to program release. However, it is safe to assume that this number will generally be sufficiently large to prevent the alarm from being issued more than once.
- (7, 1) The computation of Δt_I is made because it is very unlikely that actual conditions at t_I will be sufficiently close to nominal to yield nominal throttle control duration. Since the trajectory before ignition cannot be changed, the ignition algorithm

compensates for pre-ignition trajectory dispersions by adjusting the time at which maximum throttle will be commanded, t_I . For example, if the altitude at t_I is too great, Δt_I will be negative, allowing more time for braking. All relationships are linear, except when the craft is out of plane (non-zero Y-component of position in guidance coordinates), in which case the relationship is quadratic.

(7, 2) The computation of t_{AG} is shown at this point to provide a logical interface with the GSOP description of the MIDTOAVE Routine. In the LGC, t_{AG} is computed by the Master Ignition Routine.

(7, 3) The Flag-setting Routine is included here solely to enable interface between the GSOP Lunar Landing Guidance Information Flow and the rest of the GSOP. It is not meant to indicate that these flags are set at this point in the LGC.

(7, 4) Among the so-called Standard Pre-Ignition Procedures is the Attitude Maneuver Routine (R60) using the vector \underline{u}_{FDP} from the ignition algorithm. It should be noted that the computed navigation state vector at the end of the ignition algorithm ($\underline{r}(t_2)_P$, $\underline{v}(t_2)_P$) is not the correct initial state vector for the State-Vector-Update Routine, since the guidance cycle is started before t_{IG} .

(7, 5) This WAITLIST call is issued by the LGC Master Ignition Routine. It causes the Throttle-Command Routine to be entered at the location FLATOUT at time t_I .

(8, 1) If q is greater than q_{max} , then the landing site is visible to the astronaut and the window is pointed along the line of sight.

(8, 2) If q is less than q_{min} , then the landing site is far below the bottom of the window edge, and the window is simply pointed in the direction of the Z-axis of the guidance frame.

(8, 3) OGABIAS is zero for all thrusting programs except the lunar landing.

(8, 4) STEERSW = 0 means insufficient thrust.

(8a, 1) The notation (limit AHZLIM) indicates that the content of the subsequent brackets is magnitude limited to AHZLIM. The notation i-1 indicates the use of the command from the previous pass, and on the first P66 pass indicates the command from the final P64 pass.

(8a, 2) The priority structure of AGC jobs is not ordinarily treated in this volume. It is discussed here, however, because of its extraordinary significance to the landing guidance. The effect of this priority structure is to allow only one P66 job to run at one time. The change to priority 21 is shown in this location in order to avoid breaking up the preceding equations. Actually, AHZCX is set to GHZ, and AHZCY and AHZCZ are magnitude limited, after the priority change.

5.3-102

Revised LUMINARY IC
 Added GSOP #. R-567 PCR # 988, 1013 Rev. 8 Date 3/70

(8a, 3) If too few calls to the Throttle-Command Routine are made between P66 deletions, a non-abortive alarm is issued.

(8a, 4) Since the ROD cycle time is one second, a WAITLIST task schedules ROD at this point every other second.

(9, 1) When examining the ROD computations, it is necessary to remember that there is a finite interval of time between the time that measurements are taken (PIPA readings and radar readings) and the time when the locations holding the state vector are actually loaded with the new state vector. If the ROD computations occur during that interval of time, then PIPTIME (the validity time of the position and velocity vectors used in the computations) is not the most recent time when the measurements were taken, but is the previous time. Only in the next pass through the ROD computations will PIPTIME equal PIPTIME1. (c.f. : definitions)

(9, 2) There is no LGC equivalent of the MATRIX C_{PB} . In the LGC a vector is transformed from platform to body coordinates and vice versa by the LGC sub-routines *SMNB* and *NBSM* by rotating the vector through each successive gimbal angle, never explicitly calculating a transformation matrix.

(9, 3) \underline{v}_{BIASP} is the delta v bias accruing per second, and thus has the units of acceleration.

(9, 4) The Δt_{comp} in this equation is time since PIPTIME if this is the first time ROD is entered. Otherwise, Δt_{comp} is as defined in the nomenclature.

(9, 5) $|AFC|_c$ corrects $|AFC|_{raw}$ for an extrapolation of \dot{H} through the time interval Lag. At the extrapolated time, the change in thrust is expected to be achieved. This extrapolation prevents overshoots in \dot{H} in response to step changes in commanded \dot{H} .

(11a, 1) The constant $\Delta\Delta T_{REDES}$ assures one, and probably two, guidance frame erections after the final site redesignation. See also Note (6, 1).

(12, 1) The Flag-setting Routine initializes flags used in State-Vector-Update, LR routines, and the MIDTOAVE Routine. Flags used in LR updates are: LRPERMIT, LRBYPASS, FLV, RANGEDATA, NOLRREAD, STILLBADH, STILLBADV, READVEL, POSTHIGATE and VELDATA. FLTIG is used in the MIDTOAVE Routine, and R, l, M, and S are the initial settings of the LR Data Reasonableness Test Counters.

5.3.4.6.1 Nomenclature of the Guidance-and Control Routine

Conventions

1. A superscript asterisk denotes a phase-dependent quantity; i. e., one erasable is indicated for the braking phase, a different erasable for the approach phase. If one quantity is always used regardless of phase, as in the computation of ΔT_{REDES} , the phase is indicated by a parenthetical note.
2. a. Time quantities measured relative to liftoff are denoted by a name starting with lower-case t.
b. Time quantities measured relative to the time the spacecraft will (or if allowed would) achieve the projected target conditions are denoted by a name starting with TTT. An exception to this is the quantity t_{GO} , which is retained only because it is an input to other sections of this document.
c. Quantities representing an interval of time are denoted by a name starting with Δt .
3. Names of vector quantities are, in general, derived from the following building blocks:

<u>Δ</u>	<u>Vector</u> denotes change
<u>u</u>	Unit vector
<u>r</u>	Position vector
<u>v</u>	Velocity vector
<u>a</u>	Acceleration vector
<u>j</u>	Jerk vector ($d^3 r/dt^3$)
<u>()</u>	<u>Time of</u> vector validity in parentheses

Random Modifiers:

LS	Landing Site
T	Target
D	Desired

Coordinate Frames, always the last letter in the subscript:

G	Guidance
P	Platform
L	Lunar-Fixed (Selenographic)
I	Inertial (Basic Reference)

4. Components of a particular vector are denoted by the name of the vector, with no letters underlined, and with X, Y, or Z inserted in the penultimate position of the subscript.
5. Transformation matrices, with the exception of REFSMMAT, are denoted by C plus two subscripts, the first of which indicates the frame of the vector after transformation, and the second of which denotes the frame of the vector after pre-multiplication by the matrix.

For example, C_{GP} will transform a vector \underline{v} to guidance from platform coordinates in this manner:

$$\underline{v}_G = C_{GP} \underline{v}_P$$

6. Row vectors of a matrix are denoted as follows:

$$C_{GP} = \begin{bmatrix} \underline{C}_{XGP} \\ \underline{C}_{YGP} \\ \underline{C}_{ZGP} \end{bmatrix}$$

7. Letters F and P under the heading TYPE indicate the type of storage used for constants which are Fixed and Padloaded-Erasable, respectively. V indicates variable. (Polarities of padloaded constants can differ from those in the document).

TIME QUANTITIES

<u>NAME</u>	<u>TYPE</u>	<u>DEFINITION</u>
DELTTTRIM	P	Duration of trim maneuver.
DELTTTAP	P	An augment added to TTT during initialization for the first guidance cycle of the approach phase. When the value of TTT computed on the final guidance cycle of the braking phase is thus augmented, it nominally becomes the correct value for the first guidance cycle of the approach phase. This causes the subsequent iterative computation of TTT to converge on the first or second pass. Without this augment, the computation of TTT would require many iterations on the first guidance cycle of the approach phase and would needlessly waste computer time.
ΔT_{REDES}	V	Time remaining to redesignate the landing site.
$\Delta\Delta T_{REDES}$	F	Time interval to assure at least one computation of guidance coordinates after the last site redesignation.
GUIDDURN	F	Estimated time interval from full throttle time to t_{LAND} .
LEADTIME	P	A positive time increment used to compensate for the computation delay, plus the average of the delay in achieving the throttle change and the delay in achieving the attitude change.
PIPTIME	V	Time of validity of the state vector provided either by the pre-ignition computations or by the State-Vector-Update Routine.
TAUVERT	P	Inverse gain of velocity-error-nulling guidance used in program P65. This is the time constant of the exponential nulling of the velocity error.
t_{comp}	V	Time at which a computation takes place. That is, t_{comp} is the current time, as taken from the computer clock.
$t_{compold}$	V	Previous t_{comp} .

<u>NAME</u>	<u>TYPE</u>	<u>DEFINITION</u>
Δt_{comp}	V	$t_{\text{comp}} - t_{\text{compold}}$
t_I	V	The time computed by the pre-ignition computations for commanding maximum throttle (start of braking phase).
Δt_I	V	Correction to the maximum throttle time at the start of the braking phase to correct for deviations from nominal of the state at that point. More precisely, Δt_I corrects for the residual in the equation relating the various components of the state vector deviation from the nominal at the t_I computed on the current iteration of the pre-ignition computations.
t_{IG}	V	Time of DPS ignition (the start of the TRIM phase, minimum throttle).
t_{LAND}	P	Time of landing (initial estimate).
t_{pip}	V	A storage location for PIPTIME used by descent guidance.
t_2	V	A short synonym for t_{pip} .
t_{pipold}	V	Contains t_{pip} of the previous pass through descent guidance.
TTT	V	Time relative to achieving target conditions, a negative number = $-t_{\text{GO}}$.
$\text{TTT}_{\text{CGPF}}^*$	P	The value of TTT at which computation of the matrix C_{GP} must be terminated.
$\text{TTT}_{\text{CGPI}}^*$	P	The value of TTT at which the computation of the matrix C_{GP} can be initiated.
$\text{TTT}_{\text{END}}^*$	P	Criterion for phase switching.
TTT_{P}	V	A predicted value of TTT to account for the total delay between reading the accelerometers and achieving the thrust acceleration commanded on the basis of those accelerometer readings.
TTT_{TEMP}	V	Temporary quantity for TTT.

Names of VECTOR QUANTITIES not derived as in part 3 of Nomenclature

<u>NAME</u>	<u>TYPE</u>	<u>DEFINITION</u>
Δv_{TrP}	V	Nominal velocity increment accrued due to the thrust acceleration during the trim phase.
\underline{D}_{LAND}	V	Change in the landing site vector requested via N69. Must be initialized to zero by pad loading.
$\underline{g}(t_2)_P$	V	Acceleration due to gravity at time t_2 .
$\underline{\omega}_P$	V	Lunar rotational angular velocity, platform coordinates.
\underline{u}_{FDP}	V	Thrust acceleration command vector, platform coordinates. This vector is given a name usually associated with unit vectors because it is an output from the Guidance-and-Control Routine to the FINDCDUW Routine and to the Attitude Maneuver Routine which is used as a <u>pointing</u> Command only. There is no requirement for this vector to be of unit (or semi-unit) length, and it is not unitized.
$\underline{u}_{\rho NP}$	V	Unit vector from the current LM position to the redesignated landing site, platform coordinates.
\underline{u}_{WDP}	V	Window-pointing vector, not necessarily unit. See definition of \underline{u}_{FDP} .
\underline{v}_{VERTG}	P	Desired velocity for P65, in guidance coordinates.
$\underline{v}(t_2)_{SURFP}$	V	Velocity with respect to the surface, in platform coordinates.

MISCELLANEOUS QUANTITIES

<u>NAME</u>	<u>TYPE</u>	<u>DEFINITION</u>
AF _{TRIM}	F	Thrust acceleration expected during Trim Phase.
AZBIAS	P	Azimuth bias to correct LPD alignment error. See also ELBIAS and OGABIAS.
AZINCR	V	Commanded azimuth angle change accrued since last pass through guidance, radians.
AZINCR1	V	Net count of AZIMUTH redesignation commands issued since last pass through Guidance-and-Control Routine.
DB	V	DAP Deadband width.
ELBIAS	P	Elevation bias to correct LPD alignment error. See also AZBIAS and OGABIAS.
ELINCR	V	Commanded elevation angle change accrued since last pass through guidance, radians.
ELINCR1	V	Net count of elevation redesignation commands issued since last pass through Guidance-and-Control Routine.
FLPASS0	V	A flag which, when zero, indicates the initial guidance cycle of any given phase of the lunar landing.
FRAC	F	Maximum available thrust from DPS (used for radial control guidance).
GAIN*	P	Gain factor, to reduce excessive rotation of the guidance coordinate frame, especially during site redesignation, thereby also reducing excessive yawing during approach phase.
K _a , K _e	F	Scale factors relating AZINCR and ELINCR to AZINCR1 and ELINCR1.
K _V , K _X , K _Y	P	Coefficients used by the pre-ignition computations in computing Δt_1 .
m	(P)	Mass of the vehicle: register is pad-loaded, although the astronaut has the option to reload it after R03 is begun. The equations for updating mass are given in the State-Vector-Update Routine.
n ₁	V	Ignition algorithm loop counter.

<u>NAME</u>	<u>TYPE</u>	<u>DEFINITION</u>
n ₂	V	Ignition algorithm subloop counter.
OGABIAS	P	Outer gimbal angle bias input to FINDCDUW. See also AZBIAS and ELBIAS.
OUTOFPLN	V	Amount landing site is out of CSM orbital plane.
qmax, qmin	F	Upper and lower criteria used in choosing window-pointing command vector.
r _{IGZG} , r _{IGXG}	P	Components of the nominal position vector in guidance coordinates at t _{LAND} - GUIDDURN
s	V	Specific force, or thrust acceleration.
u _{XCRIT}	F	Maximum value for elevation increment. Ensures site will not be redesignated above the horizontal plane, which would yield a site to the rear of the current LM position.
v _{IGG}	P	Absolute value of nominal velocity in guidance coordinates at t _{LAND} - GUIDDURN.
WCHPHASE	V	The current phase of the lunar landing.

FLAGS

<u>NAME</u>	<u>MEANING WHEN = 1</u>	<u>MEANING WHEN = 0</u>
FLAUTOX	Inhibit X-axis override	Allow X-axis override
REDFLAG	Landing site redesignation permitted	Landing site redesignation not permitted
RODFLAG	No P66 deletion or Restart has occurred since RODFLAG last set	P66 deletion or restart has occurred
STEERSW	Sufficient thrust - Call Throttle-Command Routine and FINDCDUW Routine	Insufficient thrust - Do not call Throttle-Command Routine or FINDCDUW Routine

5.3-110

Revised LUMINARY 1C

Added GSOP # R-567 PCR # 988 Rev. 8 Date 3/70

QUANTITIES RELATED TO P66

<u>NAME</u>	<u>TYPE</u>	<u>DEFINITION</u>
ACC _{max} ACC _{min}	V	Current limits on thrust acceleration.
AF	V	Measured acceleration during interval since last pass through ROD computations.
AFC _c	V	Commanded acceleration, compensated for the fact that there was δf_p , a delay in achieving thrust for last throttle setting, and computational and engine response lag.
AFC _{new}	V	Tweaked, commanded acceleration, to be sent to throttling subroutine.
AFC _{raw}	V	Commanded acceleration, derived from \dot{H}_D before tweaking.
<u>AH</u> ZC	V	Commanded thrust acceleration vector for nulling horizontal velocity error.
AHZCX AHZCY AHZCZ	V	Components of <u>AH</u> ZC
AHZLIM	P	Maximum allowable value of AHZCY and AHZCZ
CNTTHROT	V	Counts number of times the Throttle-Command Routine has been called since last initialization of CNTTHROT. Initialized at -TOOFEW.
δf_p	V	F-weight: Commanded force times total system lag divided by the guidance cycle interval.
$\Delta \text{Pipa} (\Delta t_{\text{comp}})_P$	V	Total PIPA count accrued since last pass through ROD computations.
ΔT	V	$t_{\text{comp}} - \text{PIPTIME}$: Time interval since the validity time of the current state vector.
$\Delta v (\Delta T)_P$	V	Change in velocity since PIPTIME.
GHZ	F	Approximate Lunar Gravity at P66 altitude.
\underline{g}_P	V	Acceleration due to gravity at PIPTIME.
\dot{H}	V	Altitude
H	V	Measured altitude rate. Note: $\dot{H} = v_v$, computed in the State Vector Update Routine, until it is updated by P66 guidance.
\dot{H}_D	V	Desired altitude rate

5.3-111

Revised

LUMINARY 1C

Added

GSOP #. R-567

PCR # 988, 1013 Rev. 3

Date 3/70

NAME	TYPE	
Lag	P	Time lag in ROD computation system from the time when ROD reads the PIPAs, to the earliest possible time for engine to receive new thrust command.
MAXFORCE } MINFORCE }	P	Thrust limits available from throttling routine.
PRIORITY	V	Priority level assigned to each job scheduled in the AGC.
QHZ	P	A gain constant for horizontal nulling.
RODBIAS	P	PIPA bias scaling for ROD time cycle.
RODCOUNT	V	Net incremental change in desired altitude rate accrued since last pass through ROD computations (or since last pass through guidance).
RODSCALE	P	Altitude rate increment produced by one manipulation of the Rate-of-Descent Controller.
TAUHZ	P	Inverse gain constant for horizontal nulling.
TAUROD	P	Inverse gain of ROD guidance. This is the time constant of the exponential nulling of the altitude rate error.
TOOFEW	P	One less than the minimum number of throttlings between P66 deletions.
\underline{u} FDP	V	Thrust direction input to FINDCDUW. Also stores the last valid thrust acceleration vector command from P66 or P64.
\underline{v} BIASP	V	Bias accrued by PIPAs in the ROD computation period (one second), computed <u>once</u> , when P66 is first called, from pad-loaded erasables.
\underline{v} DP	V	Commanded velocity vector for horizontal guidance. Initialized at approximately lunar surface rate.
ξ	V	Angle between X-axis of vehicle and the local vertical.
2LATE466	P	Critical time interval between current time and PIPTIME of current servicer cycle beyond which one P66 pass is skipped (two ROD passes are skipped).

5.3-112

5.3.4.7 Throttle-Command Routine

The Throttle-Command Routine determines the thrust increment Δf_{th} required to achieve the thrust acceleration command s issued by the Guidance-and-Control Routine, and issues the corresponding command (ΔBIT , a bit count increment) to the Descent Engine Control Assembly.

The Throttle-Command Routine operates the engine in the throttleable region or at maximum thrust. The throttleable region is limited by manual throttle voltage at the lower end, and by mixture ratio variations and radically increasing erosion at the higher end. The upper limit of the throttleable region is defined by the switching criterion FHI, which is defined as that thrust above which the engine must not operate unless the throttle is moved to its maximum position, limited by a hard mechanical stop.

An information-flow diagram for the Throttle-Command Routine is presented as Figure 3.4.7-1. The basic characteristics of this routine are as follows:

An initial value of command thrust f is first computed, using the current estimated vehicle mass m and the command thrust acceleration s . When it is desired to place the throttle at its maximum setting, a command thrust augment δf_{sat} is required to insure that the throttle is kept and held against the mechanical stop.

The measured thrust \tilde{f} corresponding to the present throttle setting is computed using the magnitude of the measured PIPA-output data $\Delta \tilde{v}_p$ for the interval Δt between the last two PIPA-processing times. Because of computation delays and other factors, the change in throttle setting is not actually accomplished until close to the subsequent PIPA processing time. Therefore the term $\left| \Delta \tilde{v}_p \right| m / \Delta t$ is not an accurate measurement of thrust at the current throttle setting unless the throttle setting was not changed the previous guidance cycle. Accordingly, if the previous throttle cycle was within three seconds of the current cycle, a correction term δf_p is added to improve the accuracy of the measurement. The subscript p is used to indicate that the correction term used on the current cycle through the Throttle-Command Routine was computed on the preceding cycle.

In order to determine whether to operate in the throttleable region or at maximum throttle, the Throttle-Command Routine compares the present thrust command f , and the previous thrust command f_p , against the switching criterion FHI, and a somewhat lower criterion FLO. If the throttle is in the throttleable region, these tests cause the throttle to remain in the throttleable region as long as the thrust command remains below FHI. If the throttle is at maximum, these tests cause the throttle to remain at maximum as long as the thrust command remains above FLO. The separation between FLO and FHI assures the throttle will never oscillate between the maximum setting and the throttleable region.

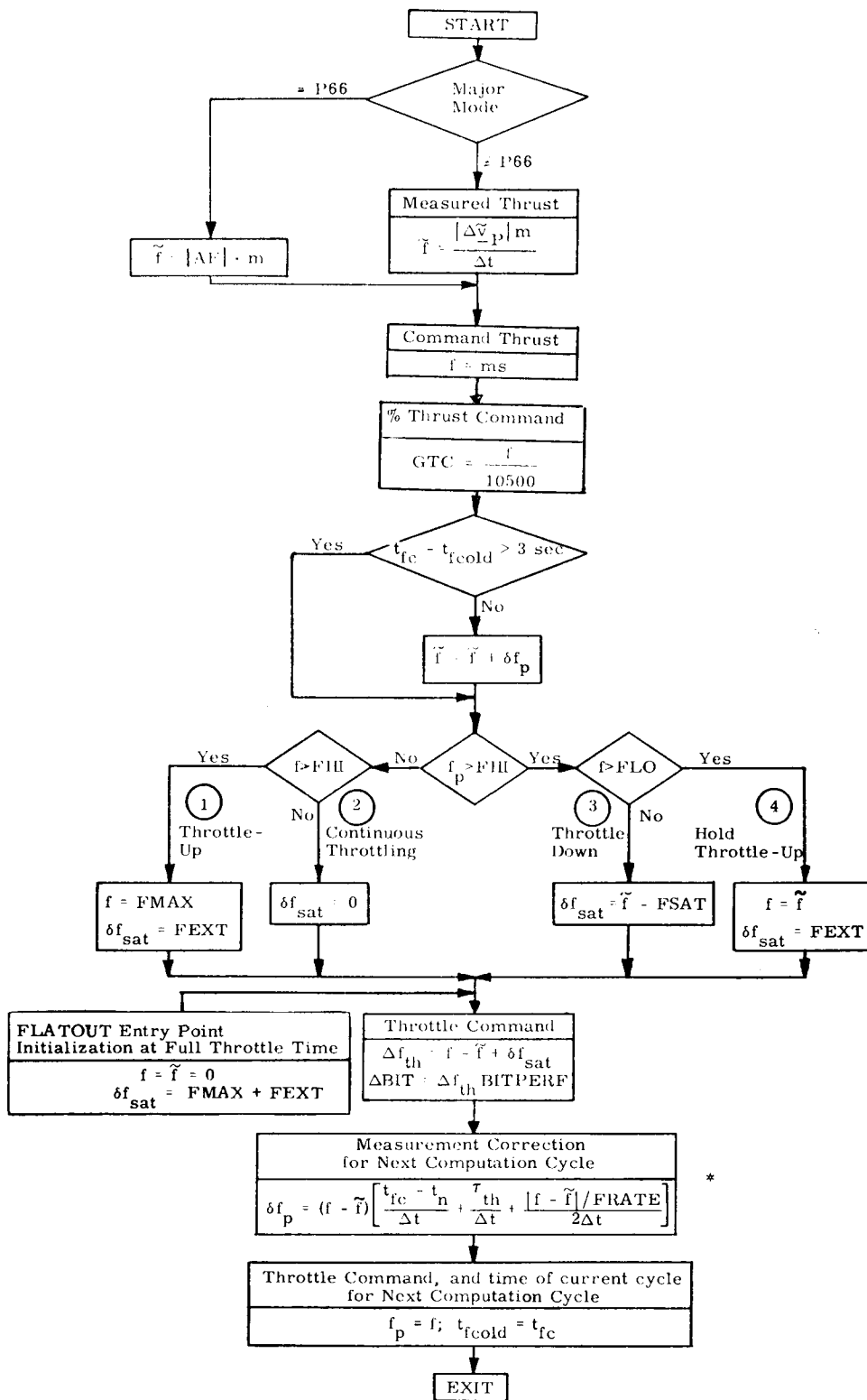


Fig. 3.4.7-1 Throttle-Command Routine

* Δt is the period of the guidance cycle of the guidance program calling the Throttle-Command Routine. Thus in P66, Δt is half its value in other guidance programs.

5. 3-114

The correction term δf_p is computed for the subsequent cycle to account for three effects. The first term within the square brackets accounts for the elapsed time between reading the PIPAs and issuing the throttle command. The second term accounts for the dynamic delay of the engine in responding to a change in throttle setting. The third term accounts for the ramp character of the throttle command proceeding from the Descent Engine Control Assembly, i. e. , following the LGC computation, the voltage from this device changes at a constant rate from the initial to the final value, it does not change in step fashion. This third term is important only when a large change in thrust is commanded, such as when throttle control is recovered during the braking phase.

5.3.4.8 Landing Maneuver Display Computations

During the lunar landing maneuver, the LGC is required to provide digital displays on the DSKY and drive analog-type vehicle displays. The digital displays are updated at the normal PIPA-output processing times, i. e., every two seconds. The analog displays are updated at approximately .25-second intervals between the normal PIPA processing times.

DSKY Display Parameters

The following digital display parameters are required for the lunar-landing programs P63 to P66 as automatic or callable DSKY displays as specified in Section 4:

- 1.) H: Altitude
The estimated local altitude of the vehicle as derived from the difference between the current position vector (r_P) and the landing-site radius magnitude (r_{LS}).
- 2.) H: Altitude Rate
The component of the vehicle's velocity that lies along the local-vertical direction (u_{hP}).
- 3.) v_P : Inertial Velocity
The magnitude of the vehicle's inertial velocity ($v(t_2)_P$)
4. ΔV_M : Measured or Accumulated ΔV
The total required velocity increment (ΔV) accumulated to the current time during the powered landing maneuver.
5. TFI: Time from DPS Ignition
The difference between the present time (t) and the DPS ignition time (t_{IG}), where t_{IG} is the time at which the trim phase is started.

- | | | |
|-----|----------------------|---|
| 6. | Δh : | Altitude Difference
The difference between the measurement of local altitude as derived from LR range data (\tilde{q}^*) and the altitude derived from the extrapolated state estimate (q'). |
| 7. | TTT: | Time relative to achieving target conditions
(TTT = $-t_{GO}$) |
| 8. | ΔT_{REDES} : | Time remaining to redesignate the landing site. |
| 9. | v_{HF} : | Forward component of the horizontal velocity of the vehicle with respect to the moon.
Computed by same equations that compute v_{HF} for analog meters. |
| 10. | OUTOFPLN: | Out-of-phase or cross-range distance between the initial LM descent orbital plane and the original landing site. This parameter, displayed to the nearest 0.1 n. mi., represents the cross-range distance that will be traversed during the powered landing maneuver. A positive sign signifies a landing trajectory out-of-plane change to the north, negative to the south. This computation is performed one time only, at the start of P63. |
| 11. | RANGE: | Range to the Landing Site
The slant range from the LM to the current landing site. |
| 12. | $\theta_{e\ell}$: | LPD Angle
The depression angle of the current landing site referenced to the Landing Point Designator. |

The computations required for the display quantities H , H , v_P , V_M , TFI, and Δh are shown in the State-Vector-Update Routine information-flow diagram. TTT, ΔT_{REDES} , v_{HF} , OUTOFPLN, RANGE and $\theta_{e\ell}$ are shown computed in the Guidance and Control Routine.

Analog Display Parameters

The LGC is required to drive four analog displays in all landing-maneuver phases, if selected to do so by the astronaut. The various analog-display quantities are the following:

- 1.) h : Altitude
The estimated local altitude of the vehicle as derived from the difference between the current position vector (r_P) and the landing-site radius magnitude (r_{LS}). The LGC drives a digital tape meter with altitude scaled at 2,345 feet/bit over a range of 0-60,000 feet.

- 2.) v_V : Altitude Rate
The component of the vehicle's velocity that lies along the local-vertical direction (u_{hP}). The LGC drives a digital tape meter with this parameter scaled at 0.5 ft/sec per bit over a range of ± 500 ft/sec.

- 3.) v_{HF} ,
 : Forward and Lateral Velocity
 v_{HL} : Orthogonal components of the horizontal velocity of the vehicle with respect to the moon, which are essentially parallel and perpendicular to the X - Z plane of the vehicle (for small pitch and roll angle displacements). The coordinates and display requirements for these velocity components are defined in ICD LIS-540-10001. The scaling of these components is for a range of ± 200 ft/sec.

The altitude (h) and altitude rate (v_V) are computed and displayed two times/second, alternating them every 0.25 second. Forward and lateral velocity (v_{HF} and v_{HL}) are computed and displayed four times/second.



The displays are kept up-to-date by reading the PIPA-output data at certain special intermediate times (t_i). The vehicle's velocity with respect to the moon at these intermediate times (\underline{v}_{MP_i}) is computed from the relations:

$$\underline{v}_{MP_i} = \underline{v}_{MP_{n-1}} + \underline{g}_{P_{n-1}} (t_i - t_{n-1}) + (\Delta\tilde{\underline{v}}_{P_i} + \Delta\tilde{\underline{v}}_{P_{n-1}}) \quad (t_n \leq t_i < t_j)$$

$$\underline{v}_{MP_i} = \underline{v}_{MP_n} + \underline{g}_{P_n} (t_i - t_n) + (\Delta\tilde{\underline{v}}_{P_i}) \quad (t_j \leq t_i < t_{n+1})$$

(3.4.8.1)

where the subscript i refers to the display-computation time, the subscript n refers to the normal PIPA-output processing time, and the subscript j refers to the actual time that the cycle through the State-Vector-Update Routine is completed.

Two different relations are required because the computations of velocity (\underline{v}_{MP}) and gravitational acceleration (\underline{g}_P) for time t_n are not completed until a later time t_j (greater than t_n). The PIPA-output registers, however, are read and then set to zero at the normal PIPA-output processing time (t_n). For this reason, it is necessary to save and use the preceding-cycle PIPA-output data $\Delta\tilde{\underline{v}}_{P_{n-1}}$ until the cycle through the State-Vector-Update Routine is completed (t_j), as shown in the first of the preceding two relations.

The vertical velocity displayed at the intermediate time (t_i), i. e., v_{V_i} is given by the relations*:

$$v_{V_i} = \underline{v}_{MP_i} \cdot \underline{u}_{hP_{n-1}} + v_{HOR_{n-1}} (t_i - t_{n-1}) \quad (t_n < t_i < t_j)$$

$$v_{V_i} = \underline{v}_{MP_i} \cdot \underline{u}_{hP_n} + v_{HOR_n} (t_i - t_n) \quad (t_j \leq t_i < t_{n+1})$$

(3.4.8.2)

where the subscripts i , j , and n refer to the times mentioned earlier in this section. It should be noted here that the local-vertical unit vectors (\underline{u}_{hP_n} and $\underline{u}_{hP_{n-1}}$) correspond to the normal PIPA-processing times (t_n and t_{n-1}). For this reason, a correction term shown above is required to account for vehicle motion (and

* v_{HOR} is computed in the State-Vector-Update Routine

rotation of \underline{u}_{hP} during the interval from the PIPA-processing time (t_n or t_{n-1}) to the display time (t_i).

The vehicle altitude displayed at intermediate times is given by the relations:

$$h_i = h_{n-1} + v_{V_{n-1}} (t_i - t_{n-1}) \quad (t_n \leq t_i < t_j)$$

$$h_i = h_n + v_{V_n} (t_i - t_n) \quad (t_j \leq t_i < t_{n+1})$$

(3.4.8.3)

where the altitude (h_n or h_{n-1}) and vertical velocity (v_{V_n} or $v_{V_{n-1}}$) are from the State-Vector-Update-Routine.

The forward and lateral velocity components (v_{HF} and v_{HL}) are computed from the same equations shown in the State-Vector-Update Routine, except that the intermediate time horizontal velocity (\underline{v}_{MP_i}) is used in place of that for the normal PIPA-output processing time (v_{MP} or v_{MP_n}). For convenience, these relations are repeated below.

$$v_{HZ_i} = \underline{v}_{MP_i} \cdot \underline{u}_{HZP}$$

$$v_{HY_i} = \underline{v}_{MP_i} \cdot \underline{u}_{HYP}$$

$$v_{HF_i} = v_{HZ_i} \cos AOG_i - v_{HY_i} \sin AOG_i$$

$$v_{HL_i} = v_{HZ_i} \sin AOG_i + v_{HY_i} \cos AOG_i$$

where the subscript i is used to indicate quantities determined for the display times (t_i). The unit vector \underline{u}_{HZP} is for the PIPA-processing time. The angle AOG is the stable member outer gimbal angle.

5.3.4.9 Landing Confirmation Routine and Post-Landing Sequences

When the lunar probe indicates contact with the lunar surface, the crew shuts down the DPS and waits for touchdown. When touchdown has been confirmed, the Landing Confirmation Routine (P68) is entered via the DSKY to terminate all of the landing maneuver routines. The Average-G Routine is allowed to cycle to the next PIPA processing time to establish the final landing condition state vector. The lunar surface flag is set, and the final LM position r_P is transformed to the Moon-Fixed Coordinate System as r_{LS} , and stored for following prelaunch programs. While on the surface, the Planetary Inertial Orientation Subroutine is used to advance the landing site position vector r_{LS} to designated times for computations performed in Basic Reference Coordinates involving the LM state vector while on the lunar surface.

The vehicle attitude in lunar-fixed coordinates is determined as in the Lunar Surface Alignment Program P57 and stored. Before PGNCS power down, an IMU fine alignment is made and the vehicle attitude redefined and stored in place of the previous attitude. These vehicle attitude vectors (the body X- and Z-axis directions) are used in the following IMU alignment operations. An AGS initialization operation is normally performed prior to PGNCS power down. The LGC computed latitude, longitude and final site altitude are displayed on the DSKY.

(Pages 5.3-122, 5.3-123 and 5.3-124 intentionally left blank.)

5. 3. 5 POWERED ASCENT GUIDANCE

5. 3. 5. 1 Guidance Objective

The objective of the Powered Ascent Guidance Program (P-12) is to control the LM ascent maneuver to achieve a specified velocity vector at a desired altitude relative to the launch site and desired distance out of the CSM orbital plane. In order to control the ascent maneuver to these three velocity and two position injection constraints, an explicit guidance concept is used which employs a linear control law. The engine configuration allows control of the direction of thrust only; thus explicit control of only radial (R) and cross range (Y) position and velocity is possible. The desired velocity in the down range (Z) direction is achieved by terminating the thrust at the proper time. The development of the ascent guidance equations is such that the best performance is achieved when the required velocity change is in the down range (Z) direction. In the lunar launch or abort maneuvers, controlled by this program, all injection position and velocity parameters are therefore controlled with the exception of the position along the velocity vector, referred to as the down range position.

The Powered Ascent Guidance Program can control LM ascent maneuvers initiated from non-coplanar launch conditions to be coplanar with the CSM orbital plane at injection, or parallel to the CSM orbital plane at a specified out-of-plane or cross range distance if the astronaut does not wish to remove all of the launch out-of-plane distance during the ascent maneuver. The program is also designed to achieve ascent injection under some off nominal APS thrust conditions (e. g. a single APS helium tank failure), and to allow for RCS injection in cases of premature APS shut down.

The Powered Ascent Guidance Program consists of three major phases.

- 1) Pre-Ignition Phase
- 2) Vertical Rise Phase
- 3) Ascent Guidance Phase

The computational procedures (Section 5.3.5.9) presented in this section are also used during abort maneuvers initiated from the powered lunar landing maneuver. These abort maneuvers can use either the DPS, APS, or a staged combination of the two to achieve the desired abort injection conditions. The abort programs and targeting used to control abort maneuvers using the Powered Ascent Guidance are presented in Section 5.4.3.

The Powered Ascent Guidance Program uses two special coordinate systems in addition to the coordinate systems defined in Section 5.1.4.

One is an instantaneous local vertical coordinate system which is used in the velocity control portion of the ascent guidance computations. This coordinate system was selected because the radial rate and horizontal components of velocity for a vehicle in orbit remain nearly fixed in the absence of thrust acceleration.

The other is an inertial moon centered pseudo coordinate system in which one axis is an arc length. This system is referred to as the Target Coordinate System.

Both coordinate systems are illustrated in Fig. 3.5-1 and are defined as follows.

Target Coordinate System

The target coordinate system is referenced to the CSM orbital plane and initial LM position by:

$$\begin{aligned}\underline{Q} &= \text{UNIT} (\underline{v}_C \times \underline{r}_C) \\ \underline{S} &= \text{UNIT} (\underline{r}_0 \times \underline{Q}) \\ \underline{P} &= \underline{Q} \times \underline{S}\end{aligned}\tag{3.5.1}$$

where \underline{r}_C and \underline{v}_C are the CSM position and velocity vectors, and \underline{r}_0 is the initial LM position vector at t_{IG} . The vehicle Y position is measured from the PS plane as shown in Fig. 3.5-1 as an arc length.

Local Vertical Coordinate System

The local vertical coordinate system of Fig. 3.5-1 is defined by:

$$\begin{aligned}\underline{u}_R &= \text{UNIT}(\underline{r}) \\ \underline{u}_Z &= \text{UNIT}(\underline{u}_R \times \underline{Q}) \\ \underline{u}_Y &= \underline{u}_Z \times \underline{u}_R\end{aligned}\tag{3.5.2}$$

where \underline{r} is the current LM position vector. The vector \underline{r}_{CO} (Fig. 3.5-1) is the LM position vector at engine cut-off.

5.3.5.3 Input - Output

The following input parameters are required by the Ascent Guidance Program in addition to the CM and LM state vectors normally stored and updated in the LGC.

Input Parameters

1. t_{IG} Ignition time
2. R_D Desired injection radius (Nominally 60,000 ft. larger than the landing site radius)

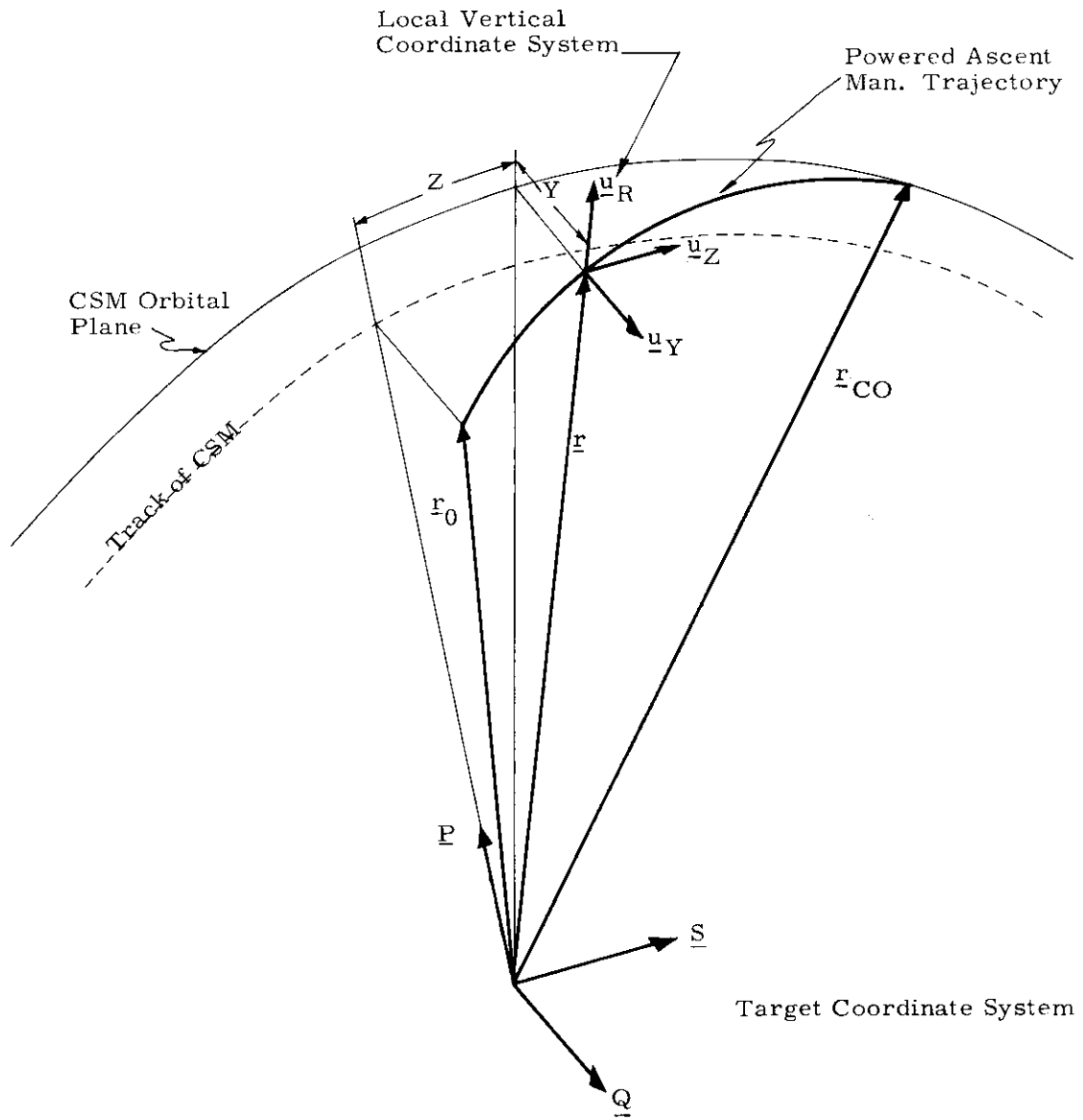


Fig. 3.5-1 Powered Ascent Guidance Coordinate Systems

3. Y_D Desired injection out-of-plane distance, measured from the CSM orbital plane
4. \dot{R}_D Desired injection radial velocity
5. \dot{Y}_D Desired injection cross range velocity
6. \dot{Z}_D Required injection down range velocity

These prestored desired injection parameters will control the ascent maneuver to cut-off in the CSM orbital plane at an altitude of 60,000 ft. with a coplanar velocity vector that will result in a 30 nm apolune altitude. The injection and trajectory apolune altitudes are measured with respect to the launch site radius magnitude.

This orbit is to be used for a "quick, early takeoff," For a normal launch, the astronaut will modify the \dot{R}_D and \dot{Z}_D target parameters to yield a 45 nmi apolune altitude. He also has the option to change Y_D , if the cross-range required to achieve $Y_D = 0$ is large enough to strain the ΔV budget.

Output Parameters

The following parameters are the basic outputs of the Ascent Guidance Program:

1. \underline{u}_{FDP} Commanded thrust attitude vector to FINDCDUW Program in IMU, or platform, coordinates
2. \underline{u}_{WDP} Command vector to FINDCDUW used for vehicle attitude control about the thrust axis in platform coordinates
3. Engine -off Signal
4. The following display parameters (See Section 4, P-12):

Pre-Ignition Phase

- a) t_{IG} Input ignition time initially displayed for verification
- b) CR Cross range distance from the launch site to the CSM orbital plane that will be taken out during the ascent maneuver
- c) \dot{Z}_D Down range injection velocity, as initially targeted
- d) \dot{R}_D Injection radial rate, as initially targeted
- e) FDAI Yaw Angle } The yaw angle that should be present on the FDAI at the end of the vertical rise phase prior to pitch over
- f) FDAI Pitch Angle } The FDAI pitch angle that should be present after pitch over has been completed in the early portion of the ascent guidance phase
- g) TFI Time from Ignition

Vertical Rise and Ascent Phase

- a) v Inertial speed
- b) \dot{h} Altitude Rate
- c) h Altitude or radial position measured from the launch site radius

Callable Displays

- Set 1 { a) t_{go} Time from cut-off
- { b) V_Y Current cross plane velocity \dot{Y}
- Set 2 { c) \underline{V}_{GB} Velocity to gain, in body coordinates



5.3.5.4 Powered Ascent Program P-12 Pre-Ignition Phase

The pre-ignition computation sequence is shown in Fig. 3.5-2. With reference to this figure, after the ignition time, t_{IG} , has been verified by the astronaut, the following is a definition of the parameters initialized (numeric values are found in Section 5.8).

$a_T(\text{APS}) =$ Initial APS thrust acceleration

$\tau(\text{APS}) =$ Initial APS mass to mass flow rate ratio

$\frac{1}{\Delta V} =$ Initial velocity parameter for the initialization of $\frac{1}{\Delta V_1}$, $\frac{1}{\Delta V_2}$, and $\frac{1}{\Delta V_3}$ in the thrust magnitude filter (Section 5.3.5.9)

$\Delta t_{\text{tail-off}}(\text{APS}) =$ A negative time increment used to correct t_{go} for the APS tail-off

$V_e(\text{APS}) =$ APS exhaust velocity

The flag, FLPI, is set to 1 to provide for the use of the Ascent Guidance Equations as a subroutine, predicting the commands at tipover.

Three of the five desired injection target parameters (R_D , Y_D , \dot{Y}_D) are then initialized as shown in Fig. 3.5-2. After the target vector \underline{Q} is computed and the time-to-go, t_{go} , is set to a nominal value of 370 sec., the LM position vector is advanced to the ignition time, t_{IG} , by the Planetary Inertial Orientation Subroutine of Section 5.5.2. The velocity $\underline{v}(t_{IG})$ is updated as a function of the landing or launch position determined by the Planetary Inertial Orientation Subroutine as noted in Fig. 3.5-2 where ω_M represents the lunar inertial rotation rate and the unit vector \underline{u}_M represents the lunar axis of rotation. The remaining two targets, \dot{R}_D and \dot{Z}_D , are obtained from fixed storage.

During the initialization, t_{XO} , the X-axis override time is set to 0. This operation is a remnant from an earlier version of Luminary in which it was possible to enter the Ascent Guidance Equations with FLVP = 0.

The initially-displayed cross-range distance, CR, to be achieved by the ascent maneuver is based on injection coplanar with the CSM ($Y_D = \dot{Y}_D = 0$). If the astronaut does not desire to have the entire launch out-of-plane distance taken out during the ascent maneuver, he can specify how much out-of-plane distance should be removed during the powered ascent by overwriting the initial CR DSKY display. Injection conditions will then be parallel, but non-coplanar with respect to the CSM orbit plane.

The radial and down-range velocity targets are those pre-stored for quick liftoff, and in a normal launch would be modified by the astronaut.

Equation (3. 5. 3) deleted by PCR #270.

With reference to Fig. 3. 5-2, the LM state vector is next extrapolated through a nominal vertical rise phase so that initial ascent guidance phase command acceleration vectors can be calculated for the pre-ignition phase FDAI displays. These ascent phase acceleration vector computations are described in Section 5. 3. 5. 9 and in Fig. 3. 5-3. The FDAI yaw and pitch angle display parameters determined by Eqs. 3. 5. 21 and 3. 5. 22 respectively represent the FDAI yaw angle that will be achieved at the end of the vertical rise phase assuming that attitude control about the thrust axis is completed in this phase, and the FDAI pitch angle that will result early in the ascent guidance phase after pitch-over transients have damped out. The time from ignition, TFI, is the final display computation made in Fig. 3. 5-2.

The final operations during the pre-ignition phase are to extrapolate the LM position vector to a condition 30 seconds before ignition, $t_{IG} - 30$, and establish the LM state vector parameters $\underline{r}(t_{IG} - 30)$ and $\underline{v}(t_{IG} - 30)$. The vertical rise phase control flag, FLVP, is set, and the Average-G Routine of Sec. 5. 3. 2 is started at $t_{IG} - 30$. The Average G Routine is cycled at 2 second intervals initiating the computation sequence illustrated in Fig. 3. 5-3 at point (B). The APS ignition is set for t_{IG} .

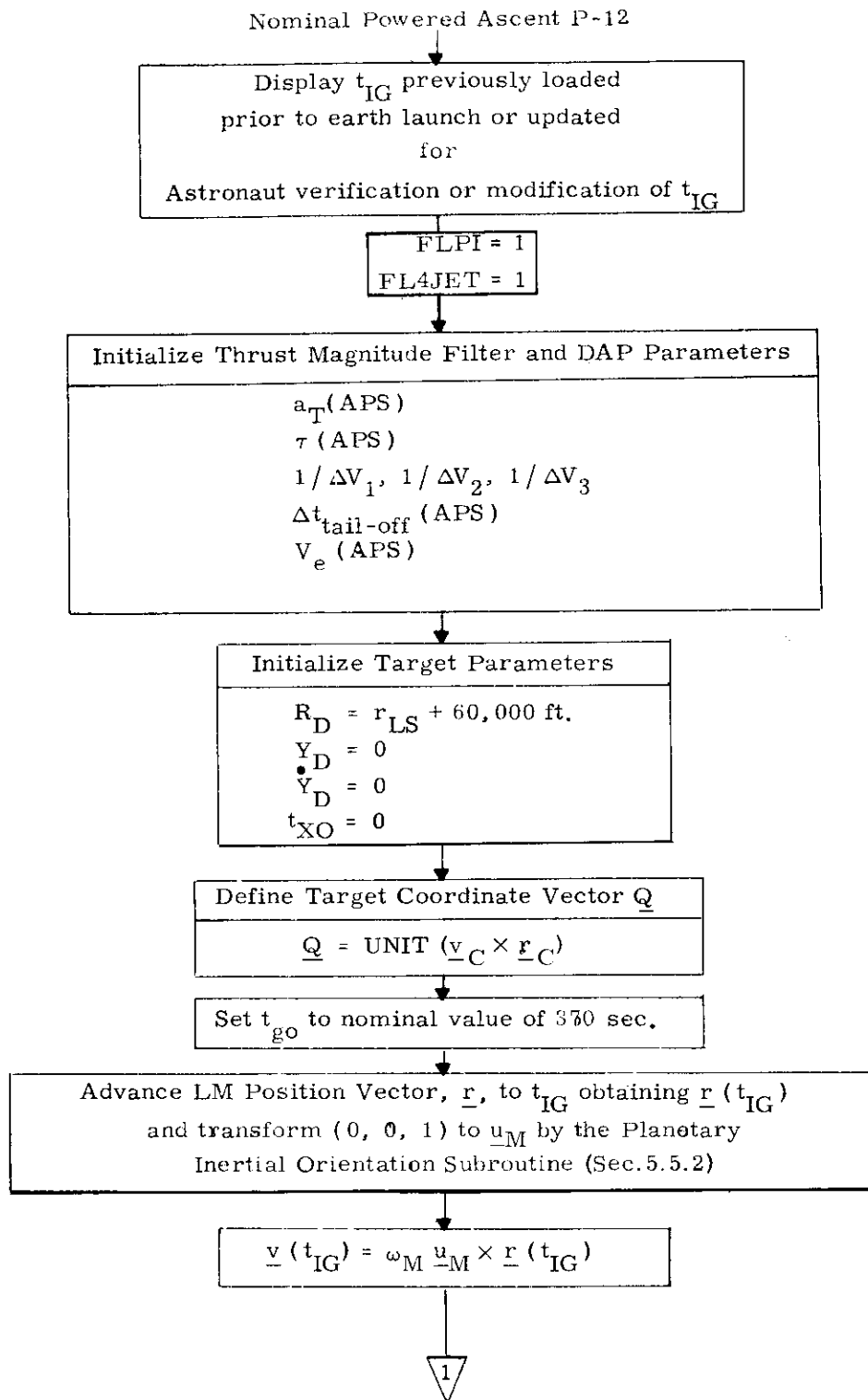


Fig. 3.5-2 Powered Ascent Guidance P-12 Pre-Ignition Mode
(Page 1 of 2)

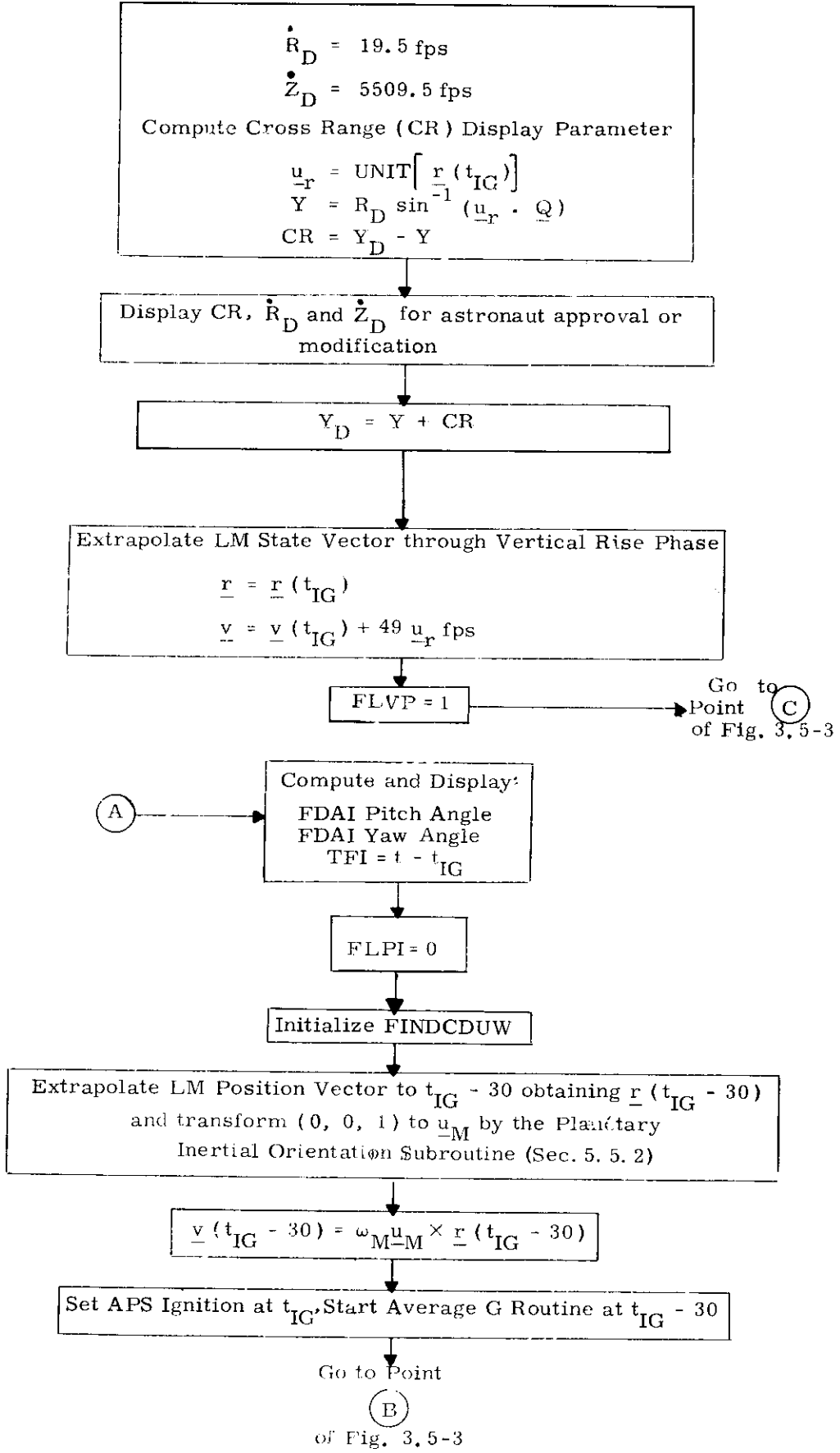


Fig. 3.5-2 Powered Ascent Guidance P-12 Pre-Ignition Mode (Page 2 of 2)

5.3.5.5 Powered Ascent Program P12 Vertical Rise Phase Control

The computation procedures during the vertical rise phase are described in Section 5.3.5.9. The proper path through these procedures is obtained by setting control flag FLVP = 1. Most of the procedures are explained in Section 5.3.5.9; the remainder is described below.

As shown on page 4 of Fig. 3.5-3 the nominal attitude X-axis override DAP mode is inhibited until 10 seconds after the vertical rise phase has been completed.

The vehicle attitude is controlled by FINDCDUW (Section 5.3.7) about the thrust or vehicle X axis such that the vehicle X - Z plane contains \underline{u}_{WDP} , the desired window pointing direction. During the vertical rise phase, \underline{u}_{WDP} is defined to be the horizontal component of the desired acceleration computed by the guidance equations, \underline{a}_T , and the commanded thrust direction \underline{u}_{FDP} is along the local vertical vector. These two vectors, \underline{u}_{FDP} and \underline{u}_{WDP} , and the yaw-control mode flag FLAUTOX are the required input parameters to the LM digital autopilot interface subroutine designated FINDCDUW. The altitude check ($h \geq 25,000$) shown in Fig. 3.5-3 page 4 is used in the abort programs P-70 and P-71 (Section 5.4.3) to add an additional altitude constraint on the vertical rise phase. ●
●
●

Powered Ascent Program P12 Ascent Guidance
Phase Control

The Ascent Guidance Phase is the third and final phase of the powered ascent program P-12 and is initiated after the vertical rise phase is terminated. The computation procedures are described in Section 5.3.5.9. The proper path through these procedures is obtained by setting control flag FLVP = 0.

The flag, ROTFLAG, is provided for abort maneuver control, and is zero during P-12 operation. The operation of the equations with ROTFLAG = 1 is discussed in Section 5.4.3.1.4.

During the Ascent Guidance Phase the vehicle thrust attitude is commanded to be along the desired acceleration vector \underline{a}_T . The vehicle attitude about the thrust axis is commanded such that the LM windows or +Z axis will be in the downward direction. The X-axis override DAP mode is not activated until 10 seconds (t_{XO}) after the termination of the vertical rise phase to allow for attitude stabilization during the initial pitch-over maneuver. The astronaut can then manually control the vehicle yaw attitude for the duration of the powered ascent maneuver if desired. If the X-axis override mode is not exercised, the existing attitude orientation will be maintained.

The principal outputs of the program are the two attitude command vectors \underline{u}_{FDP} and \underline{u}_{WDP} to the DAP interface subroutine FINDCDUW (Section 5.3.7).

If the APS is prematurely shut down prior to ascent injection or if it is desired to trim-up the ascent injection maneuver after nominal APS termination, the Ascent Guidance Program can be used to aid in the manual control of the RCS to achieve the desired velocity injection conditions. The computation sequence used in this type of operation is described in Section 5.3.5.9 and is illustrated in Fig. 3.5-3 page 1 with control flag FLRCS = 1. At APS termination the ΔV Monitor Routine places the vehicle in the attitude hold mode. The basic computation shown in Fig. 3.5-3 for the RCS mode of operation is the determination of the velocity-to-be-gained vector, \underline{v}_G . This velocity vector is transformed from IMU platform or stable member coordinates to vehicle body, or navigation base, coordinates (Section 5.6.3.2) and displayed so that the astronaut can manually complete the ascent maneuver. The RCS maneuver is done with 4 RCS jets, automatically selected by the setting of FL4JET at the beginning of P12 and P70/P71.

5. 3. 5. 8

Abort Guidance from Lunar Landing Maneuvers

The DPS Abort Program P-70 and APS Abort Program P-71 control abort programs initiated during the powered lunar landing maneuver. The basic computation sequence used by both P-70 and P-71 is the Powered Ascent Guidance Computations of Section 5. 3. 5. 9. The initialization of programs P-70 and P-71 is described in Section 5. 4. 3 and Fig. 4.3-1.

The guidance computation procedures presented in this section and Fig. 3. 5-3 determine the desired thrust direction and time of thrust termination for both the vertical rise and ascent guidance phases of program P12 (Powered Ascent Guidance) and programs P70 and P71 (DPS and APS aborts).

The figure describing the computational flow of these procedures is Fig. 3. 5-3 (pp. 1-4). The following set of switches establish the proper path thru these procedures for the various users:

FLRCS	{	1	RCS Injection Mode
		0	Not the RCS Injection Mode
FL4JET	{	1	4 RCS jets used in translation
		0	2 RCS jets used in translation
P7071FLG	{	1	Major mode is P70 or P71
		0	Major mode is P12
ROTFLAG	{	1	Rotation control is imposed
		0	FINDCDUW commands shortest rotation

FLVP	{	1	Vertical rise phase
		0	Not the vertical rise phase
FLPC	{	1	Injection position control not desired
		0	Injection position control desired
FLPI	{	1	P-12 pre-ignition phase
		0	Not the P-12 pre-ignition phase
FLDOC	{	0	LM is not docked to CSM

It should be noted that FLENG1 and FLDOC are not manipulated by P12, P70 or P71. In addition to the above control flags, the control times t_2 and t_3 of Fig. 3.5-3 are used to regulate the guidance functions as follows:

$t_2 = 2$ sec: Represents the time interval before engine cut-off during which all guidance parameters are held at their last computed values.

$t_3 = 10$ sec: The time interval before engine cut-off during which the guidance parameters B and D that control the injection position in the R and Y directions are set to zero. During this 10 second interval all position control is terminated and the guidance system attempts to achieve the desired injection velocity conditions until t_2 .

With reference to Fig. 3.5-3, the computation sequence is initiated at point (B) by the Average-G Routine every two second computation cycle. The following parameter computations are made in the sequence illustrated in Fig. 3.5-3 for the APS lunar ascent maneuver.

Thrust Filter Bypass

The existence of an engine on command is first checked and the velocity increment ΔV measured by the PIPA's over the last computation cycle is compared with a prestored minimum value ΔV_{\min} . If the measured ΔV is less than ΔV_{\min} , the speed, altitude rate and altitude are displayed and current attitude is maintained. This check is in addition to that of the ΔV Thrust Monitor Subroutine of Section 5.3.3.6. The value of ΔV_{\min} used in Luminary is 3.56 m/sec.

Thrust Filter Computations

Assuming a successful engine thrust test, the thrust filter computation is processed next. The vehicle velocity increments, ΔV , measured by the PGNCs IMU integrating accelerometers (PIPA's) include noise not only from the main engine, but also from the RCS jets used for attitude control. In order to estimate an accurate thrust acceleration performance (a_T and τ), several successive PIPA ΔV readings are averaged. Under the assumption of a constant mass flow engine, $1/a_T$ is a linear function of time. The ratio of mass to mass flow rate (τ) is then computed from four successive PIPA readings ΔV_0 , ΔV_1 , ΔV_2 and ΔV_3 as follows:

$$\Sigma = \frac{1}{\Delta V_0} + \frac{1}{\Delta V_1} + \frac{1}{\Delta V_2} + \frac{1}{\Delta V_3} \quad (3.5.4)$$

$$\tau' = \frac{\Sigma}{4} V_e \Delta t - 2 \Delta t \quad (3.5.5)$$

where Δt is the computation interval. In Eq. (3.5.5) the first term represents the estimate of τ' at the center of the series of time increments for which ΔV was measured, and the second term updates the estimate to current time. Additional smoothing is included by averaging this new value of τ' with the previous estimate of τ , updated by Δt .

$$\tau = \frac{1}{2} (\tau' + \tau - \Delta t) \quad (3.5.6)$$

The exhaust velocity V_e is prestored in the LGC. In the pre-ignition computation for lunar launch the three velocity increments $\frac{1}{\Delta V_1}$, $\frac{1}{\Delta V_2}$, and $\frac{1}{\Delta V_3}$ are prestored initialization constants. When a new $\frac{1}{\Delta V}$ is computed it is stored in the $1/\Delta V_3$ register, and the older values are shifted down. The vehicle thrust acceleration magnitude is then computed by

$$a_T = \frac{V_e}{\tau} \quad (3.5.7)$$

With reference to page 1 of Fig. 3.5-3, the following parameters are computed after the thrust filter computation.

Local Vertical Coordinate System

$$\begin{aligned}\underline{u}_R &= \text{UNIT}(\underline{r}) \\ \underline{u}_Z &= \text{UNIT}(\underline{u}_R \times \underline{Q}) \\ \underline{u}_Y &= \underline{u}_Z \times \underline{u}_R\end{aligned}\tag{3.5.8}$$

Local Velocity Components:

$$\begin{aligned}\dot{R} &= \underline{v} \cdot \underline{u}_R \\ \dot{Y} &= \underline{v} \cdot \underline{u}_Y \\ \dot{Z} &= \underline{v} \cdot \underline{u}_Z\end{aligned}\tag{3.5.9}$$

Local Position Computation

The radial and cross range positions (Fig. 3.5-1) are computed by

$$\begin{aligned}R &= r \\ Y &= R_D \sin^{-1}(\underline{u}_R \cdot \underline{Q})\end{aligned}\tag{3.5.10}$$

where the sine, $\underline{u}_R \cdot \underline{Q}$, is used as an approximation to the angle.

Effective Gravity Computation

$$g_{\text{eff}} = \frac{H^2}{r^3} - g_n \quad (3.5.11)$$

where $H^2 = [|\underline{r} \times \underline{v}|]^2$ and g_n is the magnitude of gravity as computed by the Average-G Routine of Section 5.3.2.

Velocity to be Gained Computation

The velocity-to-be-gained vector \underline{v}_G is next computed as shown in Fig. 3.5-3, page 1, and then transformed from the stable member coordinates in which the local vertical system is defined to the vehicle body or navigation base coordinate system by the transformation matrix [SMNB] as defined in Section 5.6.3.2. This velocity-to-be-gained vector referenced to navigation base coordinates is used for display parameters in the RCS injection mode.

Time-to-Go Estimation

As shown in Fig. 3.5-3, page 2, the time-to-go, t_{go} , is determined from a truncated series expansion of the exponential expression for $t_{go}(v_G)$:

$$t_{go} = \tau \left(1 - e^{-\frac{v_G}{V_e}} \right) \quad (3.5.12)$$

The series expansion used to approximate the above expression is truncated at the second order term, and the coefficient of the second order term is replaced by a parameter, K_T , which could be adjusted to over or under estimate t_{go} in the early phases of the maneuver.

$$t_{go} = \tau \frac{v_G}{V_e} \left(1 - \frac{1}{2} \frac{v_G}{V_e} \right) + \Delta t_{tail-off} \quad (3.5.13)$$

where

$$\tau = \frac{\text{mass}}{\text{mass flow rate}}$$

$$V_e = \text{Exhaust velocity}$$

$$\Delta t_{Tail-off} = \text{A negative time increment used to correct } t_{go} \text{ for the thrust tail-off.}$$

For P12 the value used is $\Delta t_{Tail-off}$ (APS)

When the guidance is operating in the RCS mode, the time-to-go computation is simplified to

$$t_{go} = V_G / A_T \text{ (RCS)}$$

where A_T (RCS) is an assumed, constant, acceleration corresponding to 4 RCS jets in a dry LM. In situations where only 2 jets are used this leads to an underestimate of t_{go} . However, t_{go} is only used in the estimation of the gravity effect, and hence, is only a second order factor. The use of a 2-jet acceleration could lead to overflow in the computation of t_{go} .

5.3-147

Revised LUMINARY

Added GSOP # 567 PCR # 846 Rev 1 Rev. 7 Date 11/69

Engine-off Test

On page 2 of Fig. 3.5-3 for control phases other than the vertical rise phase, the engine-off command is set for $t + t_{go} - \Delta t_c$ when t_{go} first falls below 4 sec, where Δt_c is the computation time delay between the reading of the PIPA's and the sending of the commands.

Guidance Parameter Computations

The basic guidance concept used for the powered ascent is represented by the equations:

$$a_{TR} = a_T [A + B(t - t_o)] - g_{eff} \quad (3.5.14a)$$

$$a_{TY} = a_T [C + D(t - t_o)]$$

where a_{TR} and a_{TY} are the required accelerations in the radial and cross range directions respectively, a_T is the filtered thrust acceleration, and t_o is the time at which the guidance coefficients A, B, C and D are computed.

There is a significant delay between the establishment of the state vector on which the commands are based and the sending of the commands by the autopilot. In addition, since the commands will be fixed for a 2-second period, while the guidance equations assume components varying linearly with time, the value of t should be selected in the middle of the 2-second command period, so that the actual commands are equal to the average of the assumed linear commands.

These delay times are all lumped together in the increment $(t - t_o)$, which is chosen to provide good guidance performance. In Luminary the present value of $t - t_o$ is fixed at 1 sec.

For satisfactory scaling within the LGC, the equations are reorganized into the form

$$\begin{aligned} a_{TR} &= \frac{1}{\tau} (A + [t-t_o] B) - g_{eff} \\ a_{TY} &= \frac{1}{\tau} (C + [t-t_o] D) \end{aligned} \quad (3.5.14b)$$

where:

$$B = [D_{21} (\dot{R}_D - \dot{R}) - (R_D - R - \dot{R}t_{go})] / t_{go} E \quad (3.5.15)$$

$$D = [D_{21} (\dot{Y}_D - \dot{Y}) - (Y_D - Y - \dot{Y}t_{go})] / t_{go} E \quad (3.5.16)$$

$$A = - D_{12} B - (\dot{R}_D - \dot{R}) / L \quad (3.5.17)$$

$$C = - D_{12} D - (\dot{Y}_D - \dot{Y}) / L \quad (3.5.18)$$

where the parameters L, D_{12} , D_{21} , and E are defined on page 2 of Fig. 3.5-3 as previously mentioned. When the time-to-go, t_{go} , falls below 10 seconds (t_3) the rate coefficients B and D are set to zero, and the injection position control is terminated. When t_{go} falls below 2 seconds (t_2) no further guidance coefficient computations are made and the previous values are held for the remainder of the maneuver.

In order to make the response of PGNCS similar to that of the back up system, AGS, the pitch rate parameter, B, is limited by

$$-0.1 \text{ fps}^3 \tau \leq B \leq 0$$

It should be noted that the restriction of B to non-positive values caused the injection to occur above the specified 60,000 ft. altitude.

Command Thrust Acceleration Vector Computation

The required acceleration components a_{TR} and a_{TY} are computed as shown on page 3 of Fig. 3.5-3 for the R and Y injection parameters respectively. These two required acceleration components are then combined as

$$\underline{a}_H = a_{TY} \underline{u}_Y + a_{TR} \underline{u}_R \quad (3.5.19)$$

$$a_H = |\underline{a}_H| \quad (3.5.20)$$

and compared with a_T , where a_T is available thrust acceleration. If a_H is greater than a_T , the commanded \underline{a}_H is reduced by a factor a_T/a_H and the Z component of \underline{a}_T is set to 0. If the required a_H is smaller than a_T , the remainder $(a_T^2 - a_H^2)^{1/2}$ is directed along \underline{Z} and \underline{a}_H is unchanged. The sense of the Z-component of the computed thrust acceleration vector, \underline{a}_T , is determined by the sign of the velocity-to-be-gained in this direction $\text{SGN}(\hat{Z}_D - \dot{Z})$. The commanded thrust acceleration vector, \underline{u}_{FDP} , is set equal to the computed vector, \underline{a}_T , subject to the modifications discussed in Section 5.3.5.5.

FDAI Angle Computations

The Flight Direction Attitude Indicator (FDAI) yaw angle at the end of the vertical rise phase, and the pitch angle after pitch-over and the start of the ascent guidance phase are required for display and monitoring purposes in the pre-ignition phase of program P-12 (Section 5.3.5.4). These FDAI display parameters are computed as follows, based on the lunar launch alignment (Section 5.1.4.2) of the IMU:

$$\text{FDAI Yaw Angle} = \sin^{-1} \left[\frac{a_{TY}}{(a_{TP}^2 + a_{TY}^2)^{1/2}} \right] \quad (3.5.21)$$

if $(a_{TP}^2 + a_{TY}^2)^{1/2} = 0$ then Yaw Angle = 0

$$\text{FDAI Pitch Angle} = -\cos^{-1} [\text{UNIT}(\underline{u}_{FDP}) \cdot \underline{u}_R] \quad (3.5.22)$$

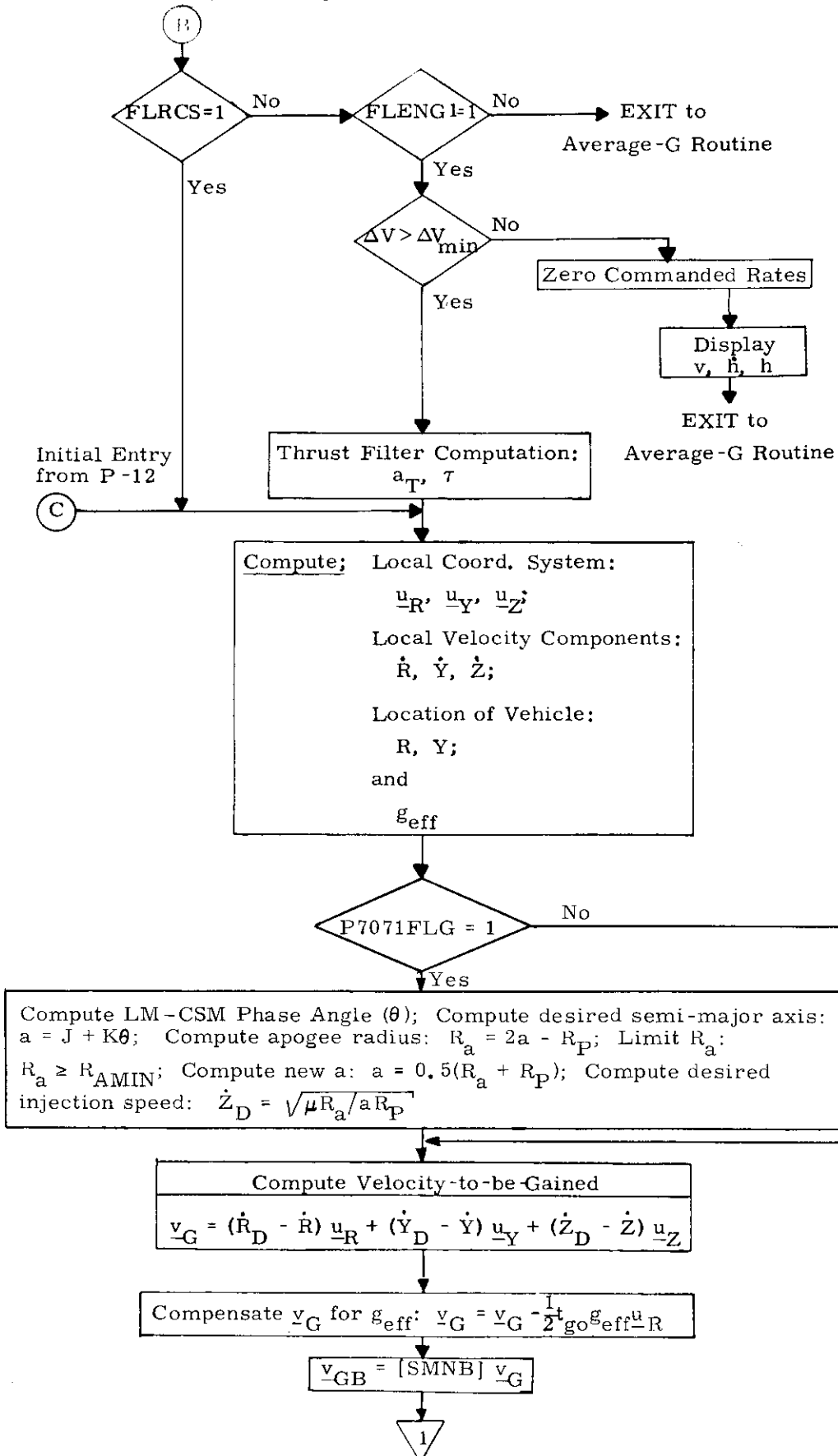


Figure 3.5-3 Ascent Guidance Computation Sequence (page 1 of 4)

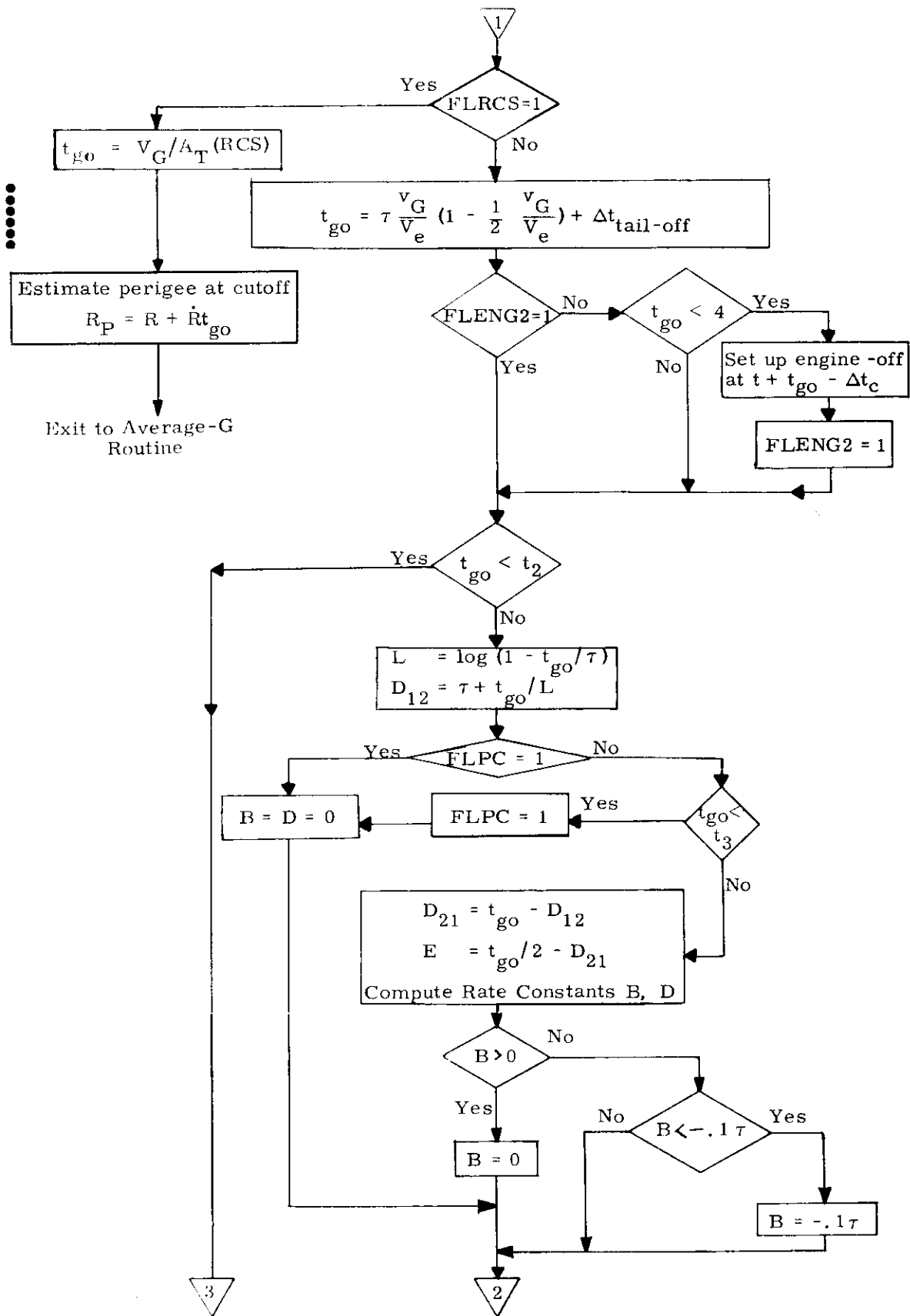


Figure 3.5-3 Ascent Guidance Computation Sequence (page 2 of 4)

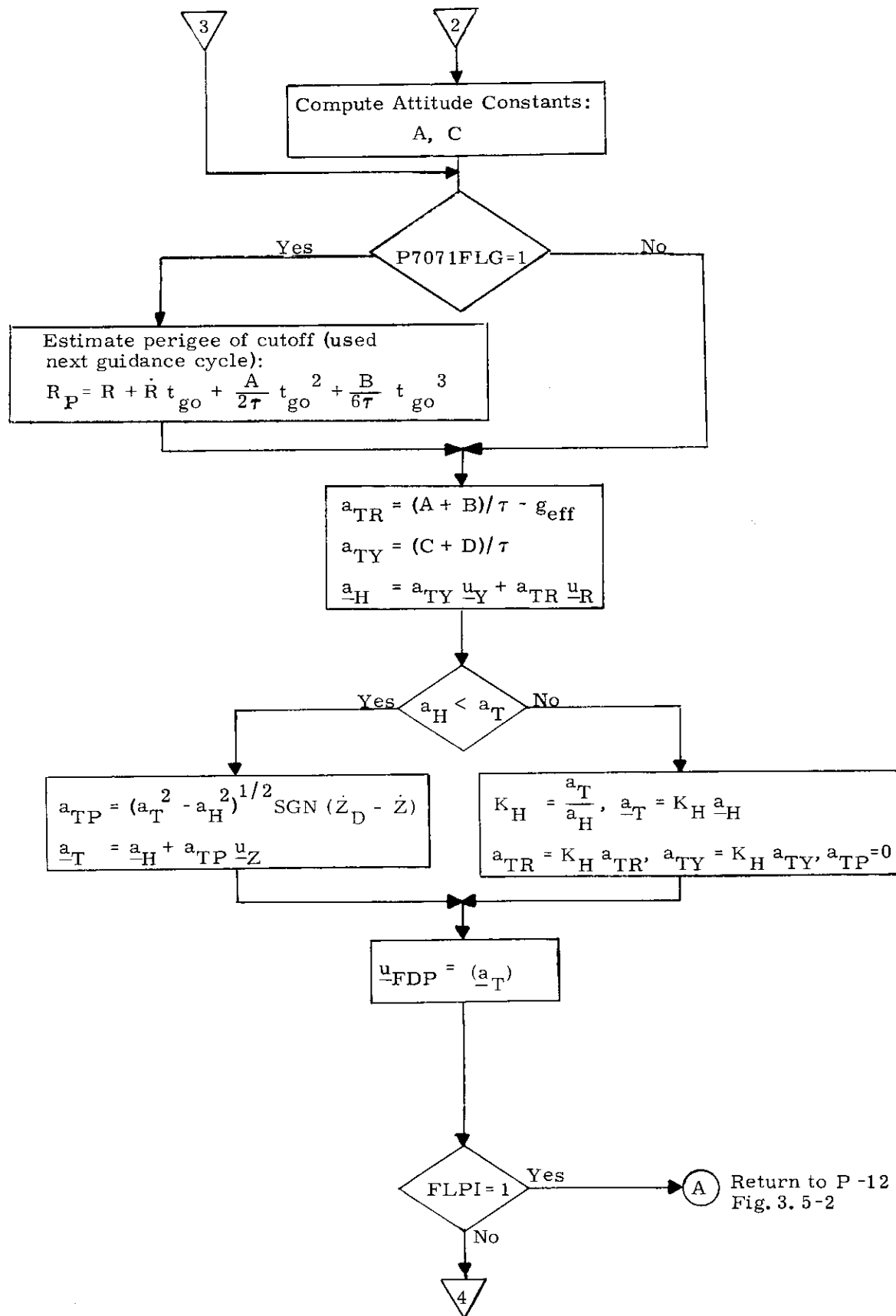


Figure 3.5-3 Ascent Guidance Computation Sequence
(page 3 of 4)

5.3-154

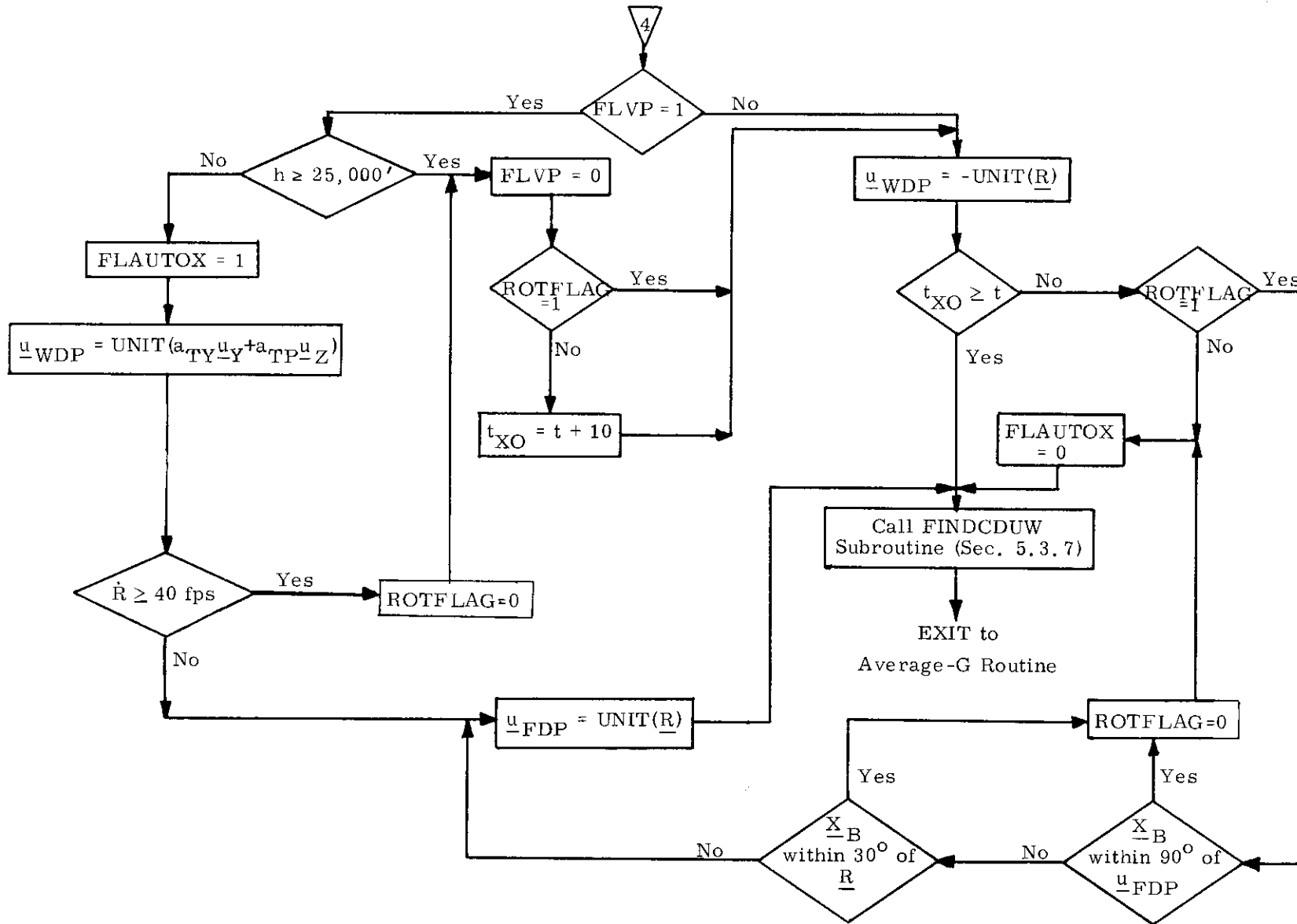


Figure 3.5-3 Ascent Guidance Computation Sequence
(Page 4 of 4)

5.3.6 THRUST MONITOR PROGRAM

The Thrust Monitor Program, P-47, is used during manual or non-PGNCS controlled maneuvers to monitor and display the velocity change applied to the vehicle. The program first suspends state vector updating by the radar data (resets the update and track flags), and advances the vehicle state vector to the current time by the Coasting Integration Routine of Section 5.2.2. This operation is continued until the state vector is advanced several seconds ahead of the current time as described in Section 5.3.8. The Average-G Routine of Section 5.3.2 is then initiated to allow thrusting to be started as soon as possible. The Average-G Routine is left on until the program is terminated after completion of the maneuver. The primary output of P-47 is the measured maneuver $\Delta \underline{V}$ in vehicle coordinates as described in Section 4.

The Thrust Monitor Program is normally used during the manually controlled terminal rendezvous phase. During this mission phase the Rendezvous Display Routine R-31 is normally called to display range, range rate and the line-of-sight angle θ . The operation of R-31 with the Average-G Routine of P-47 is described in Section 5.6.7.1.

5.3.7 FINDCDUW ROUTINE

5.3.7.1 Introduction

The purpose of this section is to describe the operation and implementation of the FINDCDUW Routine.

FINDCDUW is the interface routine between the various powered flight guidance programs and the digital autopilot. FINDCDUW receives from the guidance programs thrust and window pointing commands and provides the autopilot with gimbal angle increments, commanded attitude rates, and commanded attitude lag angles (which account for the angles by which the body will lag behind a ramp command in attitude angle due to the finite angular accelerations available). FINDCDUW aligns the estimated thrust vector with the unit thrust command vector, and, when in the automatic X axis control mode, aligns the +Z half of the LM ZX plane with the window command vector.

FINDCDUW is designed to accomplish the following:

1. To produce the normal, small angle, maneuvers determined by the guidance inputs at essentially constant attitude rate, reaching the terminal attitude at the completion of the two second computation period.
2. To produce the gross maneuvers, which are required at the boundaries between guidance phases, or upon abort, with a continuous, rate-limited attitude profile extending over whatever time duration is required to complete the maneuver.

3. To avoid excessive middle gimbal angle in all maneuvers, never reaching the critical middle gimbal angle during a maneuver in which the terminal middle gimbal angle is less in magnitude than the critical value; to display an alarm code when inputs to FINDCDUW determine a terminal middle gimbal angle in excess of the critical value, and to limit the middle gimbal excursion at the critical value.
4. To permit manual X axis control when the flag FLAUTOX is 0, not interfering when this control is being exercised, not drifting about the X axis when manual control is permitted but not exercised.
5. To provide identical interfaces except with alternate rate limit values when the Command and Service Modules are docked.
6. To use the desired gimbal angles as maintained by the autopilot rather than the actual gimbal angles when transforming measured thrust to bodyaxes to avoid feeding fuel slosh oscillations into the powered flight portion of the system.

To accomplish item 6 above, two coordinate frames are defined. The first, called the current vehicle frame, is computed using the desired gimbal angles, CDUD's, maintained by the digital autopilot. The unit vectors of the current vehicle frame would be identically the body unit vectors if the desired gimbal angles were identically the actual gimbal angles. FINDCDUW provides a thrust direction filter which transforms the measured thrust acceleration vector to current vehicle coordinates and produces a filtered unit thrust vector in this coordinate frame \underline{u}_{FV} . The second frame, called the commanded vehicle frame, is computed using \underline{u}_{FV} , the thrust command \underline{u}_{FDP} , and the

window command u_{WDP} . Since the commanded orientation is coupled to the actual orientation only through the low gain thrust direction filter, fuel slosh is not excited. Also, since the filter operates in the frame defined by the CDUD's, and the CDUD's are driven by the commands from FINDCDUW, the steady-state attitude error is zero.

The concept of FINDCDUW is as follows. The thrust and window pointing commands, and the location of the thrust vector relative to spacecraft axes, determine a required, or terminal, spacecraft attitude. The gimbal angles corresponding to this terminal attitude are computed and inputs to the autopilot are generated which drive each gimbal angle directly from its initial value to its terminal value*.

Provided the attitude is not initially in gimbal lock, and provided the input commands do not yield a terminal attitude in gimbal lock, it is impossible to pass through gimbal lock during the maneuver since the middle gimbal angle is confined to the range between its initial and terminal values.

A simplified flow diagram of the FINDCDUW Routine is shown in Fig. 3.7-1 with nomenclature defined in Section 5.3.7.2.

This routine is called every two seconds, which is the frequency at which the commanded vehicle attitude is computed.

*Except the outer gimbal angle profile may not be monotonic in the geometrically complex case of a large maneuver about multiple axes at substantial middle gimbal angle and with magnitude limiting of the X axis attitude angle change on at least one pass through FINDCDUW.

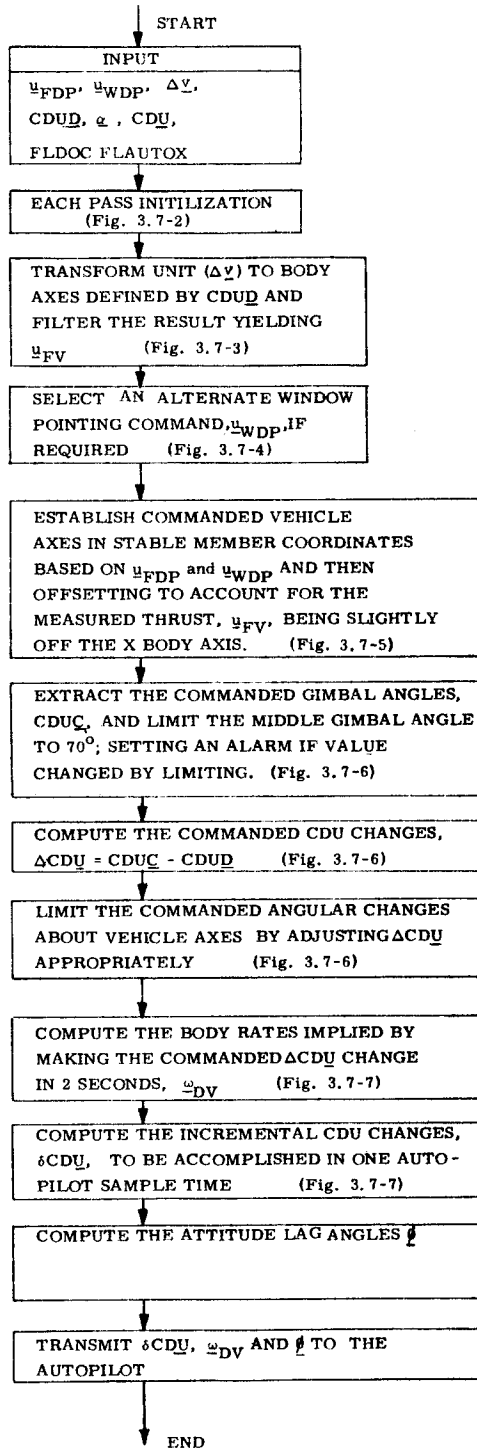


Figure 3.7-1 FINDCDUW Simplified Flow Diagram

5.3.7.2 Nomenclature

\underline{u}_{FDP}	Vector in the desired thrust direction in Platform Coordinate System. Need not be unitized.
\underline{u}_{WDP}	Vector in the desired window direction in Platform Coordinate System. Need not be unitized.
OGABIAS	Outer gimbal angle bias
$\Delta \tilde{v}_P$	PIPA measured velocity change since the last sample period, platform coordinates.
\underline{CDU}	Measured gimbal angles
\underline{CDUD}	The desired gimbal angles as maintained by the autopilot
$\underline{\alpha}$	The angular accelerations which can be achieved by the RCS, (2 Jet)
\underline{u}_{FV}	Filtered unit measured thrust direction vector
GAINFLTR	Gain of thrust direction filter
\underline{CDUC}	Commanded gimbal angles
$\underline{\omega}_{DV}$	Commanded body attitude rates sent to the autopilot
$\delta \underline{CDU}$	Commanded incremental CDU changes sent to the autopilot to occur in one autopilot sample period

FLDOC	LM docked flag = 0 LM not docked ≠ 0 LM docked
FLAUTOX	X axis override flag = 0 override permitted ≠ 0 override not permitted
OVFIND	Computer overflow indicator
PGNCS MODE	LM console switch Yes = PGNCS control No = AGS control
AUTO MODE	LM console switch Yes = Auto Mode No = Attitude Hold Mode or Off Mode
$C_{PV}(\underline{CDUV})$	3×3 matrix which transforms a vector in platform coordinates to the vehicle coordinates defined by \underline{CDUV} (Fig. 3.7-2)
\underline{u}_{ZBP}	Unit Z body axis, platform coordinates (supplied by service routine)
\underline{u}_{XBP}	Unit X body axis, platform coordinates (supplied by service routine)

5. 3. 7. 3 Detailed Flow Diagrams

5. 3. 7. 3. 1 First Pass Initialization

Prior to the first pass thru FINDCDUW in any powered guidance phase, the routine INITCDUW must be called to initialize inputs to FINDCDUW. INITCDUW does the following:

- 1) OGABIAS = 0
- 2) $\underline{u}_{FV} = 1, 0, 0$
- 3) $\underline{u}_{WDP} = 1, 0, 0$

If the user wishes an outer gimbal angle bias, he must subsequently set OGABIAS.

5. 3. 7. 3. 2 Each Pass Initialization

The initialization which must be done each pass thru FINDCDUW amounts to acquisition and preparation of constants and input data (See Fig. 3. 7-2). These are:

1. θ_{XM} , θ_{YM} , θ_{ZM} , the maximum angular changes are selected depending upon whether or not the Command and Service Modules are docked.
2. FLAGOODW is initialized to 0 or 1 according to whether X axis override is permitted or inhibited. This flag indicates whether the unit window command is good for control about the X axis.
3. OVFind, the interpretive overflow indicator, is set to zero.
4. If the switches on the LM console indicate PGNCS MODE and AUTO MODE, flag FLPAUTNO is set to zero and the CDUD's are fetched for the thrust direction filter; otherwise flag FLPAUTNO is set to 1 and the CDU's are fetched. Fetching the CDU's for the thrust direction filter when not in PGNCS MODE maintains the thrust direction filter even when the autopilot does not maintain the CDUD's, and produces no autopilot instability because no autopilot commands are issued.

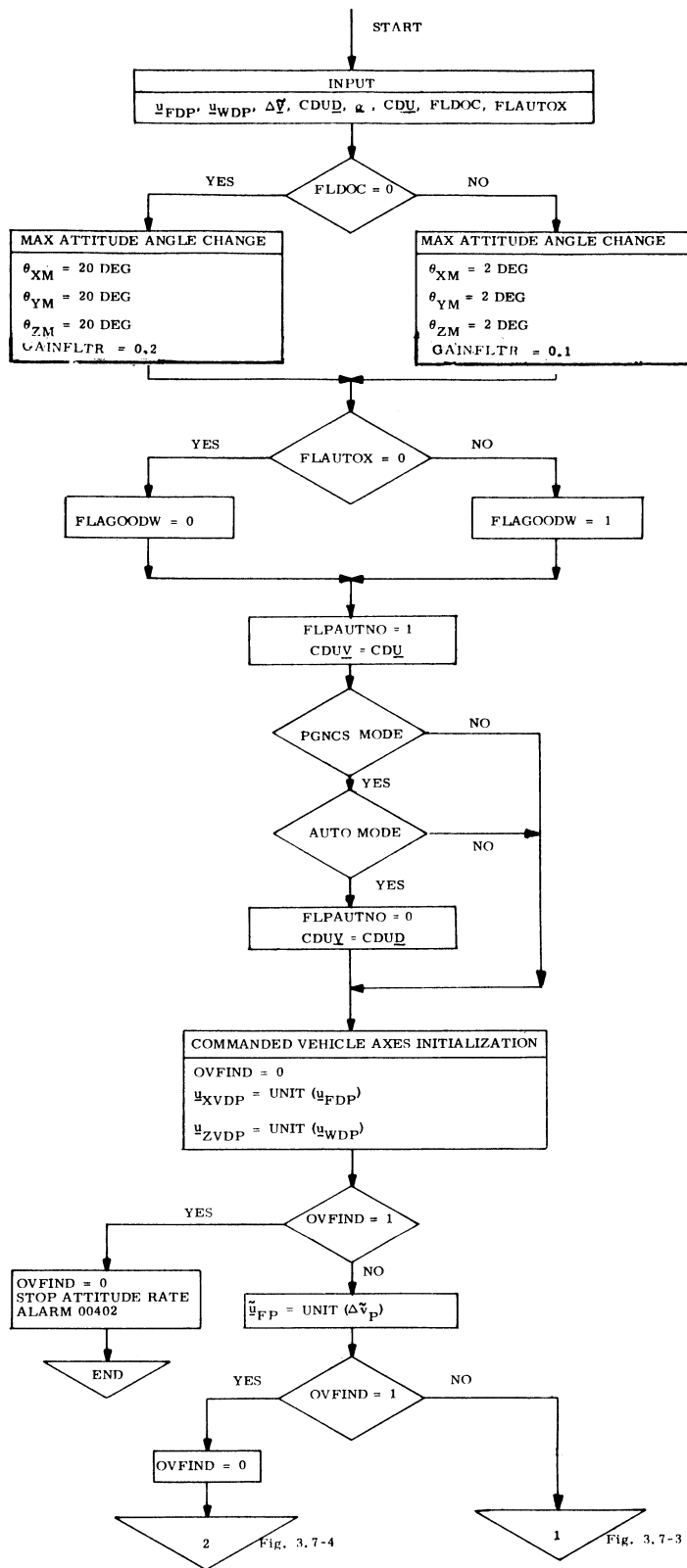


Figure 3.7-2 Each Pass Initialization

5. \underline{u}_{XVDP} and \underline{u}_{ZVDP} , the X and Z axes of the commanded vehicle frame, are initialized to the unit of the thrust command \underline{u}_{FDP} and to the unit of the window command \underline{u}_{WDP} . If unitizing either of these input commands fails either because a vector is too short or too long, the inputs are presumed unsatisfactory for attitude control, alarm 00402 is issued, attitude motion is stopped, and FINDCDUW returns to the calling program.
6. $\Delta \tilde{\underline{v}}_P$, the velocity increment vector measured by the accelerometers over the previous computation period, is unitized for the thrust direction filter. If this produces overflow, the thrust direction filter is bypassed.

5.3.7.3.3 Thrust Direction Filter

As the first step in the Thrust Direction Filter, Fig. 3.7-3 the unit measured thrust in platform coordinates is transformed to current vehicle coordinates to yield the unit measured thrust $\tilde{\underline{u}}_{FV}$. Then each of the two components, Y and Z, of the old unit thrust vector is subtracted from the corresponding component of the unit measured thrust, multiplied by the gain of the filter, GAINFLTR, and limited in magnitude to the maximum change which can be expected in one computation period, which yields the thrust direction changes Δu_{FYV} and Δu_{FZV} . The maximum changes are based on the slew rate of the trim gimbal. Each of these thrust direction changes is then added to the corresponding component of the old unit thrust vector and limited in magnitude to the maximum thrust offset which can be expected, yielding the new unit thrust vector \underline{u}_{FV} . The maximum

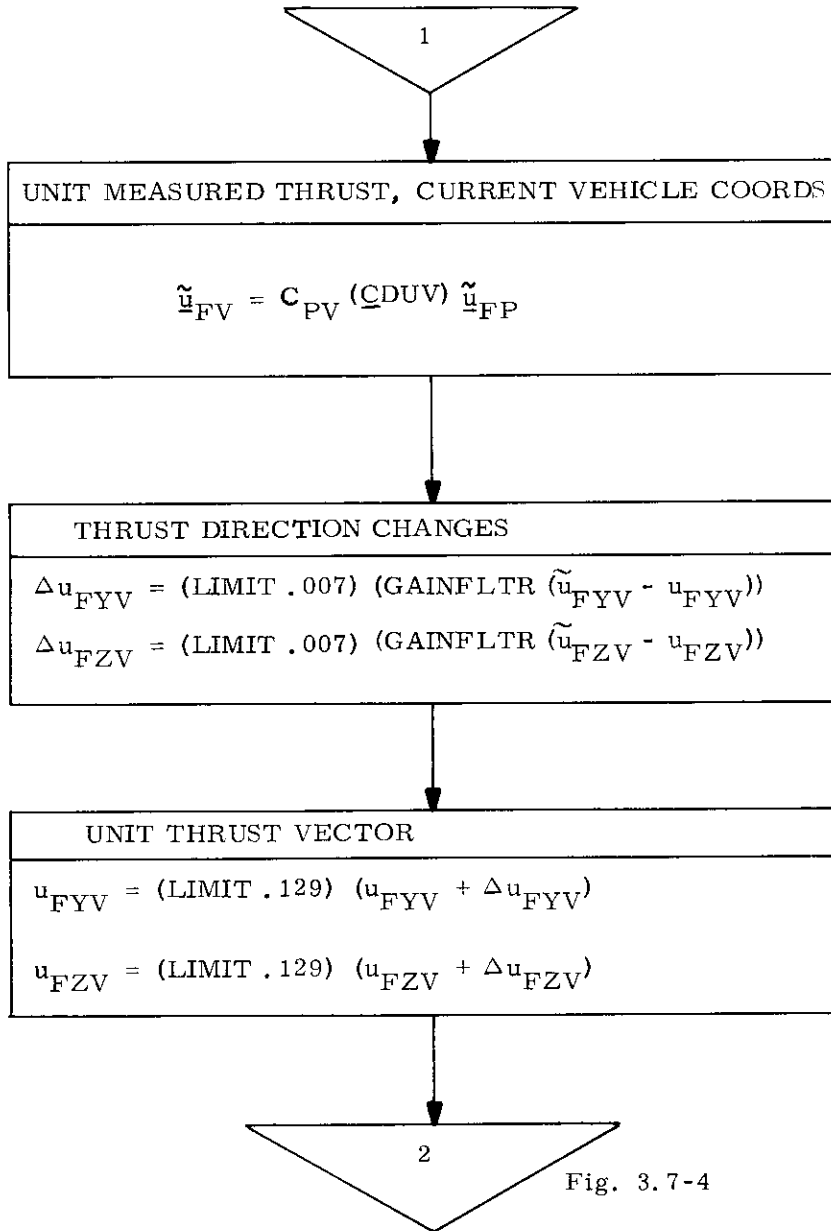


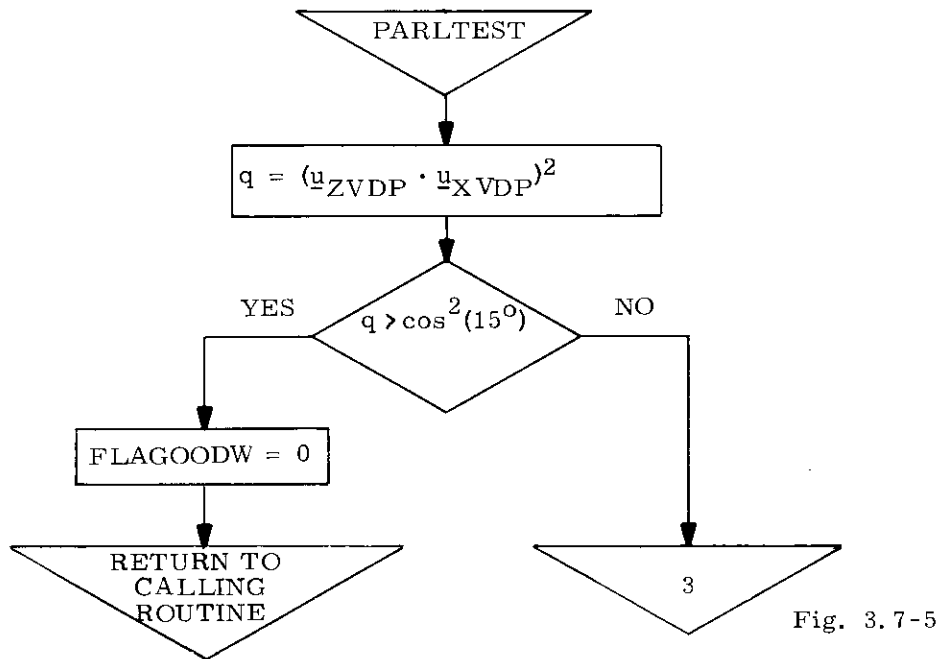
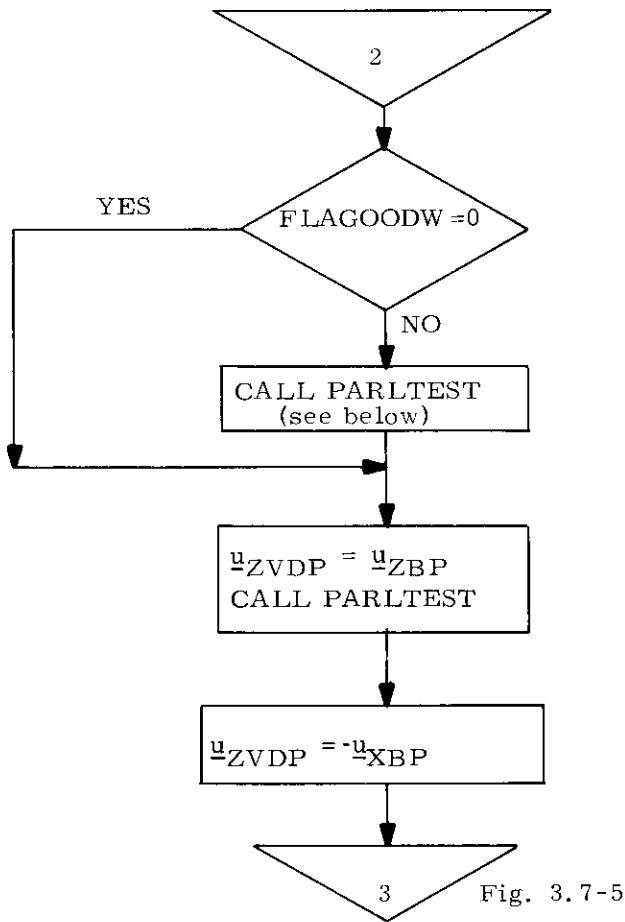
Fig. 3.7-4

Fig. 3.7-3 Thrust Direction Filter.

thrust offset is based on the maximum displacement of the trim gimbal, the elastic deflection of the gimbal mount, the mounting alinement error, and the deflection of the thrust vector relative to the nozzle centerline.

5.3.7.3.4 Window Command Selection

In order to compute the axes of the commanded vehicle frame, it is first necessary to select a suitable unit window command (Fig. 3.7-4). The unit window command obtained from guidance may not be suitable because 1) X axis override may be permitted or 2) it may fail the test PARLTEST (a test for parallelism or anti-parallelism to the unit thrust command). If the unit window command cannot be used for either reason, it is necessary to fetch an alternate such that the resulting axes of the commanded vehicle frame will provide zero steady state thrust pointing error. Any vector lying in the +Z half of the LM symmetry plane will satisfy this requirement. Among these, the vector least likely to be parallel or anti-parallel to the unit thrust command is the Z axis (\underline{u}_{ZBP}). However, it is possible during a large maneuver that the Z axis also fails PARLTEST, and in this case a third alternative, the -X axis ($-\underline{u}_{XBP}$) is selected. Because the Z axis and the -X axis cannot both be parallel to the unit thrust command, PARLTEST is omitted for -X. The Z and -X axes selected are of the body and not the vehicle frame, simply to save program and time, as the Service Routine provides the body axes for all users. Using a body axis for window pointing causes no stability problem because any time a body axis is used, FLAGOODW is set to zero so that the attitude angle change about the X axis will also be set to zero. Thus, actual spacecraft motion about the X axis produces through FINDCDUW no attitude increment command about the X axis and greatly diminished commands about the Y and Z axes.



5.3.7.3.5 Commanded Vehicle Axes

Figure 3.7-5 illustrates the erection of the commanded vehicle axes. The components of the unit thrust vector in vehicle coordinates, u_{FYV} and u_{FZV} , are treated as small angles, and the corresponding approximations are made. The commanded vehicle axes are computed in two steps. First, an orthogonal set is erected with the X axis along the unit thrust command and the Y axis normal to the plane of the unit window command and the unit thrust command. Then the X axis is displaced relative to this initial frame according to the thrust offset. The final Y axis is oriented normal to the initial Z axis and the final X axis, and final Z axis is computed to complete the right hand orthogonal triad. No unit vectors need be taken in this final iteration. The consequence of not unitizing produces an error only in the outer gimbal angle with a negligible maximum value occurring at 45° . This is explained in the next section.

With the coordinate frame erected as described in the preceding paragraph, there is a window pointing error; that is, the unit window command lies out of the ZX plane by approximately the thrust offset angle about the X axis multiplied by the sine of the angle between the Z axis and the window command vector.

5.3.7.3.6 Commanded Gimbal Angles

Commanded gimbal angles (CDUC's, Fig. 3.7-6) are the reference angles to which the CDUD's are driven, and are also the angles placed in the Noun 22 registers for the Mode 2 attitude display.

The commanded gimbal angles are derived from the matrix just computed whose row vectors are the unit axes of the commanded vehicle frame expressed in platform coordinates, and whose elements are c_0 through c_8 . This matrix may be derived in terms of the sines and cosines of the gimbal angles by starting with the matrix whose rows are these unit vectors expressed in commanded vehicle coordinates (the identity matrix) and rotating the row vectors through the negative of the outer, middle, and inner gimbal angles in that order. The resulting matrix may be used to identify the elements of interest of the matrix of the preceding paragraph as follows.

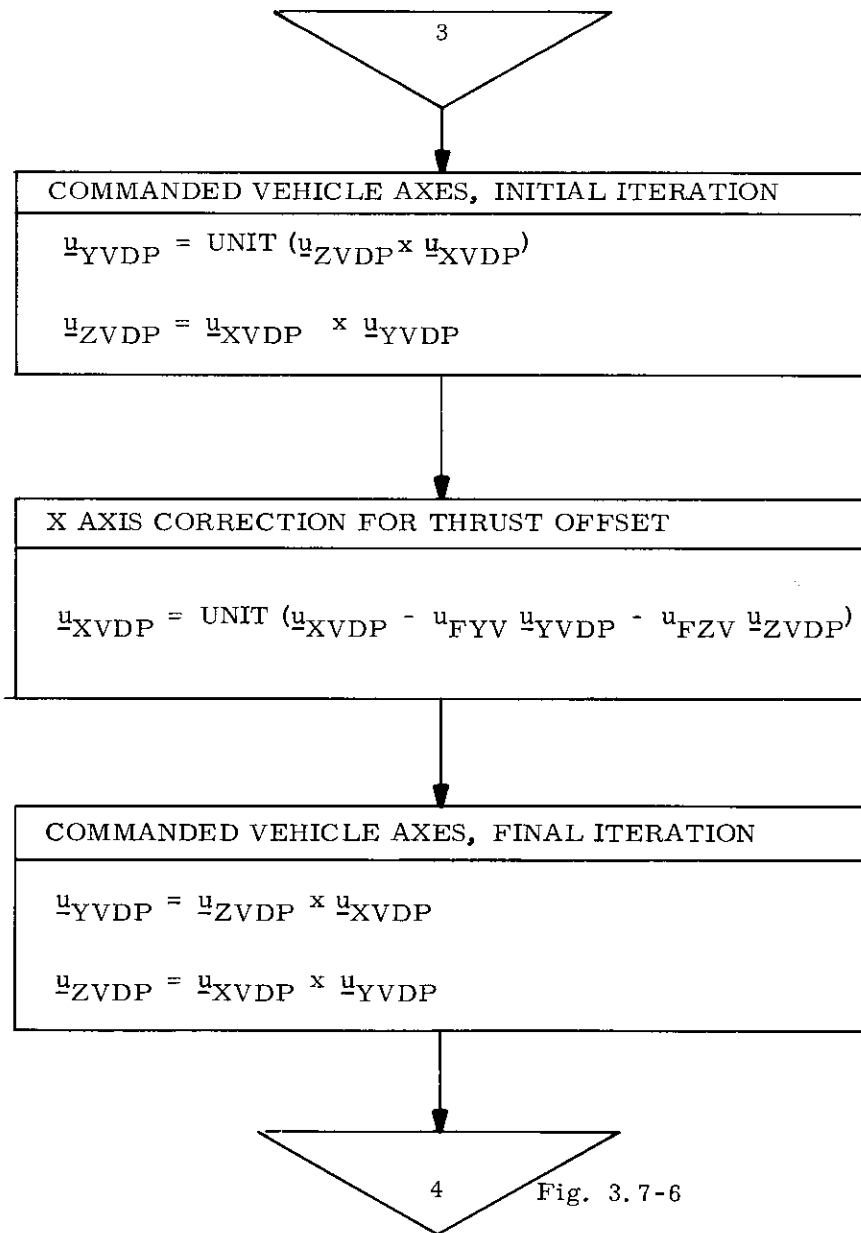
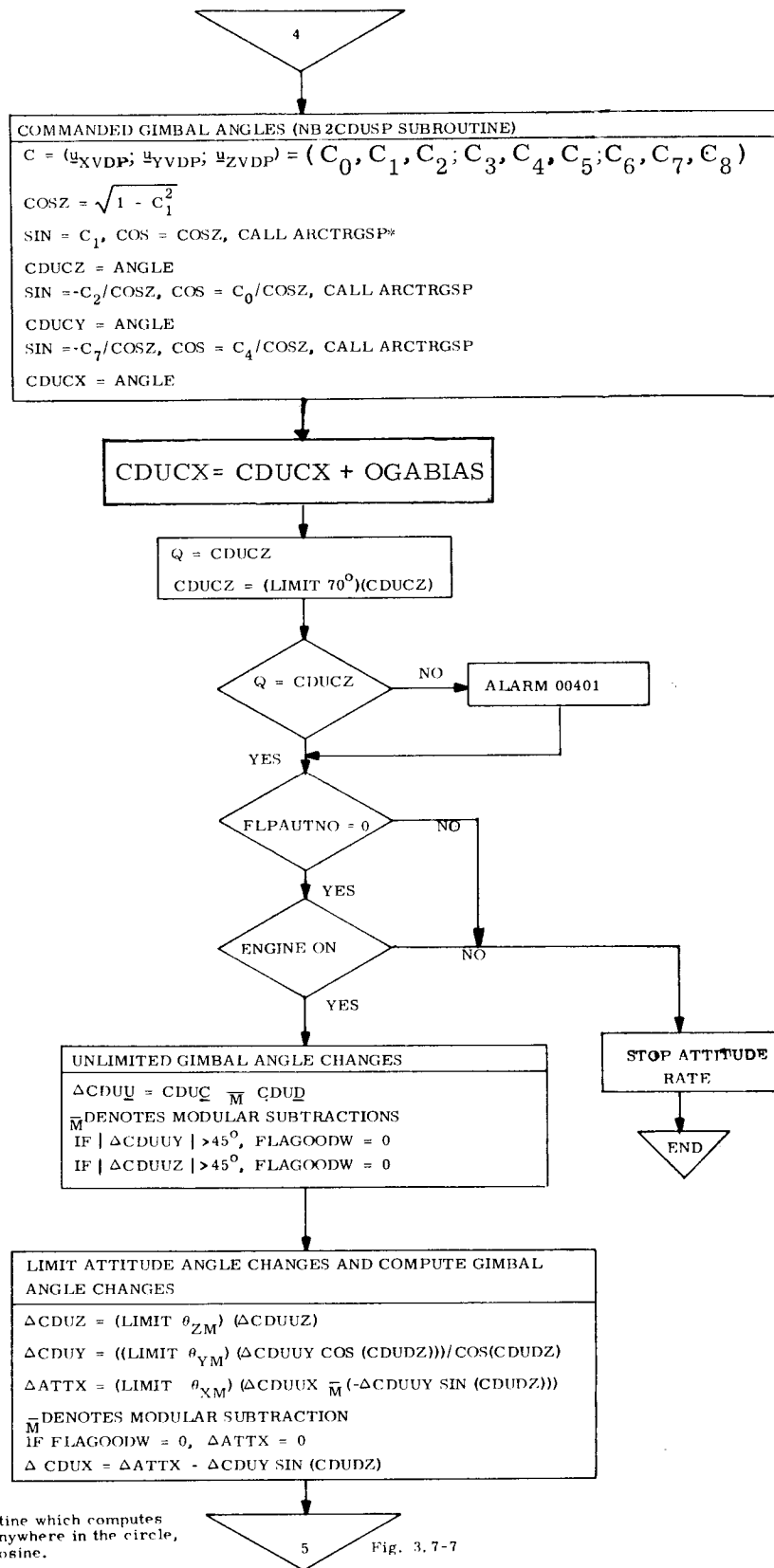


Fig. 3.7-6

Fig. 3.7-5 Commanded Vehicle Axes.



* ARCTRGSP is a subroutine which computes an unambiguous angle anywhere in the circle, given the sine and the cosine.

Fig. 3.7-7

Figure 3.7-6 Commanded Gimbal Angles and Gimbal Angle Changes

5.3-170

$$C_0 = \cos (CDUCZ) \cos (CDUCY)$$

$$C_1 = \sin (CDUCZ)$$

$$C_2 = -\cos (CDUCZ) \sin (CDUCY)$$

$$C_4 = \cos (CDUCZ) \cos (CDUCX)$$

$$C_7 = -\cos (CDUCZ) \sin (CDUCX)$$

The subroutine NB2CDUSP extracts the commanded gimbal angles, CDUC, by exploiting these relationships. $|CDUCZ|$ is $< 90^\circ$, even before limiting.

ARCTRGSP is a subroutine of the subroutine NB2CDUSP. ARCTRGSP extracts angles given the sine and the cosine. To this end, ARCTRGSP extracts the arcsine of whichever given argument is smaller in magnitude, and uses only the sign of the larger argument for determining the quadrant.

Not unitizing the vectors in the final iteration for the commanded vehicle axes, as noted in the preceding section, reduces slightly the magnitude of the arguments used to extract the outer gimbal angle. Because ARCTRGSP extracts the arcsine of whichever argument is smaller in magnitude, a "unit" vector of length .992 would produce a maximum outer gimbal angle error (at 45°) of $.458^\circ$. A more probable length is .99966 which would produce a maximum error of only $.0197^\circ$.

If flag FLPAUTNO is set or if the engine is off, a STORATE is issued and control is returned to the caller of FINDCDUW.

The unlimited gimbal angle changes are computed by modular subtracting the autopilot's CDUD's from the CDUC's. The modular subtractions yield the smallest angular differences, e. g. $177^\circ - (-178^\circ) = -5^\circ$, not $+355^\circ$. If either the Y axis or the Z axis unlimited gimbal angle change exceeds 45° in magnitude, FLAGOODW is set to zero. This is to prevent false aborts in attitude about the X axis as will be explained in the next paragraph.

Attitude angle changes, Δ ATT, are handled in a vehicle frame not previously described. The X axis of the frame is identical to the X axis of the current frame, the Z axis is along the middle gimbal axis, and the Y axis completes the orthogonal

right hand triad. Thus, this frame is the current vehicle frame rotated about the X axis to bring the Z axis into alignment with the middle gimbal axis. This frame is chosen for computational simplicity. The Z axis gimbal angle change $\Delta CDUZ$ is identical to the Z axis attitude angle change $\Delta ATTZ$. The unlimited Y axis gimbal angle change is multiplied by $\cos(\Delta CDUZ)$ to yield the unlimited Y axis attitude angle change, this is limited and divided by $\cos(\Delta CDUZ)$ to yield the limited Y axis gimbal angle change $\Delta CDUY$. The unlimited X axis attitude angle change is computed using the unlimited X and Y axes gimbal angle changes. Note that the modular subtraction always produces a result under 180° in magnitude. This computation produces the smallest angle total rotation required about the X axis, although the validity diminishes as the gimbal angle changes become large. Under certain conditions the X axis attitude angle change computed will be of opposite sign compared to the succeeding pass. To prevent starting such a maneuver in the wrong direction about the X axis, $\Delta ATTX$ is set to zero whenever either the Y axis or the Z axis unlimited gimbal angle change exceeds a limit value. Finally the limited X axis gimbal angle change, $\Delta CDUX$, is computed using the limited attitude change, $\Delta ATTX$, and the limited gimbal angle change $\Delta CDUY$.

Because orthogonal components of the attitude rate are limited independently, the length of the attitude rate vector is allowed to exceed the limit value for any one component.

5.3.7.3.7 Autopilot Data

With all limiting completed, it is now possible to compute a consistent set of input data for the digital autopilot using the ΔCDU 's (Fig. 3.7-7). The ΔCDU 's are divided by the computation period and the resulting gimbal angle rates are transformed to yield the commanded attitude rates ω_{DV} . The gimbal angle changes are multiplied by the ratio of the control sample period to the computation period to yield the gimbal angle increments δCDU . Finally, the commanded attitude lag angles ϕ , which account for the angles by which the attitude will lag behind a ramp angular command due to the finite angular accelerations available, are computed using the available angular accelerations α (the one jet accelerations supplied by the autopilot, doubled) and then individually magnitude limited.

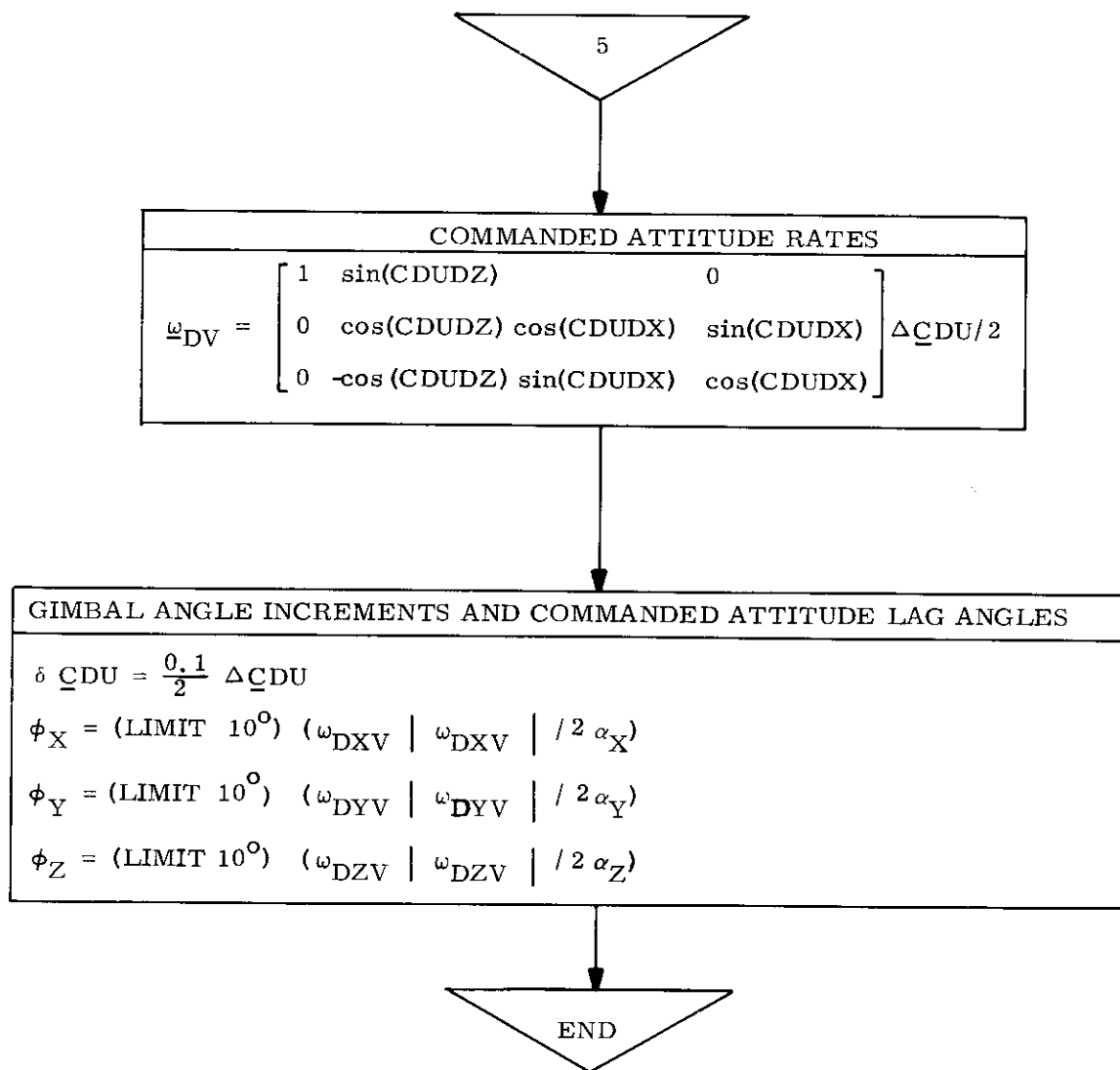


Fig. 3.7-7 Autopilot Data.

5.3.8 MIDTOAVE ROUTINE

The purpose of the MIDTOAVE Routine (R41) is to insure that the extrapolated state used to initialize the Average-G computations is consistent with the time at which the Average-G equations are actually turned on. The information flow diagram for this routine is presented in Fig. 3.8-1.

Before this routine is entered the ignition-time flag (FLTIG) is set. If the calling program is the Thrust Monitor Program (P47), FLTIG has been set to 1; otherwise FLTIG = 0. The time at which the Average-G equations are to be turned on (t_{AG}) is normally set to 29.9 seconds prior to ignition time (t_{IG}). If P47 is the calling program, t_{IG} is not defined and the Average-G equations will be turned on as soon as possible. In the powered landing maneuver t_{IG} is obtained from the Guidance and Control Routine.

Initially, a state vector (\underline{r}' , \underline{v}') and its corresponding time (t') are available to the precision Coasting Integration Routine. This time will normally be several minutes before the computed ignition time (t_{IG}). The first step in the computation cycle is to check the ignition-time flag (FLTIG). A value FLTIG = 0 indicates that the ignition time has not as yet been slipped. If FLTIG = 0, a test is made to determine whether there are at least 20 seconds remaining before the Average-G turn-on time (t_{AG}). If there are less than 20 seconds remaining, the ignition-time flag (FLTIG) is set to unity and t_{AG} is advanced to 20 seconds beyond the present time (t). In addition, an alarm is sent to the crew to signal that the ignition time will be slipped. Note that the crew alarm is sent only once, since FLTIG = 1 on subsequent passes through the routine.

5.3-174

Revised LUMINARY
 Added GSOP # 567 PCN# 758* Rev. 7 Date 11/69

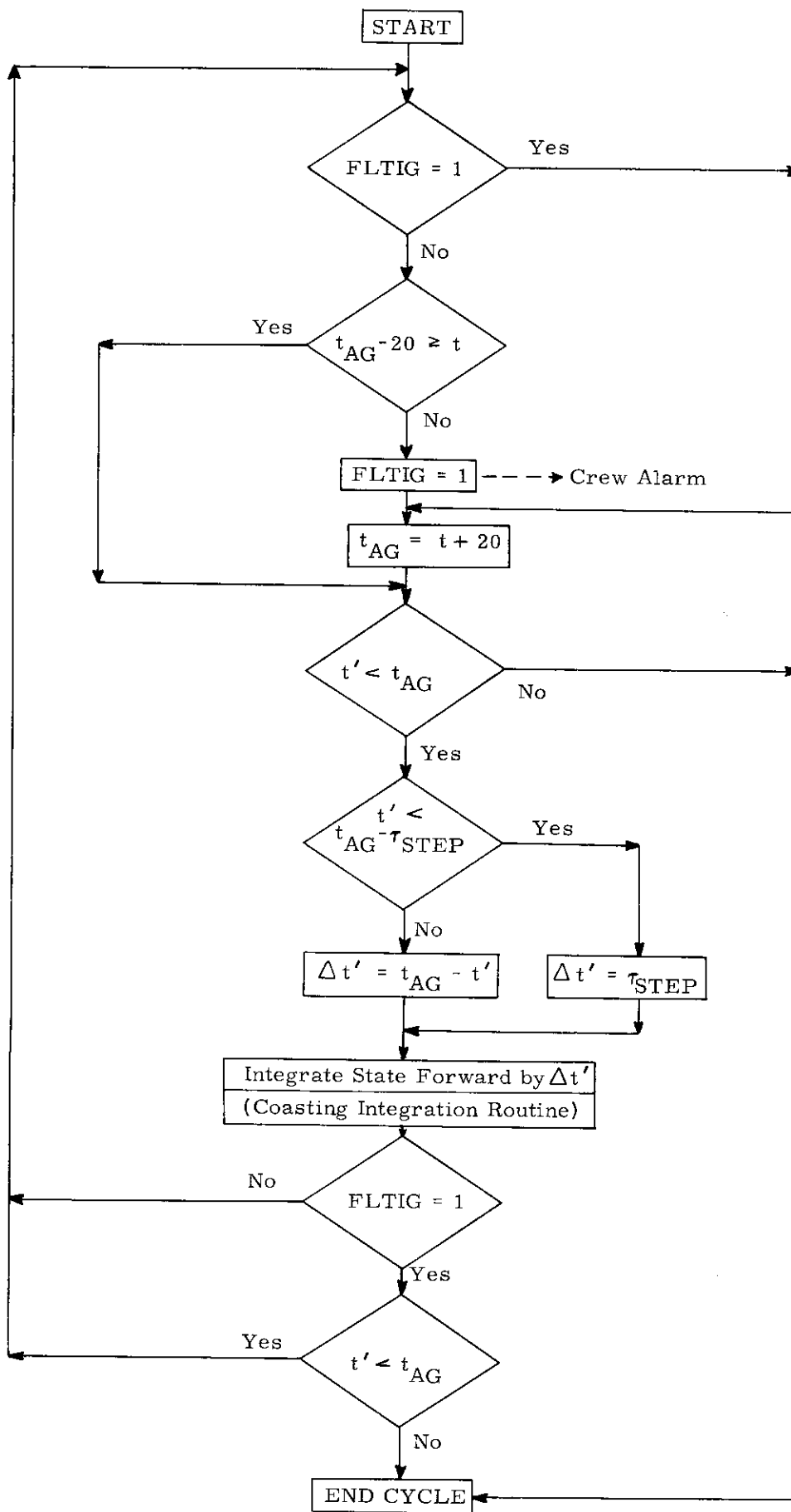


Figure 3. 8-1 MIDTOAVE Routine
5. 3-175

If integration is not finished from a previous cycle, the next step is to check the time corresponding to the computed state-vector (t') to see if it is within one time step (τ_{STEP})[†] of the Average-G turn-on time. If t' is within one time step of the turn-on time (t_{AG}), then the time step used in the Coasting Integration Routine ($\Delta t'$) is decreased from the initial value (τ_{STEP}) to the value $t_{\text{AG}} - t'$. The state is then extrapolated forward by $\Delta t'$ in the Coasting Integration Routine.

A test is made to determine if the state-vector has been integrated all the way to the desired Average-G time (t_{AG}). If the time corresponding to the extrapolated state-vector (t'), as determined by the Coasting Integration Routine, is within .03 seconds of t_{AG} the integration is complete; otherwise integration proceeds as shown in the figure.

This completes the computation cycle through the MIDTOAVE Routine. It should be noted that the ignition time (t_{IG}) will be computed, if required, by the calling program. Normally, t_{IG} will be set to 30 seconds past the finally-selected Average-G turn-on time.

[†]The nominal integration time step (τ_{STEP}) used in the Coasting Integration Routine is presently 350 seconds for a lunar orbit and requires 2.5 seconds of LGC time.