# Appendix E

# GUIDANCE EQUATIONS

# Appendix E

# GUIDANCE EQUATIONS

## I. INTRODUCTION

This appendix describes the operational IGS guidance equations.

In general, these equations have been derived in Appendix A and in references 1 through 7 listed in Appendix H.

Certain portions of the guidance equations are presented which have specific application to the GEMINI IGS and have not been derived in the referenced documents. A discussion of the IGS equations as implemented in the simulation has been included in this appendix to present the simulation operation in more detail.

## II. DESCRIPTION OF IGS ASCENT GUIDANCE OPERATION

This section defines the operation of the Ascent Guidance equations during launch. Numbers in parentheses refer to areas or blocks in the Ascent Math Flow Diagrams (Figures E-1A and E-1B). See Table E-I for the definitions of symbols. The platform is aligned as described in Appendix A. The word "continuously" when used in the discussion of computer operations means that the information is updated at approximately 0.5-sec. intervals.

### A. FROM START TO PLATFORM RELEASE

The computer will first be turned to the Ascent mode. All quantities which require initialization are contained in block A102. Agena ephemeris data will be inserted into the computer approximately 60 sec. prior to launch. This information will be continuously updated (A112) and will be used to define the azimuth orientation of the platform X-axis with respect to East ($\gamma$) and the angle between the orbit plane and the launch site ($\delta$). The initial conditions on the spacecraft velocity along platform axis (A117) and the initial platform roll gimbal angle ($\phi_N$ - A116) will also be computed continuously.

At approximately 30 sec. prior to launch, the platform will be released from torquing.* The guidance system is now in the inertial mode.

---

*As of 28 January 1963, this time — 30 sec. prior to launch — is not definite and may eventually be "time of engine ignition."

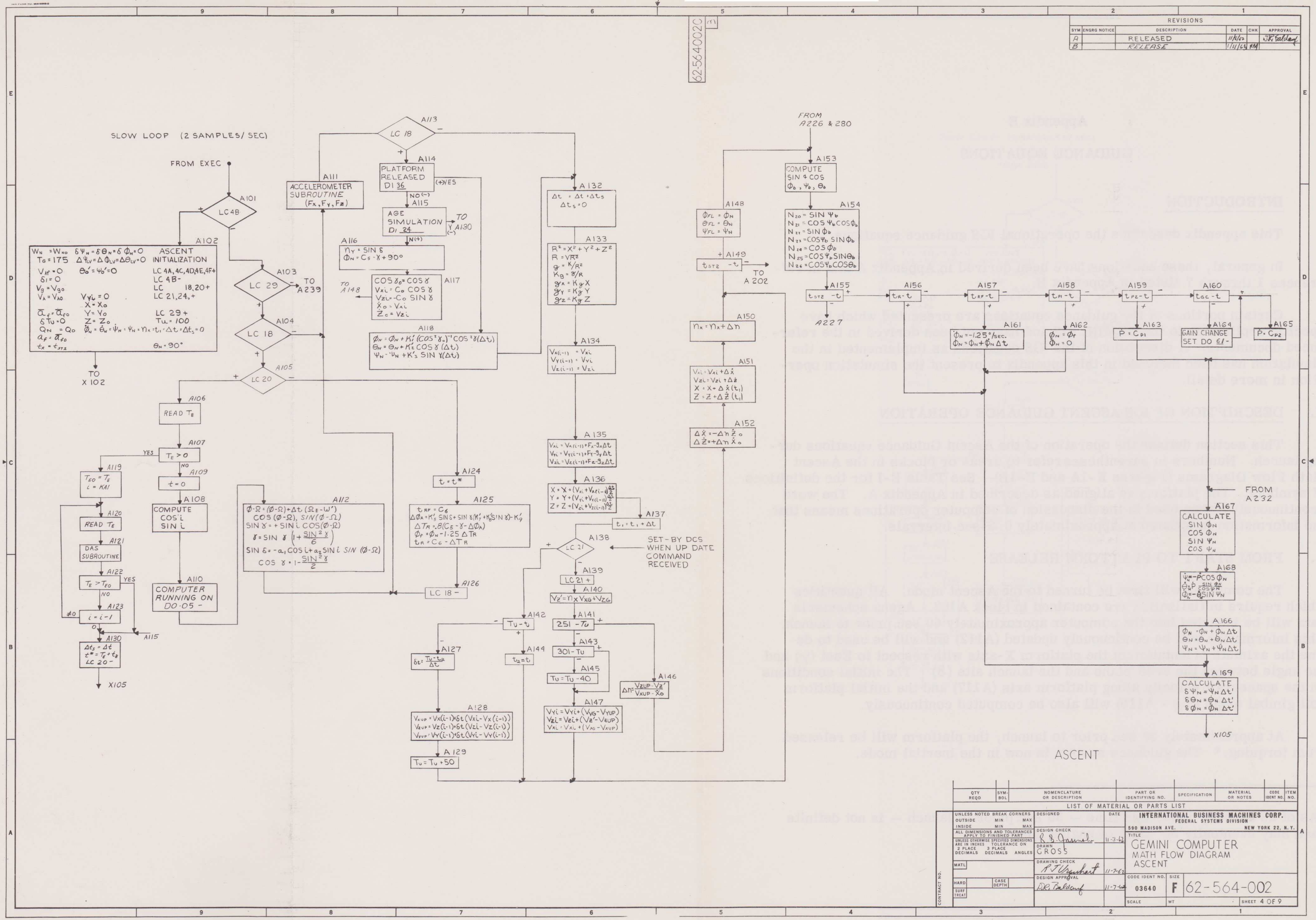
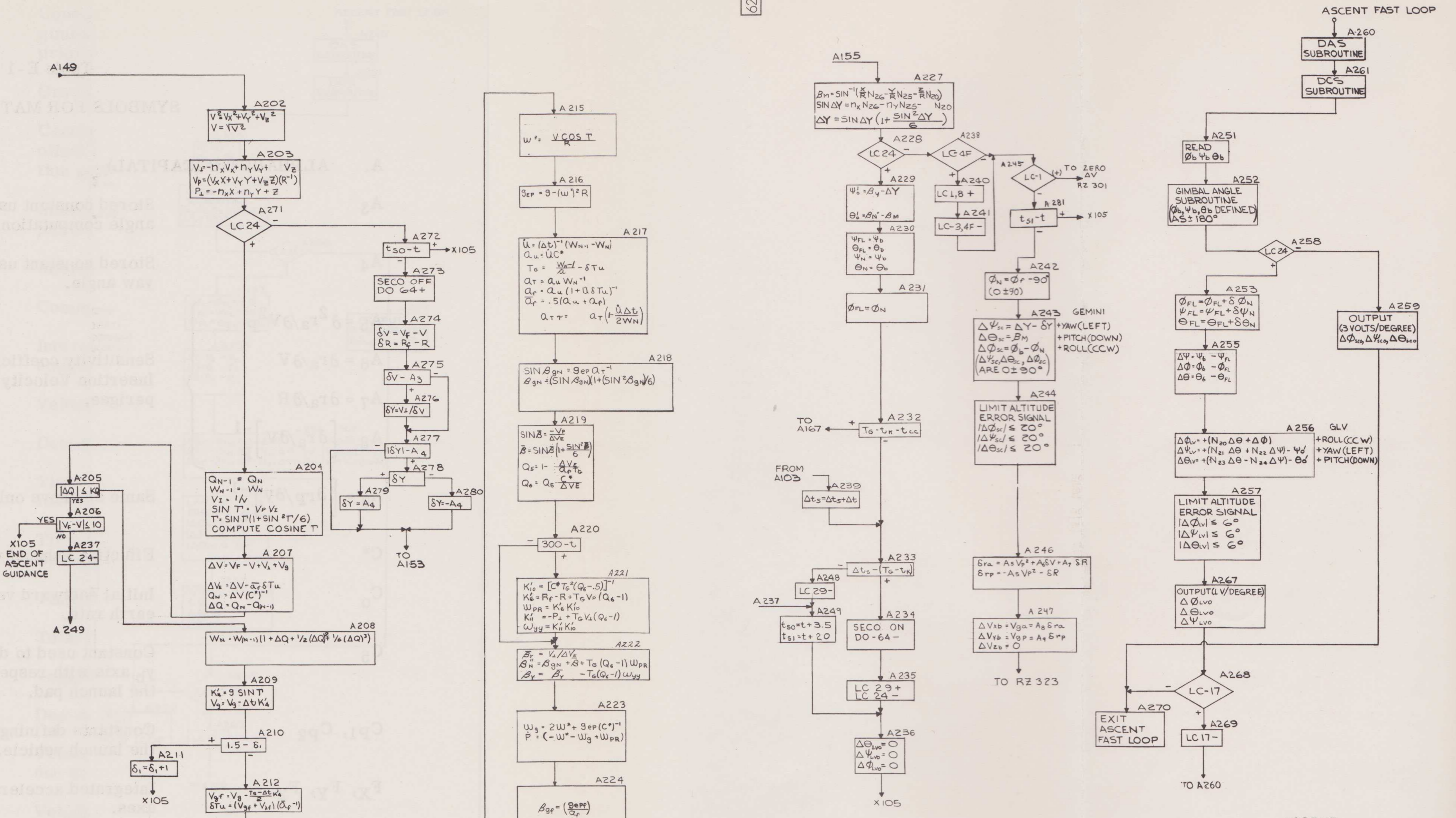


Figure E-1A. GEMINI Computer Math Flow Diagram Ascent

62-5640020 B

| REVISIONS | | | | | |
|---|---|---|---|---|---|
| SYM | ENGRG NOTICE | DESCRIPTION | DATE | CHK | APPROVAL |
| A | | RELEASE | 11/8/62 | | D.E. Baldauf |
| B | | RELEASE | 1/11/63 | RR | |



ASCENT

| QTY REQD | SYMBOL | NOMENCLATURE OR DESCRIPTION | PART OR IDENTIFYING NO. | SPECIFICATION | MATERIAL OR NOTES | CODE IDENT NO. | ITEM NO. |
|---|---|---|---|---|---|---|---|

LIST OF MATERIAL OR PARTS LIST

CONTRACT NO.

UNLESS NOTED BREAK CORNERS
OUTSIDE MIN MAX
INSIDE MIN MAX
ALL DIMENSIONS AND TOLERANCES APPLY TO FINISHED PART
UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES TOLERANCE ON
2 PLACE DECIMALS 3 PLACE DECIMALS ANGLES
MATL
HARD CASE DEPTH
SURF TREAT

| | | DATE |
|---|---|---|
| DESIGNED | | |
| DESIGN CHECK | R.B. Jasinski | 11-3-62 |
| DRAWN | | |
| DRAWING CHECK | R J Urquhart | 11-7-62 |
| DESIGN APPROVAL | D.E. Baldauf | 11-7-62 |

**INTERNATIONAL BUSINESS MACHINES CORP.**
FEDERAL SYSTEMS DIVISION
590 MADISON AVE. NEW YORK 22, N.Y.

TITLE
GEMINI COMPUTER
MATH FLOW DIAGRAM
ASCENT

CODE IDENT NO. 03640
SIZE F
62-564-0020
SCALE
WT
SHEET 5 OF 9

## Table E-1

## SYMBOLS FOR MATH FLOW

### A. ALPHABETIC (CAPITAL)

| | |
|---|---|
| $A_3$ | Stored constant used to inhibit spacecraft yaw angle computation. |
| $A_4$ | Stored constant used to limit spacecraft allowable yaw angle. |
| $A_5 = \partial^2 r_a/\partial V^2_P$ | Sensitivity coefficients used for Spacecraft Insertion Velocity Adjustment Evaluated at perigee. |
| $A_6 = \partial r_a/\partial V$ | |
| $A_7 = \partial r_a/\partial R$ | |
| $A_8 = \left[\partial r_a/\partial V\right]^{-1}_P$ | |
| $A_9 = \left[\partial r_P/\partial V\right]^{-1}_a$ | Same as above only evaluated at apogee. |
| $C^*$ | Effective exhaust velocity. |
| $C_o$ | Initial eastward velocity at launch point due to earth rate. |
| $C_5$ | Constant used to define orientation of spacecraft $y_b$ axis with respect to east, while vehicle is on the launch pad. |
| $C_{P1}$, $C_{P2}$ | Constants defining step 1 and step 2 pitch rate of the launch vehicle. |
| $F_X$, $F_Y$, $F_Z$ | Integrated acceleration components along platform axes. |

## Table E-1. Symbols For Math Flow

| | |
|---|---|
| $K_1'$, $K_2'$, $K_3'$ | Constants used in update of commanded platform gimbal angles after the platform is released but prior to liftoff. |
| $K_4'$ | Gravity component along vehicle velocity vector. |
| $K_5'$, $K_7'$, $K_8'$, $K_9'$ | Constants used to compute Stage 1 azimuth angle offset to compensate for initial velocity and position perpendicular to the orbit plane. |
| $K_6'$ | Intermediate computer quantity used in calculating pitch rate in steering. |
| $K_g$ | Intermediate quantity used in gravity computation. |
| $\dot{P}$ | Commanded vehicle pitch rate. |
| $Q_5$, $Q_6$, $Q_N$ | Intermediate coefficients used in steering equations. |
| $R$ | Vehicle distance from earth center. |
| $R_F$ | Desired insertion altitude. |
| $T_E$ | Time read from TRS. Will remain zero until lift-off. |
| $T_G$ | Time to go. Used in Stage 2 steering equation. Defines time to effective thrust termination. |
| $T_u$ | Time of update. |
| $V$ | Total vehicle velocity. |
| $V_\perp$ | Vehicle velocity perpendicular to the orbit plane. |
| $V_f$ | Desired insertion velocity. |
| $V_g$ | Velocity loss term approximating expected loss due to gravity. |
| $V_P$ | Velocity along the vehicle position vector (Radial velocity). |

Table E-1. Symbols For Math Flow (cont)

| | |
|---|---|
| $V_{\lambda}$ | Velocity loss term approximating expected loss due to steering (angle of attack). |
| $V_{gf}$ | Final value of gravitational velocity loss term remaining at shutdown. |
| $V_{\lambda f}$ | Final value of steering velocity loss term remaining at shutdown. |
| $V_{XG}$, $V_{YG}$, $V_{ZG}$ | Velocity update components transmitted from ground. |
| $V_{Xi}$, $V_{Yi}$, $V_{Zi}$ | Measured platform velocity components. |
| $V_{XUP}$, $V_{YUP}$, $V_{ZUP}$ | Platform velocity components interpolated to update time. |
| $V_{ga}$ | Horizontal velocity increment required at perigee to reach apogee. |
| $V_{gP}$ | Horizontal velocity increment required at apogee to reach perigee. |
| $V'_{Z}$ | Ground velocity update components corrected for azimuth orientation of platform. |
| $W_N$ | Intermediate computer quantity used in vehicle kinematic computations ($W = e^{\Delta V/C^*}$). |
| X, Y, Z | Vehicle position components. |
| $\dot{X}_o$, $\dot{Z}_o$ | Eastward velocity of earth resolved into platform frame. |

## B. ALPHABETIC (SMALL)

| | |
|---|---|
| $a_1$ | Sine $\lambda$, where $\lambda$ is latitude of launch point. |
| $a_2$ | Cosine $\lambda$, where $\lambda$ is latitude of launch point. |
| $a_f$ | Final value of thrust acceleration. |
| $a_T$ | Thrust acceleration now. |

Table E-1. Symbols For Math Flow (cont)

| | |
|---|---|
| $a_u$ | Value of thrust acceleration at nominal shutdown time when W = 1. |
| $\bar{a}_f$ | Average value of thrust acceleration between nominal and actual shutdown time. |
| $a_{T\tau}$ | Average thrust acceleration over previous computation cycle. |
| $g$ | Gravity at vehicle altitude. |
| $g_{ep}$ | Effective gravity - gravity minus centripetal acceleration - along geocentric vertical. |
| $g_{epf}$ | Effective gravity at nominal shutdown condition. |
| $g_X$, $g_Y$, $g_Z$ | Gravity components resolved along platform axis. |
| $\dot{i}$ | Inclination of Agena orbit plane. |
| $t$ | Time after lift-off |
| $t_B$ | Constant - time bias used to correct for delays in receipt and detection of lift-off signal. |
| $t_{cc}$ | Constant approximately equivalent to the duration of computer Stage 2 slow loop. Used in connection with the initiation of SECO countdown. |
| $t_{GC}$ | Time of autopilot gain change following lift-off. |
| $t_K$ | Constant used to compensate for any thrust acceleration imparted to the vehicle following the issuance of the shutdown discrete (mostly cut-off impulse). |
| $t_R$ | Time to start the roll program. |
| $t_{RF}$ | Time to stop the roll program. |
| $t_{P1}$ | Time to begin the step 1 pitch program. |
| $t_{P2}$ | Time to begin the step 2 pitch program. |

Table E-1. Symbols For Math Flow (cont)

| | |
|---|---|
| $t_{SO}$ | Computer quantity used to delay entry into insertion velocity adjust program and to delay turn-off of the SECO discrete. |
| $t_{ST2}$ | Time to start Stage 2 guidance. |
| $\dot{u}$ | Intermediate quantity used in vehicle kinematic computations. |

C. GREEK (CAPITAL)

| | |
|---|---|
| $\bar{\beta}$ | Thrust attitude required to compensate for vehicle radial velocity. |
| $\beta''$ | Final value of commanded thrust attitude. |
| $\beta_M$ | Computed value of vehicle pitch attitude. Actual vehicle pitch attitude with respect to local horizontal. |
| $\beta_Y$ | Commanded vehicle yaw angle (with respect to orbit plane). Includes explicit yaw steering. |
| $\bar{\beta}_Y$ | Vehicle yaw angle (with respect to orbit plane) required to kill velocity perpendicular to the orbit plane. |
| $\beta_{gN}$ | Thrust attitude required to compensate for effective gravity. |
| $\beta_{gf}$ | Final value of thrust attitude required for gravity compensation. |
| $\beta''_N$ | Commanded vehicle thrust attitude with respect to local horizontal. |
| $\Gamma$ | Vehicle flight path angle with respect to local horizontal. |
| $\Delta t$ | Length of slow loop computation cycle. |
| $\Delta t_S$ | Accumulated time following entry into "SECO countdown" loops. Also used to correct initial computation cycle time following detection of lift-off. |

Table E-1. Symbols For Math Flow (cont)

| | |
|---|---|
| $\Delta T_R$ | Time vehicle will roll at a constant rate to reach the proper azimuth orientation. |
| $\Delta t'$ | Fast-loop cycle time (50 msec). |
| $\Delta V$ | Total velocity to be gained prior to nominal shutdown including approximated velocity loss due to gravity and steering. |
| $\Delta V_E$ | Velocity to be gained corrected for actual shutdown time $( V_e = \int_t^{\text{cutoff}} a_T)$ |
| $\Delta Y$ | Computed value of vehicle yaw attitude. Angle between vehicle X-axis and orbit plane. |
| $\theta, \phi, \psi$ | Refer to pitch, roll, and yaw, respectively. |
| $\Delta\theta_{LV}, \Delta\phi_{LV}, \Delta\psi_{LV}$ | Computed vehicle attitude errors. |
| $\Delta\theta_{LVO}, \Delta\phi_{LVO},$<br>$\Delta\psi_{LVO}$ | Limited vehicle attitude errors delivered to autopilot. |
| $\Delta\theta_{SC}, \Delta\phi_{SC}, \Delta\psi_{SC}$ | Computed spacecraft attitude errors during insertion velocity adjust. |
| $\Delta\theta_{SCO}, \Delta\phi_{SCO},$<br>$\Delta\psi_{SCO}$ | Limited vehicle attitude errors displayed to the astronaut during insertion velocity adjust. |
| $\Delta\phi_A$ | Vehicle roll offset required to compensate for vehicle position and velocity perpendicular to the orbit plane. |
| $\theta'_o$ | Vehicle pitch attitude error quantity. |
| $\theta_b, \phi_b, \psi_b$ | Measured gimbal angles. |
| $\theta_N, \phi_N, \psi_N$ | Commanded platform gimbal angles during Stage 1. During Stage 2, $\theta_N$ and $\psi_N$ are equated to actual platform gimbal angles once per slow loop. |

Table E-1. Symbols For Math Flow (cont)

| | |
|---|---|
| $\theta_{FL}, \phi_{FL}, \psi_{FL}$ | Fast-loop commanded gimbal angles. This includes the effects of the pitch rate ($\dot{P}$) term. |
| $\dot{\theta}_N, \dot{\phi}_N, \dot{\psi}_N$ | Commanded gimbal rates. |
| $\lambda$ | Coefficient used in computation of steering loss. |
| $\bar{\lambda}$ | Average value of steering loss coefficient between now and shutdown. |
| $\lambda_f$ | Final value of steering loss coefficient at time of shutdown. |
| $\phi$ | Longitude of vehicle with respect to Greenwich. |
| $\psi_o'$ | Vehicle yaw attitude error quantity. |
| $\Omega$ | Longitude of ascending node of Agena orbit. |
| $\Omega_E$ | Rate of earth rotation. |

D. GREEK (SMALL)

| | |
|---|---|
| $\gamma$ | Prior to platform release, angle between east and the platform X-axis. The platform is torqued so that its X-axis is parallel to the orbit plane. $\gamma$ is positive when X is displaced north. |
| $\gamma_o$ | Value of $\gamma$ at the time of platform release. |
| $\delta$ | Represents the angle between the launch site and the orbit plane; positive when vehicle is below orbit plane. |
| $\delta_Y$ | Spacecraft yaw angle required to kill velocity perpendicular to the orbit plane. |
| $\delta_1$ | Quantity used to allow convergence of certain Stage 1 computations upon initiation of Stage 2 guidance. |
| $\delta_R$ | Position increment above or below nominal insertion altitude. |

Table E-1. Symbols For Math Flow (cont)

| | |
|---|---|
| $\delta t$ | Computed quantity which is used in the linear interpolation of velocity data. Velocity data is corrected to update time. |
| $\delta T_u$ | Computed quantity representing the adjustment to nominal Stage 1 engine shutdown time. |
| $\delta V$ | Velocity increment above or below nominal insertion velocity. |
| $\delta r_a$ | Total computed position increment above or below apogee. |
| $\delta r_p$ | Total computed position increment above or below perigee. |
| $\delta\theta_N, \delta\phi_N, \delta\psi_N$ | Fast-loop gimbal angle increments used to produce desired pitch rate. |
| $\eta_X, \eta_Y, \eta_Z$ | Matrix coefficients used to obtain platform components perpendicular to the orbit plane. |
| $\omega'$ | Nodal precession rate of the orbit plane. |
| $\omega^*$ | Pitch rate term used to keep thrust attitude constant with respect to local vertical. |
| $\omega Pr$ | Pitch rate term used to satisfy vehicle altitude constraint. |
| $\omega_g$ | Pitch rate term used to compensate for apparent rotation of gravity vector. |

## B. PLATFORM RELEASE TO LAUNCH

Following platform release, the computer will commence with the navigation function (A133, A134, A135 and A136). In addition, the computer will update platform gimbal angles, which are changing due to earth rotation prior to launch (A118). The launch vehicle attitude errors (A256) will be computed during this time as well as prior to platform release. These quantities, if monitored, could serve to provide some information on the operational readiness of the IGS system.

In block A106 the computer will be continuously reading the output of the spacecraft time reference system ($T_E$). Any change in this value from zero will indicate that lift-off has occurred. The computer will then go into a fast-loop (A120, A121 and A122) to obtain the time of lift-off within approximately 10 msec.

## C. STAGE 1 – OPEN-LOOP STEERING

The magnitude of the roll maneuver (including offset for vehicle position and velocity perpendicular to the orbit plane) is computed (A125) following lift-off. This angle in combination with $t_{RF}$ will be used to compute the time to start a constant rate (1.25 deg./sec.) roll program ($t_R$), which will bring the vehicle to the required azimuth.

The vehicle will rise vertically and the computer will test (approximately every 0.5 sec.) for time to start the roll program (A156). At the proper time, gimbal angles and rates will be defined in block A161 and attitude errors for the launch vehicle will be generated in block A256. Following the completion of the constant roll program, the commanded roll gimbal angle ($\phi_N$) will be set equal to the value computed for the final roll gimbal angle ($\phi_f$ - A162).

The start of the first- and second-step pitch maneuvers will be controlled by blocks A158 and A159. The time to provide the output for the gain change discrete will be controlled by A164. The pitch profile produced is such as to approximate a gravity turn, thus minimizing the angle of attack and, therefore, the normal forces on the vehicle. During Stage 1 as well as Stage 2 operation, the platform gimbal rates will be computed in block A168 so as to produce the pitch rate desired. The open-loop pitch profile will be continued through the staging interval and will be concluded upon computation of the first Stage 2 steering commands (A223 and A229).

## D. STAGE 2 -- CLOSED-LOOP STEERING

Block A155 will control the time to initiate the Stage 2 steering computations. Two passes are used to initialize $W_N$ (A208) in Stage 2. During this time, the open-loop pitch maneuver is continued for approximately 1 sec. On the third entry into the Stage 2 equations, the commanded attitudes and rates obtained from the explicit steering equations are computed.

The equations as programmed will steer the vehicle into a plane, defined by the ephemeris data, to the height desired for insertion and will orient the velocity vector so as to achieve the desired orbit. When the desired magnitude of velocity is reached, an engine shutdown command (A234) will be given. This function is discussed in detail in the following paragraphs.

The vehicle velocity perpendicular to the orbit plane ($V_\perp$), the velocity along the vehicle radius vector from the center of the earth ($V_P$), and the vehicle position perpendicular to the orbit plane ($P_\perp$) will be computed in block A203. These quantities will be used in A221 and A222 to compute the vehicle commanded angles and rates. Actual pitch attitude of the vehicle with respect to its local horizontal ($\beta_M$) and actual yaw attitude with respect to the orbit plane ($\Delta Y$) will be computed in A227. The quantities will be used in A229 to determine the pitch ($\theta'_0$) and yaw ($\psi'_0$) attitude errors of the vehicle. $\theta'_0$ and $\psi'_0$ are then inserted into A256 where the fast-loop attitude errors are generated. The desired vehicle pitch rate ($\dot{P}$-Block A223) is used in A168 to obtain the desired gimbal rates which, in turn, are also used in the fast-loop attitude error equations. ($\dot{P}$, as computed above, does not include any excess rate which might be required to bring the vehicle to the commanded pitch attitude.)

Time-to-go ($T_G$) is continuously tested in A232. When this quantity is reduced to approximately 2 sec., the attitude errors (A236) will be set to zero (thus allowing the vehicle rates to go to zero) and a fast countdown on SECO will begin. At the proper time (A232), a SECO signal (A234) will be delivered.

## E. ORBIT VELOCITY ADJUST

The ascent equations provide a capability to refine the spacecraft velocity to meet the insertion conditions. The equations do this by using spacecraft energy. The capability is required for the following reasons:

1. The guidance system may not satisfy these insertion conditions accurately.
2. Uncertainties associated with residual thrust of the vehicle may exist following insertion.
3. The payload capability of the booster may fall short of the energy required to meet insertion.

When the payload capability of the booster falls short of the insertion conditions, a test is provided on $\Delta Q$ (A205). This test allows entry into the orbit velocity adjust equations even when a SECO signal is not delivered.

The perturbations from nominal insertion conditions ($V_\perp$, $V_P$, $\delta V$, and $\delta R$) are computed continuously and form the basic inputs for the orbit velocity adjust equations. Approximately 20 sec. (A249, A281) is allowed to elapse before any commands are generated for spacecraft thrusting. This elapse allows the zero attitude error signals (A236) to remain available for the launch vehicle during thrust decay; it also allows the astronaut time to separate the spacecraft from the launch vehicle.

Following this time, the commanded platform roll gimbal angle will be set to approximately zero, and the spacecraft attitude errors will be computed. The astronaut will first roll the vehicle approximately 90 deg. to null the roll error. This will be done in response to the roll attitude error display in the capsule. He will then null the yaw and pitch error appearing on the same display.

The horizontal velocity will then be computed. This velocity will be either added to or subtracted from the spacecraft at perigee to reach apogee and at apogee to reach perigee. These quantities will appear on the $\Delta V$ meters in the spacecraft. While nulling out the attitude errors, the astronaut will then thrust the vehicle to either add or subtract the velocity appearing on the $\Delta V$ indicators. As the spacecraft approaches the desired apogee (A246), the velocity to be added or subtracted at perigee will go to zero. When this condition is reached, thrusting is discontinued.

The velocity to be added or subtracted at apogee will be recorded by the astronaut. At this point, ascent guidance is essentially concluded. The astronaut will use the "Catch-Up" mode of the computer to obtain the velocity increment desired at apogee.

## F. PLATFORM UPDATES

The computer has the capability of accepting velocity data from the ground to correct for platform misalignment, and integrated errors in the platform (A127, A128, A129 and A138 through A147). The use of velocity data in correcting the azimuth orientation of the platform is discussed in Section II-B of this appendix.

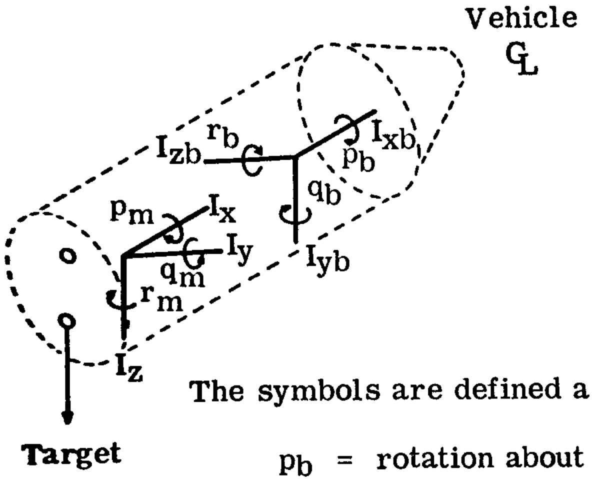
The early updates in flight ($t < 240$ sec.) will be used to correct the azimuth alignment of the platform; the updates following this time will be used to correct the measured platform velocities.

## G. SWITCHOVER FADE-IN

One additional feature, which is not shown on the math flow but is scheduled to be inserted into the computer, is the set of equations used to fade in the attitude error signals when switchover occurs from the primary system to the IGS. These equations will have the effect of fading in the attitude errors when a slow drift malfunction occurs. They will also allow the major percentage of the attitude error signal to be delivered to the autopilot when a rapid malfunction occurs. The result is to allow maximum control and response during rapid malfunctions and limited response during slow drift malfunctions.

## II. DERIVATIONS

### A. RELATIONSHIP BETWEEN ANGULAR RATES OF GIMBALS, SPACECRAFT AND LAUNCH VEHICLE



This diagram shows the relationship between the launch vehicle and spacecraft reference axes.

The symbols are defined as follows:

$p_b$ = rotation about body $x_{axis}$ - spacecraft roll, positive as indicated

$q_b$ = rotation about body $y_{axis}$ - spacecraft pitch

$r_b$ = rotation about body $z_{axis}$ - spacecraft yaw

$p_m$ = vehicle roll

$q_m$ = vehicle pitch

$r_m$ = vehicle yaw

The following relationships exist:

$$\begin{aligned} p_b &= p_m & \qquad \omega_{xb} &= \omega_{xm} \\ q_b &= r_m & \qquad \omega_{yb} &= \omega_{zm} \\ r_b &= -q_m & \qquad \omega_{zb} &= -\omega_{ym} \end{aligned}$$

The autopilot command signals are documented as follows: To cause the launch vehicle to rotate in a positive direction ( + roll, + pitch, and + yaw as defined above), the d-c voltage signal to the autopilot must be negative. The derivation of the relationship between gimbal rates and body rates is equivalent to the development of Euler's kinematical equations with specific application to the GEMINI gimbal system.

It can be shown that

$$\omega_{xb} = \dot{\theta} \sin \psi + \dot{\phi},$$

$$\omega_{yb} = \dot{\theta} \cos \phi \cos \psi + \dot{\psi} \sin \phi,$$

and

$$\omega_{zb} = -\dot{\theta} \cos \psi \sin \phi + \dot{\psi} \cos \phi .$$

Also,

$$\omega_{x} = \dot{\theta} \sin \psi + \dot{\phi},$$

$$\omega_{y} = \dot{\theta} \cos \psi \sin \phi - \dot{\psi} \cos \phi,$$

and

$$\omega_{z} = \dot{\theta} \cos \phi \cos \psi + \dot{\psi} \sin \phi .$$

Multiplying by $\tau$, we have

$$\Delta p_m = \Delta\theta \sin \psi + \Delta\phi,$$

$$\Delta q_m = \Delta\theta \cos\psi \sin\phi - \Delta\psi \cos\phi,$$

and

$$\Delta r_m = \Delta\theta \cos \phi \cos \psi + \Delta\psi \sin\phi .$$

$\Delta p_m$ = Launch vehicle roll change in time $\tau$,

$\Delta q_m$ = Launch vehicle pitch change in time $\tau$,

and

$\Delta r_m$ = Launch vehicle yaw change in time $\tau$.

Also

$\Delta\theta$ = Change in $\theta$ in time $\tau$,

$\Delta\psi$ = Change in $\psi$ in time $\tau$,

and

$\Delta\phi$ = Change in $\phi$ in time $\tau$.

To implement a change in gimbal angle, the following notation will be assigned to present and desired gimbal angles:

$\theta_b, \psi_b, \phi_b$ = present gimbal angles

$\theta_N, \psi_N, \phi_N$ = desired gimbal angles

Now $\Delta\theta = \theta_N - \theta_b,$

$\Delta\psi = \psi_N - \psi_b,$

and

$\Delta\phi = \phi_N - \phi_b.$

As previously mentioned, a negative polarity is required to cause a positive vehicle rotation. Therefore, $\Delta p_m$, $\Delta q_m$, and $\Delta r_m$ must be multiplied by -1 before being converted to analog signals. This can be accomplished by the following operations:

$\Delta\theta_c = \theta_b - \theta_N$

$\Delta\psi_c = \psi_b - \psi_N$

$\Delta\phi_c = \phi_b - \phi_N$

Subscript c denotes commanded change.

When a specific body angular rate is desired such as a pitch only maneuver during Stage 1, the following relationships are required:

$$\tau\omega_x = \Delta\phi + \Delta\theta \sin\psi$$

$$\tau\omega_y = \Delta\theta \cos\psi \sin\phi - \Delta\psi \cos\phi$$

$$\tau\omega_z = \Delta\theta \cos\phi \cos\psi + \Delta\psi \sin\phi$$

or

$$\Delta\theta = \tau\omega_y \frac{\sin\phi}{\cos\psi} + \tau\omega_z \frac{\cos\phi}{\cos\psi}$$

$$\Delta\psi = \tau\omega_z \sin\phi - \tau\omega_y \cos\phi$$

$$\Delta\phi = \tau\omega_x - \tau\omega_y \frac{\sin\phi \sin\psi}{\cos\psi} - \tau\omega_z \frac{\cos\phi \sin\psi}{\cos\psi}$$

Setting

$\omega_x = \omega_z = 0$, and $\omega_y = \dot{P}$, we have

$$\dot{\theta} = \dot{P} \frac{\sin\phi}{\cos\psi},$$

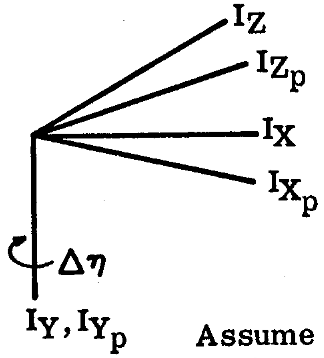
$$\dot{\psi} = -\dot{P}\cos\phi,$$

and

$$\dot{\phi} = -\dot{\theta}\sin\psi.$$

## B. UPDATE OF COMPUTED ANGLES AND VELOCITY

Consider a platform which has been erected to the vertical precisely, but is misaligned by an angle $\Delta\eta$ about the platform Y axis.



$I_X, I_Y, I_Z$ = Desired platform orientation.

$I_{X_p}, I_{Y_p}, I_{Z_p}$ = Actual platform frame.

$\Delta\eta$ = Misalignment in azimuth, positive rotation about platform Y axis

Assume that the ground tracking device can perfectly measure velocity and transform the velocity into the desired platform frame. This frame is known explicitly.

Now

$$V_X = V_{O_X} + \int_0^t a_X \, dt,$$

$$V_Y = \int_0^t a_Y \, dt,$$

and

$$V_Z = V_{O_Z} + \int_0^t a_Z \, dt.$$

$V_X$, $V_Y$, and $V_Z$ = actual velocities in desired frame as well as measured velocities by ground station.

$V_{O_X}$ and $V_{O_Z}$ = calculated velocity due to earth's rotation in the desired frame.

With a misoriented platform,

$$V_{Xp} = V_{O_X} + \int_0^t a_X \cos \Delta\eta \, dt - \int_0^t a_Z \sin \Delta\eta \, dt,$$

$$V_{Yp} = \int_0^t a_Y \, dt,$$

and

$$V_{Zp} = V_{O_Z} + \int_0^t a_Z \cos \Delta\eta \, dt + \int_0^t a_X \sin \Delta\eta \, dt.$$

Here, $V_{O_X}$ and $V_{O_Z}$ are values inserted during initialization.

If $\Delta\eta$ is small and constant,

$$\cos \Delta\eta = 1,$$

$$\sin \Delta\eta = \Delta\eta ,$$

and

$$V_{Zp} = V_{O_Z} + \int_0^t a_Z \, dt + \Delta\eta \int_0^t a_X \, dt.$$

Now

$$V_{Zp} - V_Z = \Delta\eta \int_0^t a_X \, dt = \Delta\eta \, (V_X - V_{O_X}),$$

or

$$\Delta\eta = \frac{V_{Zp} - V_Z}{V_X - V_{O_X}}$$

It is now possible to establish the relationship between measurements in the desired and actual frames.

$$\begin{bmatrix} I_{Xp} \\ I_{Yp} \\ I_{Zp} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\Delta\eta \\ 0 & 1 & 0 \\ \Delta\eta & 0 & 1 \end{bmatrix} \begin{bmatrix} I_X \\ I_Y \\ I_Z \end{bmatrix}$$

This matrix must be used to transform ground velocities into the platform frame.

$$V_{Xp}' = V_X - \Delta\eta \; V_Z$$

$$V_{Yp}' = V_Y$$

$$V_{Zp}'' = \Delta\eta \; V_X + V_Z$$

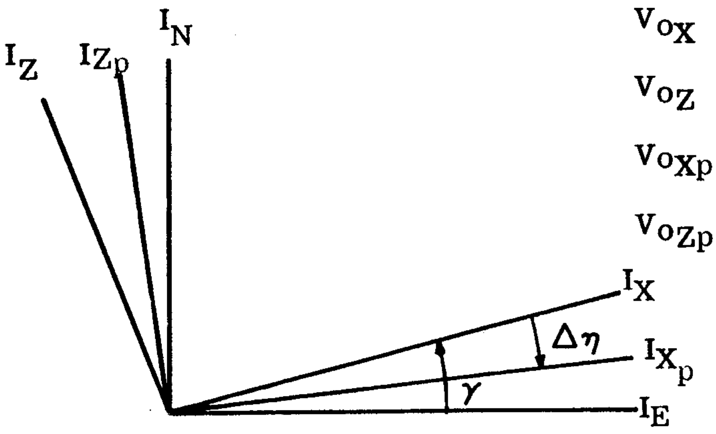
Future updates now require that the primed values denoting ground measured velocity be transformed to calculated platform axes. Since $\eta_X$ is originally assumed equal to zero, the following relationships are valid for the general case of angle update:

$$V'_{Zp} = \eta_X V_X + V_Z$$

$$\Delta\eta = \frac{V_{Zp} - V_{Z'p}}{V_{Xp} - V_{oX}}$$

$$\eta_{X_i} = \eta_{X_{i-1}} + \Delta\eta$$

The effect of platform misalignment on initial calculated velocity errors ($V_{oX}'$ and $V_{oZ}'$) will now be investigated.



Initial velocity - $C_o I_E$

$$V_{oX} = C_o \cos\gamma$$

$$V_{oZ} = C_o \sin\gamma$$

$$V_{oXp} = C_o \cos(\gamma - \Delta\eta)$$

$$V_{oZp} = -C_o \sin(\gamma - \Delta\eta)$$

By expanding and making small angle approximations, we have

$$V_{o_{Xp}} = C_o \cos\gamma + C_o \sin\gamma\Delta\eta = V_{o_X} - V_{o_Z}\Delta\eta ,$$

and

$$V_{o_{Zp}} = C_o \sin\gamma + C_o \cos\gamma\Delta\eta = V_{o_Z} + V_{o_X}\Delta\eta .$$

$V_{Xp}$ and $V_{Zp}$, however, must be updated to reflect the error in initial values. Therefore,

$$V_{Xp_c} = V_{Xp} - V_{o_Z}\Delta\eta ,$$

and

$$V_{Zp_c} = V_{Zp} + V_{o_X}\Delta\eta .$$

Positions X and Z must also be updated. So,

$$X_c = X - V_{o_Z}\Delta\eta \; T,$$

and

$$Z_c = Z + V_{o_X}\Delta\eta \; T$$

T is the elapsed time from platform release until update.

## C. COMPUTATION OF VEHICLE AZIMUTH AND PITCH ANGLE ORIENTATION

During Stage 2 flight, the actual vehicle attitude is required to implement the steering commands. The required angles are shown in Figure E-2.

$\bar{R}_b = \bar{I}_x X + \bar{I}_y Y + \bar{I}_z Z$ defines the vehicle position in the platform frame

The X-axis of the launch vehicle can be defined with respect to the platform by using the following standard GEMINI platform-to-body relationships:

$$\begin{bmatrix} \bar{I}_{Xb} \\ \bar{I}_{Yb} \\ \bar{I}_{Zb} \end{bmatrix} = \begin{bmatrix} & & \\ & \begin{matrix} b_{ij} \\ \text{Platform to} \\ \text{Body} \end{matrix} & \\ & & \end{bmatrix} \begin{bmatrix} \bar{I}_X \\ \bar{I}_Y \\ \bar{I}_Z \end{bmatrix}$$

and

$$\bar{I}_{Xb} = \bar{I}_X \cos\psi_b \cos\theta_b - \bar{I}_Y \cos\psi_b \sin\theta_b - \bar{I}_Z \sin\psi_b$$

Also

$$\bar{R}_b \qquad \bar{I}_{X_b} = //R// \cos\theta = R \sin\beta_M \ ,$$

and

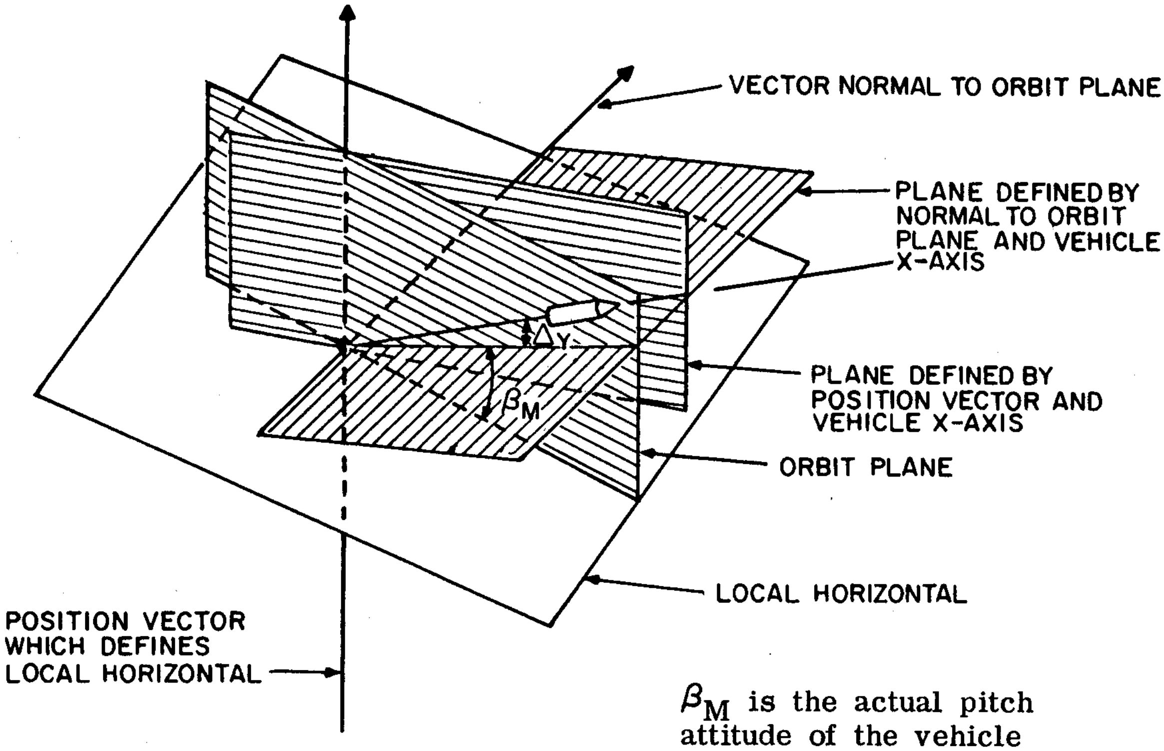
$$\sin\beta_M = \frac{X}{R} \cos\psi_b \cos\theta_b - \frac{Y}{R} \cos\psi_b \sin\theta_b - \frac{Z}{R} \sin\psi_b$$

Similarily, a unit vector perpendicular to the orbit plane has been defined in terms of the platform axis:

$$\bar{I}_\perp = \eta_x \bar{I}_X + \eta_y \bar{I}_Y + \eta_z \bar{I}_Z$$

By using a similar equation,

$$\sin \Delta Y = \eta_x \cos\psi_b \cos\theta_b - \eta_y \cos\psi_b \sin\theta_b - \eta_z \sin\psi_b$$



$\beta_M$ is the actual pitch attitude of the vehicle with respect to the local horizontal.

$\Delta Y$ is the azimuth angle with respect to the orbit plane.

Figure E-2. Vehicle Azimuth and Pitch Angle Orientation

## D. SUMMARY

The guidance equation portion of the simulation consists of the following subroutines: (1) DI (discrete inputs, (2) FAST (IGS fast-loop attitude error computations), (3) GUID 1 (Stage 1 open-loop guidance equations) and (4) GUID 2 (Stage 2 closed-loop guidance equations). Fortran listings of these programs are included in Table E-II.

### 1. Differences Between Math Flow and Equations Used in Simulation

To date, simulation effort has been primarily devoted to the verification of the compatibility of the IGS equations and the vehicle model. The following portions of the IGS math flow remain to be programmed and/or exercised:

- Test for Lift-off
- Gimbal Angle Update Following Platform Release
- Launch Azimuth Offset
- Platform Velocity Update
- Orbit Velocity Adjust
- Stage 2 SECO

All of the above items will be incorporated into the simulation prior to the final simulation report.

### 2. Operation of IGS Guidance Simulation (See Figures II-1 and D-2)

The guidance equations accept ephemeris data from the main program. Guid 1 then updates this data while waiting on platform release. Upon release, Guid 1 begins navigation; inputs to the navigation equations (the measured accelerations) are calculated by the main environment. After a nominal elapsed time increment, lift-off is simulated and fast-loop attitude error computations are begun. The fast loop accepts the gimbal angles from the main environment which is keeping track of vehicle inertial attitude. Commanded gimbal angles are generated in either Guid 1 or Guid 2. The attitude errors generated in the fast loop are then used as inputs to the simulated autopilot and vehicle model which, in turn, send vehicle angular rates back to the main program. The main program uses these rates to update the vehicle gimbal angles. Subsequently, another fast loop computation cycle may be performed.

With the exception of a few discretes, no other data need be transferred between the environment and IGS equations.

The fundamental computation cycle time used in the simulation of the Guid 1 as well as Guid 1 and 2 equations is 0.5 sec. This increment may become slightly smaller for the Guid 1 operations and slightly larger for the Guid 1 and 2 operations when the computation cycle time of the IGS equations as programmed in the GEMINI computer is defined more accurately.

Table E-2

## SYMBOLS USED IN ASCENT GUIDANCE SIMULATION (Stage 2)

### A. CONSTANTS

| Symbol | Env. Fortran Symbol |
|---|---|
| A3 | 52K ( 1) |
| A4 | 52K ( 2) |
| A5 | 52K ( 3) |
| A6 | 52K ( 4) |
| A7 | 52K ( 5) |
| A8 | 52K ( 6) |
| A9 | 52K ( 7) |
| $\omega_N$ | 52K ( 8) |
| QN | 52C ( 1) |
| $V_P$ | 52C ( 2) |
| $V_\lambda$ | 52C ( 3) |
| $V_g$ | 52C ( 4) |
| $\bar{a}_f$ | 52C ( 5) |

| Symbol | Env. Fortran Symbol |
|---|---|
| C* | 52C ( 6) |
| KQ | 52C ( 7) |
| $V_a$ | 52C ( 8) |
| $T_E$ | 52C ( 9) |
| $V_{\lambda F}$ | 52C (10) |
| $K^2_4$ | 52C (11) |
| $K^{12}_1$ | 52C (12) |
| $t_{cc}$ | 52C (13) |
| $R_F$ | 52C (14) |
| $t_K$ | 52C (15) |
| $t_{S1}$ | 52C (16) |
| $G_{epf}$ | 52C (17) |

## Table E-2. Symbols Used In Ascent Guidance Simulation (Stage 2) (cont)

### B. VARIABLES

| Symbol | Env. Fortran Symbol | * From |
|---|---|---|
| $t$ | TIME | E |
| $\theta B$ | THEB | E |
| $\psi B$ | PSIB | E |
| N20 | A20 | 1 |
| $\delta\epsilon_P$ | DELEP | 2 |
| $\delta\epsilon_Y$ | DELEY | 2 |
| $\Delta\phi_{LV}$ | ER | 2 |
| $\Delta\theta_{LV}$ | EP | 2 |
| $\Delta\psi_{LV}$ | EY | 2 |
| $V_x$ | VP (1) | 1 |
| $V_y$ | VP (2) | 1 |
| $V_z$ | VP (3). | 1 |
| $x$ | RP (1) | 1 |
| $y$ | RP (2) | 1 |
| $z$ | RP (3) | 1 |
| $\eta_x$ | ETX | 1 |

| Symbol | Env. Fortran Symbol | * From |
|---|---|---|
| $\eta_y$ | ETY | 1 |
| $\eta_z$ | ETZ | 1 |
| $g$ | GT | 1 |
| N25 | A25 | 1 |
| N26 | A25 | 1 |
| $\Delta t_S$ | DTS | 2 |
| LC29 | LC29 | 2 |
| $\Delta t_{SECO}$ | —— | 2 |
| $\dot{P}$ | P1 | 2 |
| $\phi f$ | PHIF | 1 |
| LCA232 | LCA232 | 2 |
| $R$ | RM | 1 |
| $\delta_1$ | — | — |
| PS2D | PS2D | K |
| LC24 | — | — |

*E - Environment
1 - 1st Stage Guidance
2 - 2nd Stage Guidance
K - Constant

```
C         BOB COFER
C         JAN.  ,1963
         SUBROUTINE GUID1
         DIMENSION VPO(3),VPI(3),F(3),G(3),RP(3),VP(3),X(3),V(3),VUP(3),
        CVG(3),DVP(3),V1(3),DV5(3)   ,DV1(124),FPO(3),DV6(24),RRP(3)
        C ,ROFE(3)
          COMMON      DV1,DI1,DI2,DI3,
        C             XINCL,DFL,DT,OMEGAE,OMEGAP,C0,C5,Y0,TU,A1,A2,GK,TBECO,
        C             TRF,TR,TP1,TP2,TGC,CP1,CP2,C1,C2,G1P,FP,   DV5,JUP,F,
        C  VG,
        C     THEB,  PSIB,PHIB,THEFL,PSIFL,PHIFL,DTHEN,DPSIN,DPHIN,
        C             A20,A21,A22,A23,A24
        C ,DV6,VP,RP,ETX,ETY,ETZ,GT,A25,A26,DTS,LC29,PHIN ,P1,PHIF ,LCA232,
        C RM,DEL1,MS2D,LC24,DTS
      9  FORMAT (6E18.8)
         IF (LC1) 114,113,113
    113 RRP(2) = -.20909749E8
         RP(2) = -.20909749E8
        SINI = SINF(XINCL)
      7 COSI = COSF(XINCL)
         LC1 = -2
    114  N = N+ 1
         IF (LCA232) 118,119,119
    119  IF (LC29) 112,117,117
    117  IF (N - 11) 111,112,112
    111  IF (DI2) 116,115,115
    115  CALL DI
    116  CALL FAST
         RETURN
    118  DT = DTS
    112  N = 0
     71 T = T + DT
        IF (LC18) 73,72,72
     72 T = 0.
     73 IF (LC29) 51,84,84
     84  DFL = DFL + DT * (OMEGAE - OMEGAP)
         B1 = SINF(DFL)
         B2 = COSF(DFL)
        SGM = +SINI*B2
        GM = SGM*(1. - SGM*SGM/6.)
     85 IF (LC26) 13,11,11
     11 IF (DI3 )13,12,12
     12  SDL = -A1*COSI + A2*SINI*B1
        CGM = 1.-SGM*SGM/2.
```

```
      VPO(1)= CO*CGM
      VPO(3)=-CO*SGM
      VP(1) = VPO(1)
      VP(3) = VPO(3)
      ETY = SDL
      ETZ = 1.
      PHIN = C5 - GM
      THEN = 1.5707963
      PHIN1 = 0.
      THEN1 = 0.
      PSIN1 = 0.
      PHIFL = PHIN
      THEFL = THEN
      PSIFL = PSIN
      PHIB = PHIN
      THEB = THEN
      PSIB = PSIN
       GO TO 53
   13 R2 = RP(1)**2 + RP(2)**2 + RP(3)**2
      RM = SQRTF(R2)
      GT =GK/R2
  131 XKG = GT/RM
      DO 18 I = 1,3
      G(I) = XKG *RP(I)
   18 VPI(I) = VP(I)
       FSTORE = F(2)
       F(2) = -F(3)
       F(3) = FSTORE
  152 DO 16 I =1,3
       VP(I) = VP(I) + F(I) - G(I)*DT
       FPO(I) = F(I)
   16  F(I) = 0
  161 DO 17 I = 1,3
      ROFE(I)=ROFE(I)+(VP(I) + VPI(I))*QT/2.
   17 RP(I) =RRP(I) + ROFE(I)
       QT = DT
  171 IF (LC21) 25,19,19
   19 IF (TU - T) 23,21,21
   21 T1 = T
      DO 22 I = 1,3
   22 V1(I) = VP(I)
      GO TO 31
   23 DELT = (TV - T1)/DT
      DO 24 I = 1,3
```

```
 24 VUP(I) = V1(I) + DELT*(VP(I)- V1(I))
    TU = TU + 50.
    GO TO 31
 25 LC21 = 2
    IF (308.- TU) 28,26,26
 26 ZDS = VUP(1)*ETX + VUP(2)*ETY + VUP(3)*ETZ
    DPSI =(VG(3) - ZDS)/VUP(1)
    DVP(1)= +DPSI*VPO(3)
    DVP(3)= -DPSI*VPO(1)
    VP(1) = VP(1) + DVP(1)
    VP(3) = VP(3) + DVP(3)
    RP(1) = RP(1) + DVP(1)
    RP(3) = RP(3) + DVP(3)
    ETX = ETX - DPSI*ETZ
    ETZ = ETZ + DPSI*ETX
    GO TO 31
 28 TU = TU - 40.
    DO 29 I = 1,3
 29 VP(I) = VP(I) + VG(I) - VUP(I)
 31 Z1 = SINF(PHIB)
    Z2 = COSF(PHIB)
    Z4 = COSF(PSIB)
 32 A20 = SINF(PSIB)
    A21 = Z4 * Z2
    A22 = Z1
    A23 = Z4 * Z1
    A24 = Z2
    A25 = Z4 *SINF(THEB)
    A26 = Z4 *COSF(THEB)
     IF (LCA232) 322,323,323
322  LCA232 = 2
     RETURN
323 IF (TBECO - T) 52,33,33
 33 IF (LC18) 36,34,34
 34 IF (DI2 ) 35,53,53
 35 LC18 = -2.
     TR = TRF - 45.836624*(C5 -GM -1.5707963 +C1*SDL + C2*CDL)
    PHIF = PHIN -       (TRF - TR) *.021816616
 36 PHIFL = PHIN
    THEFL = THEN
    PSIFL = PSIN
361 IF (TR - T) 37,49,49
 37 IF (TRF - T) 39,38,38
 38 PHIN1 =-.021816616
```

```
      PHIN = PHIN + PHIN1*DT
      GO TO 49
   39 IF (TP1 -T) 42,41,41
   41 PHIN = PHIF
      PHIN1 = 0.
      GO TO 49
   42 IF (TP2 -T) 44,43,43
   43 P1 = CP1
      GO TO 48
   44 IF (TGC - T) 45,46,46
   45 D057 = -2
   46 P1 = CP2
   48 PSIN1 =-P1*COSF(PHIN)
      THEN1 = P1*SINF(PHIN)/COSF(PSIN)
      PHIN1 =-THEN1*SINF(PSIN)
      PHIN = PHIN + PHIN1*DT
      THEN = THEN + THEN1*DT
      PSIN = PSIN + PSIN1*DT
   49 DPSIN = PSIN1*DT/10.
      DTHEN = THEN1*DT/10.
      DPHIN = PHIN1*DT/10.
       GO TO 53
   51  CALL GUID2
       RETURN
   52  CALL GUID2
       GO TO 48
 53   GO TO 114
       END
*     WADDING  6059   GEMINI ASCENT GUIDANCE                                  48
C      JAN.  ,1963
C     6059  GEMINI ASCENT GUIDANCE-2ND STAGE  DICK WADDING
      SUBROUTINE GUID2
      DIMENSION D1(24),D2(5),D3(49),D4(6),D5(4),D6(2),D7(17),D1A(25),
     C D3A(31)
       COMMON D1,T,D1A,
     C         A3,A4,A5,A6,A7,A8,A9,WN,D2,QN,VF,VL,VG,AFB,CS,CK,VA,TE,V
     1LF,C42,C112,TCC,RF,TK,TS1,GEPF,D3,TAU,D3A,
     C  TB,PB,PHB,D4,EN20,D5,TP0,PP0,D6,DPLV
     20,DTLV0,DSLV0,D7,X1,Y1,Z1,X,Y,Z,ETX,ETY,ETZ,C3,EN25,EN26,DTS,LC29,
     3PHN ,P1,PF,LCA232,R,DEL1,MS2D,DTC2
      IF (LC29) 500,40,40
 40   VN2=X1**2+Y1**2+Z1**2
      LCA232=1
      VN=SQRTF(VN2)
      VYP=VY
      VPP=VP
      VY=-X1*ETX+Y1*ETY+Z1
 9    VP=(X*X1+Y*Y1+Z*Z1)/R
      R1=-ETX*X+ETY*Y+Z
      H1=R1
      IF (LC24) 90,99,99
 90   IF (TS0-T) 92,91,91
 91   RETURN
 92   D064=1
      DELV=VF-VN
      DELR=RF-R
      IF (DELV-A3) 94,93,93
```

```
      DVE=DV-AFB*DTU
      QNP=QN
      QN=DV/CS
      DQ=QN-QNP
      IF (ABSF(DQ)-CK) 100,100,103
100   IF (ABSF(VF-VN)-VA) 102,101,101
101   SENSE LIGHT 4
      WRITE OUTPUT TAPE 9,1010
1010  FORMAT (20H ERROR AT BLOCK A206)
      RETURN
102   LC24=-1
      GO TO 521
103   WNP=WN
      WN=WNP*(1.+DQ*(1.+DQ*(.5+1./6.*DQ)))
      C4=C3*SGM
      VG=VG-TAU*C4
      IF (DEL1-1.5) 104,104,12
104   DEL1=DEL1+1.
      GO TO 200
12    VGF=VG-(TE-TAU)/2.*C4
      DTU=(VGF+VLF)/AFB
      DEL3=T-C42
      IF (DEL3) 14,13,13
13    VLF=VL-BLM*DVE
14    WS2=VN2*CGM**2/R**2
      WS=VN*CGM/R
      GEP=C3-WS2*R
      U1=(WNP-WN)/TAU
      AU=U1*CS
      TE=(WN-1.)/U1-DTU
      AT=AU/WN
      AF=AU/(1.+U1*DTU)
      AFB=.5*(AU+AF)
      ATT=AT*(1.-U1/2.*TAU/WN)
      SBG=GEP/AT
      BGNP=BGN
      BGN=SBG*(1.+SBG**2/6.)
      SBB=-VP/DVE
      BB=SBB*(1.+SBB**2/6.)
93    DELY=VY/DELV
94    IF (ABSF(DELY)-A4) 98,95,95
95    IF (DELY) 96,97,97
96    DELY=-A4
      GO TO 1800
97    DELY=A4
98    GO TO 1800
99    VI=1./VN
      SG=VP*VI
      CG2=1.-SG**2
      CG=.5*CG2+.5
      GM=SG*(1.+SG**2/6.)
      SGM=SG
      CGM=CG
10    DV=VF-VN+VL+VG
```

```
141   Q5=1.-DVE/AF/TE
      Q6=Q5/DVE*CS
      IF (T-C112) 15,15,16
15    C6=RF-R-TE*VP*(1.-Q6)
      WPR=C6/CS/TE**2/(Q6-.5)
      WYY=(-R1+TE*(Q6-1.)*VY)/(CS*TE**2*(Q6-.5))
16    WG=2.*WS+GEP/CS
      BYB=VY/DVE
      SS=TE*WPR*(Q6-1.)
161   BNPP=BGN+BB+SS
      BY=BYB-TE*(Q6-1.)*WYY
      P1=WPR-WS-WG
      BGF=GEPF/AFB
      BPP=BB+TE*Q6*WPR+BGF
      FLAM=.5*(BPP**2+BY**2)
      XL=.5*((BNPP-GM)**2+(BY-VI*Z1)**2)
      BLM =.5*(XL+FLAM)
      VL=VL-XL*TAU*ATT
1800  BM=ARSINF(X/R*EN26-Y/R*EN25-Z/R*EN20)
      SDT=ETX*EN26-ETY*EN25-EN20
      DT=SDT*(1.+SDT**2/6.)
      IF (LC24) 171,170,170
170   PPO=BY-DT
      TPO=BNPP-BM
      PFL=PB
      TFL=TB
      PN=PB
      TN=TB
      PHFL=PHN
      IF (TE-TK-TCC) 50 ,50,501
501   LCA232= 1
      GO TO 200
500   DTS=DTS+.05
50    DTC2=DTS-(TE-TK)
51    IF (DTC2) 52,53,53
52    LC29=-1
521   TS0=T+3.5
      TS1=T+20.
520   DTLV0=0.
      DPLV0=0.
      DSLV0=0.
      P1=0.
      GO TO 200
53    D064=-1.
      LCA232=-1
      LC29=1
      LC24=-1
      GO TO 520
171   IF (LC4F) 173,172,172
```

```
172   LC1=1
      LC3=1
      LC8=1
      LC4F=-1
173   IF (LC1) 175,174,174
174   GO TO 200
175   IF (TS1-T) 176,174,174
176   PHN=PF-1.5708
      DSSC=DT-DELY
      DTSC=BM
      DPSC=PHB-PHN
      COCO=.349066
      IF (ABSF(DSSC)-COCO) 178,178,177
177   DSSC=SIGNF(COCO,DSSC)
178   IF (ABSF(DTSC)-COCO) 180,180,179
179   DTSC=SIGNF(COCO,DTSC)
180   IF (ABSF(DPSC)-COCO) 182,182,181
181   DPSC=SIGNF(COCO,DPSC)
182   DRA=A5*VP**2+A6*DELV+A7*DELR
      DRP=-A5*VP**2-DELR
      DVXB=A8*DRA
      VGA=DVXB
      DVYB=A9*DRP
      VGP=DVYB
      DVZB=0.
200   IF (MS2D) 201,202,202
201   RETURN
202   WRITE OUTPUT TAPE 9,203,LCA232,LC29,LC24,LC1,LC3,LC8,LC4F,VN,VY,VP
     1,R1,D064,DELV,DELR,DELY,GM,DV,DVE,QN,DQ,WN,C4,VG,VGF,DTU,VLF,WS,GE
     2P,U1,AU,TE,AT,AF,AFB,ATT,BGN,BB,Q5,Q6,C6,WPR,WYY,WG,BYB,BNPP,BY,P1
     3,BGF,BPP,FLAM,XL,BLM,VL,BM,SDT,DT,PP0,TP0,PFL,TFL,PN,TN,PHFL,DTS,D
     4TC2,TS0,TS1,DTLV0,DPLV0,DSLV0,PHN,DSSC,DTSC,DPSC,DRA,DRP,VGA,VGP,D
     5VZB
203   FORMAT (7I5/(8E15.7))
       CALL PDUMP(D1(1),D1(500),1)
      RETURN
      END
```

```
C      BOB COFER
       SUBROUTINE FAST
       DIMENSIONDV1(125),DV2(35)
       COMMON  DV1,DI2,DV2,
     C           THEB,PSIB,PHIB,THEFL,PSIFL,PHIFL,DTHEN,DPSIN,DPHIN,A20
     C          ,A21,A22,A23,A24,DELEP,DELEY,DEP,DEY,ER,EP,EY
       IF (DI2) 3,4,4
    3  PHIFL = PHIFL + DPHIN
       PSIFL = PSIFL + DPSIN
       THEFL = THEFL + DTHEN
       DELEP = DELEP + DEP
       DELEY = DELEY + DEY
       DPSI =-PSIFL + PSIB
       DPHI =-PHIFL + PHIB
       DTHE =-THEFL + THEB
      ER=+( A20*DTHE + DPHI)
      EP=+((+A23*DTHE - A24*DPSI) - DELEP)
      EY=+((A21*DTHE + A22*DPSI) - DELEY)
    2  FORMAT (6E18.8)
       QQ =QQ +1.
       IF (QQ - FP) 4,5,5
    5  WRITE OUTPUT TAPE 9,2, ER,EY,EP,            PSIB,PHIB,THEB
       QQ = 0
    4  RETURN
       END
C      BOB COFER
       SUBROUTINE DI
       DIMENSION CT(3),DV1(118)
       COMMON DV1,CT,TFSPR,TFPR,DTIME,DI1,DI2,DI3
       TA= TA+ DTIME
       TFPR = TA -TFSPR
       IF (CT(3) - TA)3,3,8
    3  DI3 = -2.
       IF (LC1) 4,31,31
   31  TFSPR = TA
       LC1 = -2
    4  IF (CT(1) - TA) 5,5,8
    5  DI1 = -2.
    6  IF (CT(2) - TA) 7,7,8
    7  DI2 = -2.
    8  RETURN
       END
```