PRELIMINARY ASCENT GUIDANCE SIMULATION REPORT

(Analytic Report No. 6)

IBM No. 63-564-0159

ORIGINATING GROUP:

GEMINI Systems Engineering Staff

CONTENT APPROVED BY:

DATE: 31 January 1963

PURCHASE ORDER: Y20163

CONTRIBUTORS:

D. P. Bedford
R. E. Cofer
T. W. Hill
R. B. Jasinski
J. Park
J. R. Sims
R. V. Wadding

IBM

FSD SPACE GUIDANCE CENTER

OWEGO NEW YORK

FOREWORD

This preliminary report on GEMINI Ascent Guidance Simulation is submitted to the McDonnell Aircraft Corp. (MAC), St. Louis, Missouri, in partial fulfillment of requirements under Task 2, Booster Guidance Studies, Statement of Work, Purchase Order Y20163.

ABSTRACT

This report presents the results of preliminary studies on the simulation of GEMINI Ascent Guidance. These studies have been made to simulate the Stage 1 (open-loop) and Stage 2 (closed-loop) guidance equations to verify the compatibility of the computer equations with the vehicle and control system and, to some degree, to determine the accuracy of the equations. The simulation is described in detail along with derivations of equations used in several areas of the simulation and operational math flow.

The major results of the simulation studies discussed in Section III follow:

- The ability of the guidance equations to control the vehicle has been demonstrated.
- The navigation errors in the guidance equations are small.
- The equations implemented to steer the vehicle during Stage 2 will bring the vehicle to nominal insertion conditions.

In several areas, the simulation will deviate from the most recent math flow release. These deviations are indicated in the appendices. Plans for further study and refinement in the simulation are presented in Section IV.

TABLE OF CONTENTS

Section I

INTRODUCTION

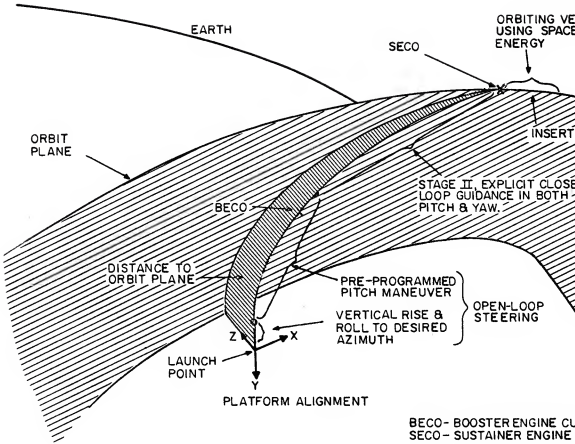


Figure I-1. GEMINI Ascent

Section I

INTRODUCTION

This report presents the results of preliminary studies on the simulation of GEMINI Ascent Guidance at the IBM Space Guidance Center. These studies have been made to simulate the Stage 1 (open-loop) and Stage 2 (closed-loop) guidance equations to verify the compatibility of the computer equations with the vehicle and control system and, to some degree, to determine the accuracy of the equations. Figure I-1 presents a typical ascent profile of a GEMINI mission. A detailed description of this profile is attached as Appendix E.

The simulation is described in detail along with derivations of equations used in several areas of the simulation and operational math flow.

In summary, the major results of the simulation studies discussed in Section III are as follows:

- The ability of the guidance equations to control the vehicle has been demonstrated.
- The navigation errors in the guidance equations are small. (See Figures III-14, -15, -17, and -18, and Table III-I in Section III.) The platform Z-axis velocity and position components differ from those in the environment; however, the addition of the deflection of the gravity vector will correct this situation (Appendix A).
- The equation implemented to steer the vehicle during Stage 2 will bring the vehicle to nominal insertion conditions.

In several areas, the simulation will deviate from the most recent math flow release. These deviations are indicated in the Appendices. Plans for further study and refinement in the simulation are presented in Section IV.

Section II

DESCRIPTION

Section II

DESCRIPTION

A. INTRODUCTION

Initially, the GEMINI Ascent Simulation was to include the use of IBM's combined analog-digital simulation facilities with a Pace analog computer for instrumentation of autopilot and rigid body equations. Subsequently, however, IBM determined that an all-digital simulation was feasible, and study was defined and initiated accordingly. As a result, Stage 1 simulation was conducted on both the combined and all-digital programs; Stage 2 simulation was conducted only on an all-digital program. Due to the organization of the program, the only redundancy in the Stage 1 program occurs in the autopilot and rigid body dynamic equations. The remainder of the programs are identical.

The entire simulation is based on the use of a main program, which defines the external world or environment (including time) into which a vehicle model is inserted along with a self-contained guidance computer. Each of these three subjects – main program, vehicle model, and guidance computer – are discussed in this section. Figure II-1 is a functional drawing of the program. Dotted lines on the drawing indicate the relationship between Figure II-1 and these three subject areas.

B. MAIN ENVIRONMENTAL PROGRAM

Data constants are read into the computer to initiate the program on the ascent problem. The main program computes the total vehicle acceleration – gravitational and perturbative. The perturbative accelerations are provided as accelerometer inputs to the guidance equations which, in turn, provide inputs to the autopilot. The vehicle angular rates, as outputs of the autopilot, are then transmitted to the main program. These rates are used to update vehicle attitude, including the platform gimbal angles, which are also provided as an input to the guidance equations.

The main program integrates total acceleration in the platform frame and updates position and velocity values. Aerodynamic forces are also computed in the main environment and are supplied to the vehicle model equations for inclusion in the solution of the rigid body equations.

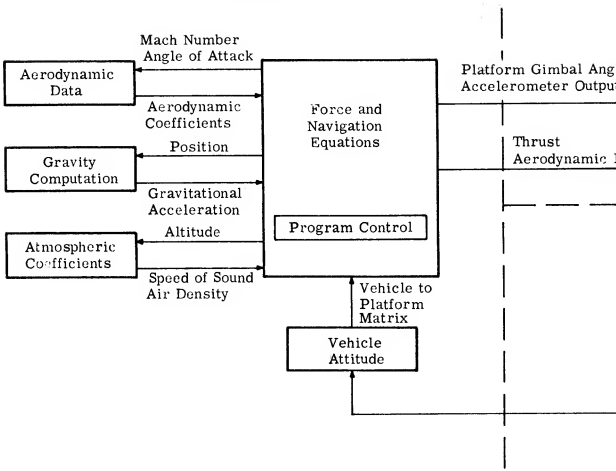


Figure II-1. GEMINI Ascent Guidance Simulation

In order to provide the ability for changing parts of the program without affecting the rest of the program materially, as many functions as possible were incorporated as subroutines to the primary program. Also, the program for the guidance equations and vehicle model were written as subroutines to the main program. In Figure II-1, note that functions such as aerodynamic data, gravitational acceleration, atmospheric coefficients and vehicle attitude determination are logical choices for inclusion as subroutines.

The coordinate systems employed in this simulation for presentation of results concur with GEMINI coordinate systems defined in the compatibility data sheets and various GEMINI Specification Control Drawings. The three basic coordinate systems include the platform, spacecraft and launch vehicle references. These systems as well as the various intermediate and computational frames are described in Appendices A and D.

Since computer subroutines previously written were available early in the program for the mathematical models which were used for atmospheric and gravitational terms, the subroutines were used without modification. Some variation exists between the models used and those intended for GEMINI simulation. However, specified models can be incorporated into the program by inserting the new subroutines, when available, without further modification to the program.

The basic timing of the program is such that one computer cycle is the equivalent of 0.05 sec. This value was chosen on the basis of limiting round-off error while still providing updated platform gimbal angles to the fast-loop guidance equations at the required rate (20 times/sec). Note that the slow loop guidance equations are entered once every ten computer cycles (2 times/sec). This closely estimates the real time computation cycle that will be experienced in the GEMINI computer.

Certain effects, assumed not to affect the simulation, were omitted: Thrust variation with change in altitude, body bending modes, fuel slosh, aerodynamic damping terms, gravitational oblateness beyond second harmonic, platform errors, misalignment errors, and errors in rate and angle terms resulting from displacement of instruments from the vehicle's center of gravity.

A complete description of the simulation program can be found in Appendix A, and the math flow and program listings are in Appendix D.

C. VEHICLE MODEL

The vehicle model is conveniently divided into two parts. The first consists of the autopilot and engine servos; the second deals with the solution of the rigid body dynamics. In defining these parts, IBM had to rely on available data which, unfortunately, has been revised subsequent to writing the program. The data affected by these revisions is associated primarily with the aerodynamic characteristics. Complete data sheets are available in Appendix G; however, the sources of this data are obscure at present. Although the data is not consistent with most recent compatibility data, the variation is small and represents a physical model differing only slightly from the currently defined GEMINI Launch Vehicle. In the case of the autopilot and engine servos, the model used agrees with the latest available compatibility data except that it has been simplified to a fourth order polynomial.

The Stage 1 vehicle model was programmed in two mediums – analog and digital. These analog and digital programs are described in Appendices B and C, respectively. In each program, the model is reached by the main computer through a specific subroutine. The subroutine varies to provide the proper interface and real time control for the analog or digital model.

While the rigid body dynamics is an integral part of the vehicle model, aerodynamic forces and location of the vehicle center-of-pressure are computed in the main environment and supplied as inputs to the vehicle model. The various vehicle constants (center-of-pressure location, aerodynamic coefficients) are stored as tabular data in the computer, and linear interpolation is used to evaluate specific values.

D. GUIDANCE COMPUTER

The model used for the guidance computer was formed by modifying the GEMINI Computer Ascent Guidance Math Flow as described in Appendix E. Additional logic was provided to permit entering specific parts of the program as a function of time. For example, in the case of fast-loop versus slow-loop computations, a simple counter is used to gate program flow around the slow loop nine out of ten times. An additional subroutine provides the discrete inputs required by the guidance computer. Due to the sequential nature of writing and checking the simulation program, Stage 2 guidance equations are written in the form of a subroutine to the Stage 1 equations.

In writing the programs for the guidance equations, some variation is expected in the actual performance of the guidance computer and the program. Some of these variations are as follows:

1. The simulation program is written in floating-point arithmetic with eight significant places.

2. Time increment for the slow loop is an exact multiple of the fast loop and is constant.

3. Time increment for both Stage 1 and Stage 2 slow-loop computations was assumed to be 0.5 sec.

4. Update logic was not exercised.

5. Orbit Velocity Adjust Equations were not simulated.

6. Quantization of the attitude error outputs by the D/A converters was not accounted for.

Section III

RESULTS

Section III

RESULTS

A. INTRODUCTION

The primary purposes of the simulation are to prove (1) the adequacy of the GEMINI IGS guidance equations and (2) compatibility between the computer, launch vehicle autopilot, and control system. The work performed to date has been devoted primarily to obtaining a functioning model to satisfy these goals. Certain refinements to the current model are required (see Section IV); however, a working model has been defined and has produced preliminary data. These data are presented in this section. The specific data runs to be considered are defined as follows:

- Run A — Combined analog-digital simulation, 0 to 150 sec., Stage 1 (Guid 1) equations. No aerodynamic forces considered. See Figures III-1 through III-4.
- Run B — Digital simulation, 0 to 150 sec., Stage 1 equations, aerodynamic forces included. See Figures III-5 through III-19.
- Run C — Same as Run A except aerodynamic forces included. See Figures III-5 through III-19.
- Run D — Digital simulation, 0 to 333 sec., Stage 1 and 2 equations used. Aerodynamic forces included for Stage 1 only. See Figures III-20 through III-28.

B. STAGE 1

1. GUID 1 NAVIGATION

A simplification was incorporated into the program which affected the calculation of the Z components of gravity (see Appendix A). In essence, this simplification neglects the fact that the actual platform is aligned to the local gravity vector rather than to the geocentric radius vector. Consequently, considerable error is experienced when computing the Z component of gravity in the platform frame (Figures III-16 and III-19).

Note that in Figures III-14, -15, -17, and -18, the errors experienced in velocity and position computations are within acceptable bounds. Specifically, the errors are as follows:

1. The maximum error in V_X occurs near 90 sec and attains a value of 0.07 ft/sec. This error then recedes to 0.025 ft/sec at the termination of Stage 1 Guidance.

2. The maximum error in R_X occurs near 120 sec, reaching a value of 2.7 ft (average V_X error of 0.02 ft/sec). This error then recedes to 1.6 ft at the termination of Stage 1 guidance.

3. The error in V_Y increases as a function of time, attaining a value of 0.14 ft/sec at the end of Stage 1 guidance.

4. The error in R_Y also grows as a function of time, reachin a value of 326 ft at the end of Stage 1 guidance. This discrepancy is unrealistic since the error in velocity, V_Y, is 0.07 ft/sec. or less. This apparent conflict can be attributed to computation round-off error in the environment equations, which are being solved at 10 times the rate of the IGS equations. Solutions to this problem have been established and will be included in future runs.

While the guidance equations do not result in appreciable navigation errors, remember, that the computations were programmed in 7090 word size with floating point arithmetic. The shorter word and fixed point operation in the actual GEMINI computer may result in more error. This is discussed in Section IV, "Plans for Continued Simulation Study."

2. SIMULATION NEGLECTING AERODYNAMIC FORCES

The primary purpose of effecting simulation Run A was to provide various checks on the program. In addition to simplifying program debugging, the various autopilot gains were verified. Figure III-1 illustrates the relationships between attitude errors, vehicle rates and engine deflection angles. Figures III-2, -3, and -4 are records of the angles of attack, velocity and position during ascent. Note the yaw angle of attack (α_Y) in Figure III-2. Ideally, this angle should start at zero, but due to a singularity in the computation equations at $t = 0$ ($\alpha_Y = \tan^{-1} V_{yb}/V_{xb}$, $V_{xb} = 0$ at $t = 0$) any spurious velocity computed for V_{yb} produces a finite angle of attack. This condition quickly corrects itself as V_{xb} obtains a finite value. The environment body-platform matrix is not updated during holddown, and can result in an error approaching 0.3 ft/sec. in V_{yb} at lift-off. At $t = 2$ sec., this would

result in an indicated angle of attack of approximately 0.3 deg. ($V_{Xb} \approx V_Y \approx 50$ ft/sec.). Further investigation and corrective action on this error will be performed even though its effect on the simulation is small.

3. REPEATIBILITY

The data runs presented in this report were compared to similar runs to demonstrate the repeatibility of results. The difference in data obtained from these comparisons was always two orders of magnitude less than the value of the function,or, in the case of zero magnitude, the difference approached the machine noise level (quantization). Since the largest variations were experienced in angles of attack, the actual comparison of this parameter is interesting to note. The maximum error occurs at 108 sec. (max α_N) and has a value of 0.08 deg. when the angle of attack α_N is 6.3 deg.

4. AUTOPILOT DELAY

Provisions in the digital autopilot permit the effective time of data transfer to vary from the guidance equations to the autopilot model. A computer run was made with zero time delay and compared with Run B, which includes a 20-msec delay. This was done in an effort to identify the differences between the combined and pure digital simulations. The results of this run have not been plotted, but the behavior of the difference terms between Run B and this run are similar to those between Run C and Run B.

Due to the mechanization of the guidance equations, there is a 50 msec (one computation cycle) delay in sending the attitude error signals to the analog autopilot. Thus, the combined simulation results lag the digital results. This is shown in Figures III-8, -9, and -10. The maximum variation in pitch angle of attack, which occurs immediately after autopilot gain change, is approximately 0.3 deg.

5. GUID 1 CONTROL

One of the primary objectives of the simulation program is to prove that the guidance equations, as programmed, are able to control the attitude of the launch vehicle. The various functions plotted in Figures III-7, -8, -9, and -11 demonstrate this controllability. Areas in the referenced Figures requiring additional comment are discussed below.

a. Roll Program Transient

The step seen in the roll attitude error signal (Figures III-1 and -11) at the conclusion of the roll program is due to the fact that the duration of the roll program is fundamentally controlled by the 0.5-sec. slow-loop computation

cycle. Essentially, the start and stop of the desired roll program have been computed correctly; however, the equations are such that the duration of roll maneuver is controlled to some multiple of the slow-loop computation cycle (0.5 sec.). The step may be undesirable, and will be corrected by setting $\phi_{FL} = \phi_f$ in Block A162 of Figure E-1A in Appendix E.

b. Pitch Angle of Attack Perturbation at Gain Change (α_N Excursion at 108 sec.)

In Figure III-8, note the magnitude of angle of attack, α_N, at 105 sec (just prior to gain change). Note also the perturbation on angle of attack following gain change. A detailed check on these results as well as discussions with other Contractors doing similar work indicate that the perturbation is reasonable. Presumably, the three-step pitch profile, which is expected to be available for the launch vehicle in February 1963, will minimize some of these effects. The pitch maneuver used in the present simulation is a two-step profile which approximates a gravity turn.

c. Pitch Rate

In Figure III-11, the pitch rate is slightly erratic in the time period 40-70 sec. This departure from the desired constant pitch rate occurs at a time when the angle of attack has an appreciable value and aerodynamic pressure is approaching a maximum. A detailed analysis of this variation will be performed in an attempt to explain this behavior.

d. Yaw Attitude

As mentioned above on the results of simulation which neglect aerodynamic forces, a change will be made to the simulation to remove the apparent errors in yaw angle of attack. Further discussion on this subject is not warranted here except to repeat that this error is attributed to the simulation rather than guidance equations, and the results of the error are small.

C. STAGE 2

Vehicle staging is simulated by assuming that Stage 1 shutdown, Stage 1 and Stage 2 separation, and Stage 2 ignition occur simultaneously after 148 sec. following lift-off. A more accurate simulation of this operation is being considered; however, it is not important in terms of the over-all IBM ascent simulation objectives.

A comparison of the environment and IGS navigation data at the time of Stage 2 ignition as well as at other discrete times during flight is contained in Table III-1. The data compares favorably. The major portion of the position difference on the Y-axis is believed due to round-off errors in the environment navigation equation and not in the IGS equations. The difference along the Z-axis is due to the difference in velocity obtained by the environment and the IGS equation. Both of these subjects are discussed more completely in the Stage 1 portion of this section and in Appendix A to this report.

The various programmed constants used in the Stage 2 guidance equations are contained in Appendices E and F.

Some of the initial conditions on the vehicle at the time of first explicit guidance steering commands are outlined in Table III-2. $P_{\perp}$ is -80,407.52 ft because Stage 1 roll azimuth offset (implicit steering) has not been used in the run presented. Note the magnitude of difference between the vehicle attitude and commanded attitude at this time. The magnitude of the out-of-plane position accounts for the large commanded yaw angle. The relatively large pitch was commanded because the flight path angle is low and the vehicle is low in the trajectory. Figures III-21 and -22 present the first pitch and yaw maneuvers of the Stage 2 steering equations. A fade-in technique was used in this run for approximately 20 sec. at the start of Stage 2 guidance. This sophistication is not in the present math flow (Figures E-1A and E-2A), and it is questionable whether something similar would be desired. If a trajectory of this type were actually experienced and a fade-in technique were not employed, a 6 deg. attitude error signal would be delivered to the autopilot. Consideration should be given to those effects as well as those due to malfunctions and switchover during this time if fade-in were employed. Note that the combination of IGS limiting and K-factors in the autopilot will limit these rates to 9 deg./sec.

Figures III-22 and -23 present additional steering data over the entire Stage 2 flight. The various computed quantities were rather stable, considering the magnitude of the initial changes required in vehicle orientation.

Vehicle attitude error signals are contained in Figure III-24. These signals are stable except near shutdown when they tend to deviate considerably as the equations near the pole at insertion. Although the attitude error data presented is smooth, Figure III-27 presents a detailed picture of the construction of the signals being delivered to the autopilot in this simulation. In all the data presented, the resolution of the commands in the simulation exceeds the resolution to be obtained in reading the platform gimbals and also exceeds the resolution to be expected from the output ladder networks of the computer. In order to better represent the physical system, plans are being made to include the effects of platform gimbal angle uncertainty and resolution as well as the output ladder network resolution in further simulation efforts.

Table III-1

COMPARISON OF ENVIRONMENT & IGS
NAVIGATION DATA DURING STAGE 2 FLIGHT

Parameter		Time of Flight			
		t = 148.0	t = 156.5	t = 331.5	t = 333.15
Position in ft.	X_E	469,940.36	543,112.46	3,102,063.3	3,143,765.6
	X_G	469,938.52	543,110.61	3,102,137.7	Not Computed
	Y_E	-21,081,311	-21,102,675	-21,185,532	-21,179,483
	Y_G	-21,081,672	-21,103,057	-21,186,309	Not Computed
	Z_E	7,248.3934	7,657.5726	86,528.575	86,522.584
	Z_G	7,764.1767	8,231.3245	88,933.193	Not Computed
	R_E	21,086,349	21,109,664	21,411,610	21,411,708
	R_G	Not Computed	21,110,046	21,412,399	Not Computed
Velocity in ft./sec.	$\dot{X}_E$	8,429.5037	8,789.8511	25,065.476	25,486.752
	$\dot{X}_G$	8,429.4739	8,789.8409	25,066.128	Not Computed
	$\dot{Y}_E$	-2,586.4419	-2443.0887	3602.0795	3780.4620
	Y_G	-2,586.3138	-2442.9486	3602.3227	Not Computed
	$\dot{Z}_E$	48.406679	47.870196	21.662948	-29.290453
	Z_G	55.049632	54.865414	35.436428	Not Computed
	V_E	8581.6850	9123.182	25,322.98	25,765.622
	V_G	Not Computed	9123.174	25,323.68	Not Computed

$E_{\sim}$ Environment; $G_{\sim}$ Guidance.

Note:

Last computed value of time to go, T_G, at 331.5 sec. equaled 1.650838 sec. This would define shutdown at 333.150838 sec.

R_F = 21,411,962 ft.

V_F = 25,765.9 ft./sec.

Table III-2

VEHICLE INITIAL CONDITIONS
AT INITIATION OF STAGE 2 STEERING COMMANDS
(t = 158.0 sec.)

Vehicle Velocity, V_η,	9,160.475	f_{PS}
Vehicle Velocity Perpendicular to Orbit Plane, $V_\perp$,	44.651	f_{PS}
Vehicle Velocity Along Radius Vector, V_P,	2,655.541	f_{PS}
Vehicle Position Perpendicular to Orbit Plane, $P_\perp$,	-80,407.52	ft.
Flight Inertial Path Angle, Γ,	0.2939	Radians
Vehicle Pitch Attitude, β_M	0.3287	Radians
Vehicle Commanded Pitch Attitude, β''_N	0.4843	Radians
Vehicle Yaw Attitude, ΔY	-0.0014	Radians
Vehicle Commanded Yaw Attitude, β_Y	-0.1953	Radians

The discontinuity in the attitude errors (Figure III-27) every 10 computation cycles is attributable to the introduction of new information from slow-loop calculations. Yaw rate terms are not used in the fast-loop equations; however, the pitch rate terms are included. For this reason, the discontinuity in the yaw channel may be expected to be greater than in the pitch channel, especially if heavy yaw steering exists.

Figure III-28 illustrates the vehicle angle of attack during Stage 2 flight. This information corresponds well with the data presented in Figures III-22 and -23. Some of the intermediate computer constants are shown in Figure III-25. Note that the loss terms at the end of flight are 770 ft/sec.for gravity and 190 ft/sec.for steering. These quantities may also be on the high side because of the magnitude of steering during Stage 2. Computation of V_{λ_f} on Figure III-25 is not started until approximately 200 sec. following lift-off. For this reason, the transient appears in δT_U at this time.

Plots of various Stage 2 steering parameters are included as Figure III-26. Unfortunately, the data does not extend to SECO and had to be extrapolated. Note that the operational math flow is set up to do a fast-loop countdown on time-to-go, T_G, beginning approximately 2 sec prior to shutdown. At the same time, the vehicle attitude error signals will be set to zero so as to zero out the vehicle rates. The GEMINI IGS System similarly will not be computing these quantities during flight. In the simulation, the environment is still keeping track of the vehicle over this period of time. Some of the results at shutdown are included in Table III-1.

Note that Figure III-26 indicates that R is approximately 650 ft greater than R_F, and the environment indicates Ru approximately 210 ft less. The difference in results can be explained by the Y-axis roundoff errors previously discussed. However, the IGS equations should have done better in making insertion altitude. Several reasons may be postulated, the most significant of which may be the bypassing of the pitch and yaw rate equations following 300 sec. Other sources such as differences between β_N'' and P may contribute to the difference in results. This area requires more detailed investigation.

The quantities V_η, V_p, $V_\perp$ in Figure III-26 are all nominally small at or near insertion. Table III-1 would tend to indicate that vehicle velocity is within 0.3 ft/sec of V_F.

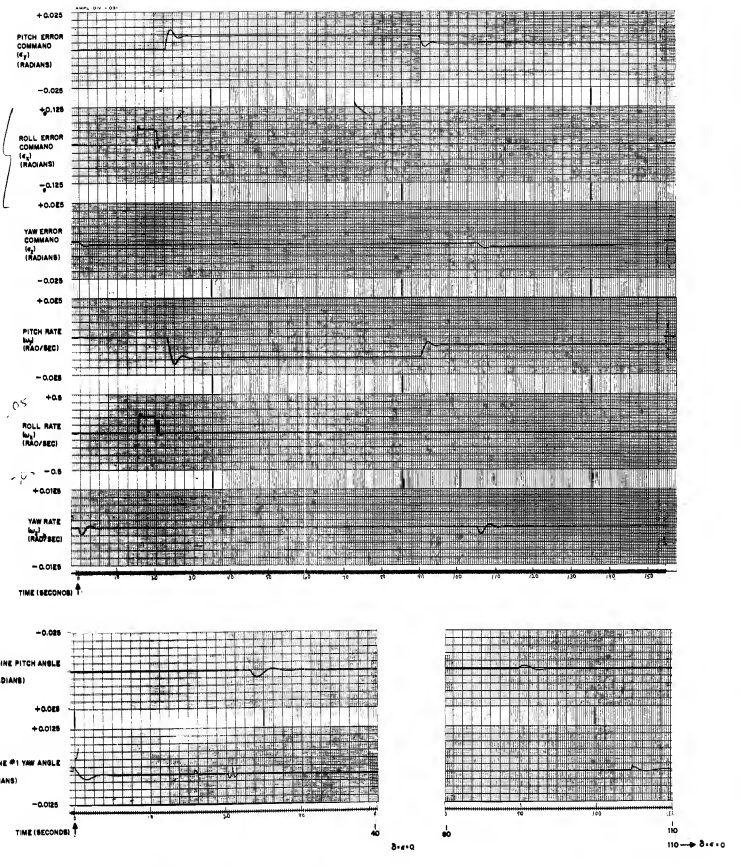


Figure III-1. Results Run A Combined Simulation, No Aerodynamic Forces

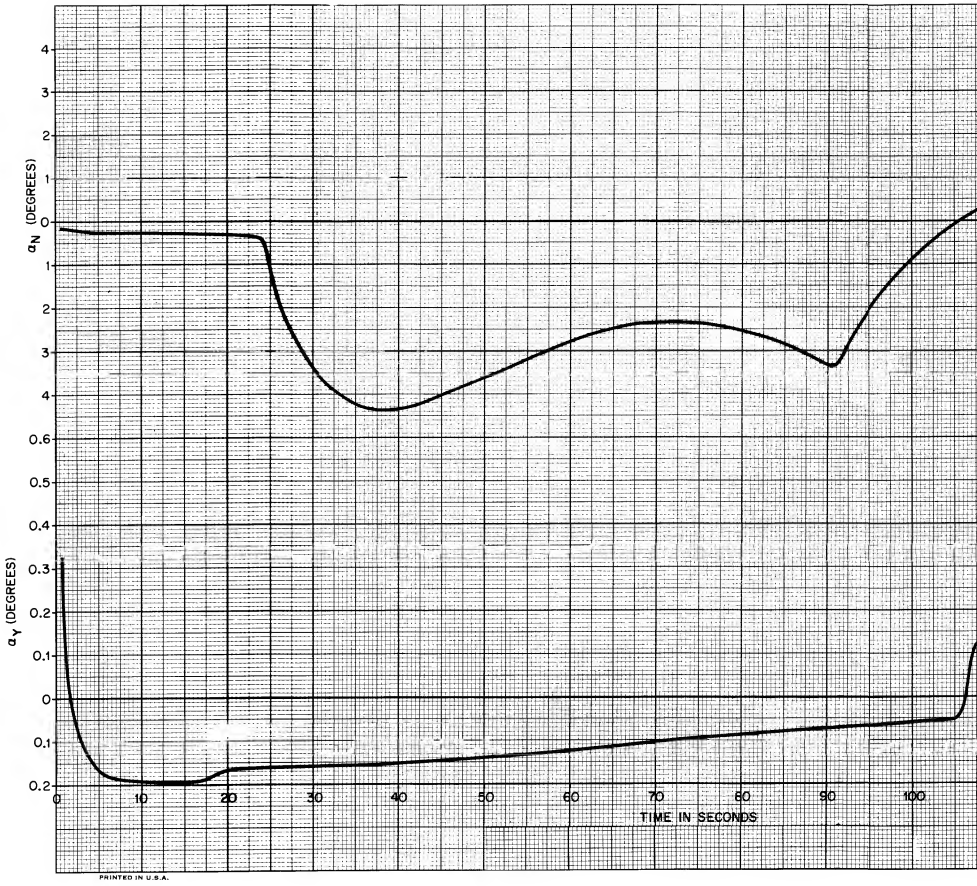
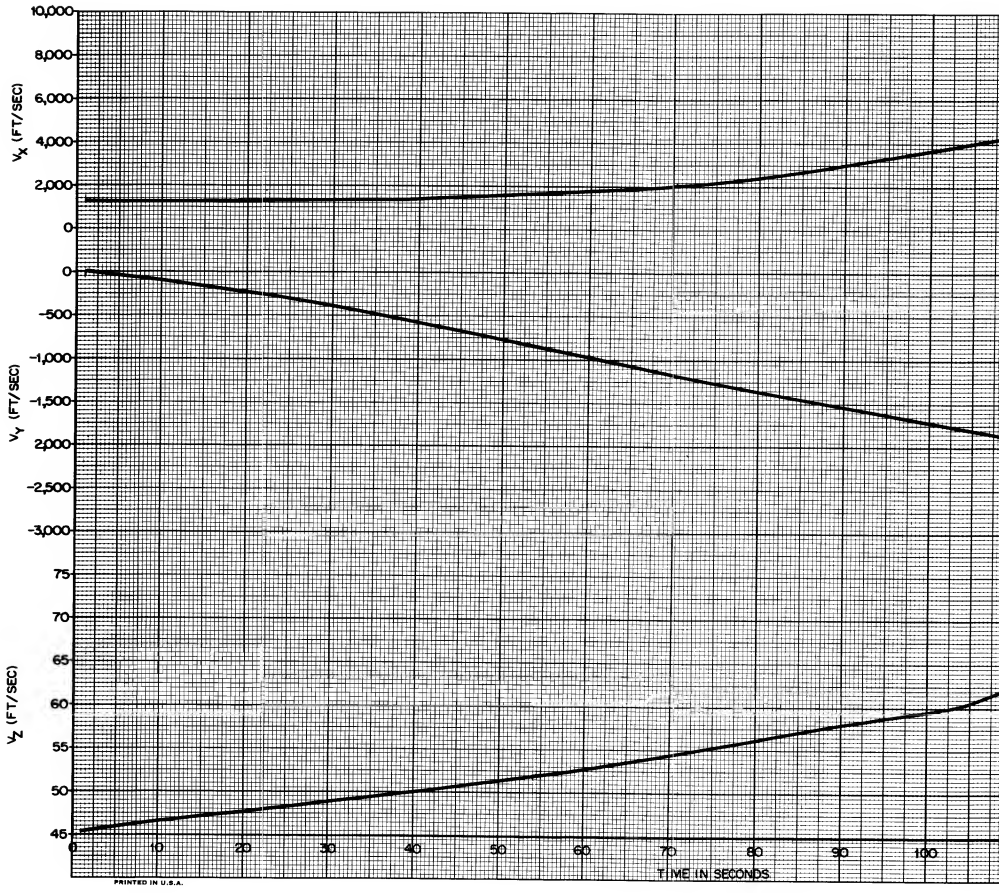


Figure III-2. Angles of Attack (α_N and α_Y) Run A


10,000
8,000
6,000
4,000
2,000
0
V_X (FT/SEC)
0
-500
-1,000
-1,500
-2,000
-2,500
-3,000
V_Y (FT/SEC)
75
70
65
60
55
50
45
V_Z (FT/SEC)
0
10
20
30
40
50
60
70
80
90
100
TIME IN SECONDS
PRINTED IN U.S.A.

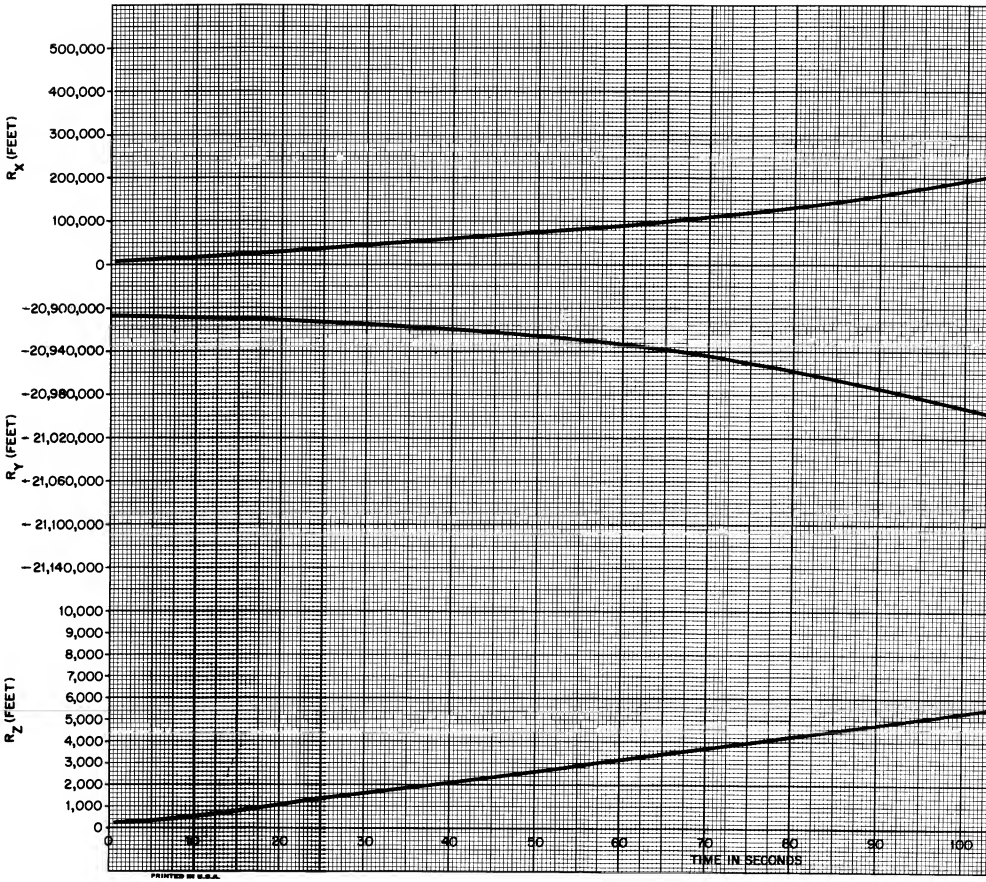
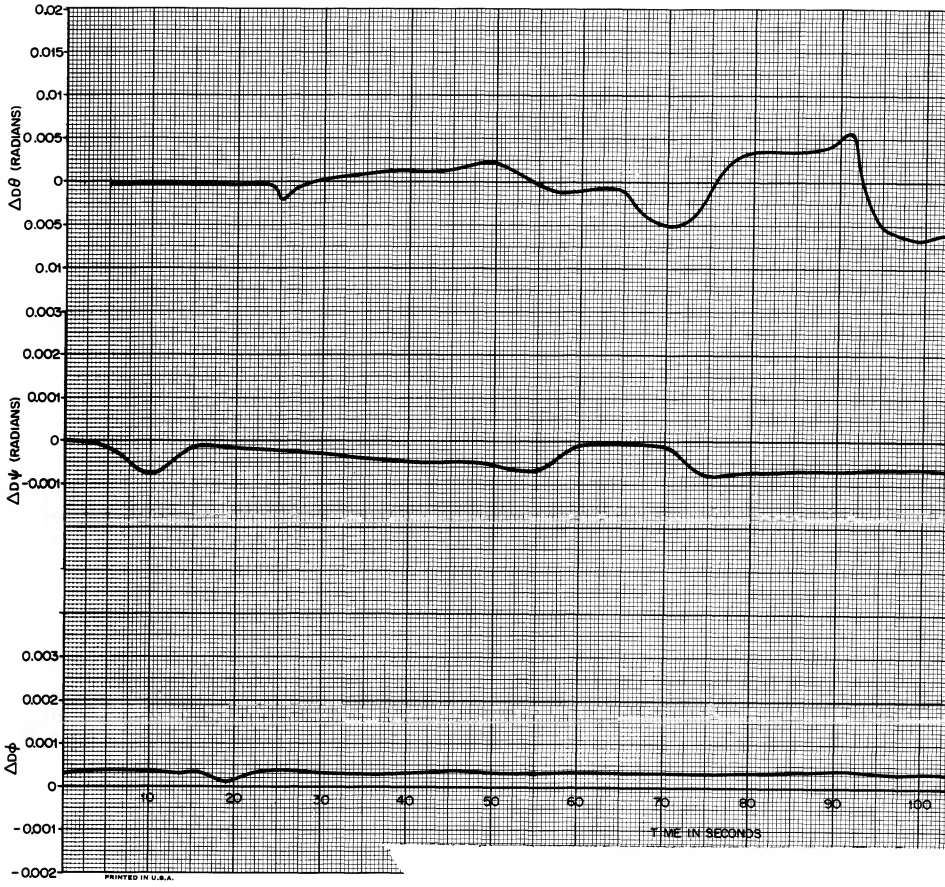

R_X (FEET)
500,000
400,000
300,000
200,000
100,000
0
R_Y (FEET)
−20,900,000
−20,940,000
−20,980,000
−21,020,000
−21,060,000
−21,100,000
−21,140,000
R_Z (FEET)
10,000
9,000
8,000
7,000
6,000
5,000
4,000
3,000
2,000
1,000
0
0
10
20
30
40
50
60
70
80
90
100
TIME IN SECONDS

Figure III


0.02
0.015
0.01
0.005
0
0.005
0.01
ΔDθ (RADIANS)
0.003
0.002
0.001
0
-0.001
ΔDψ (RADIANS)
0.003
0.002
0.001
0
-0.001
-0.002
Δdφ
10
20
30
40
50
60
70
80
90
100
TIME IN SECONDS
PRINTED IN U.S.A.

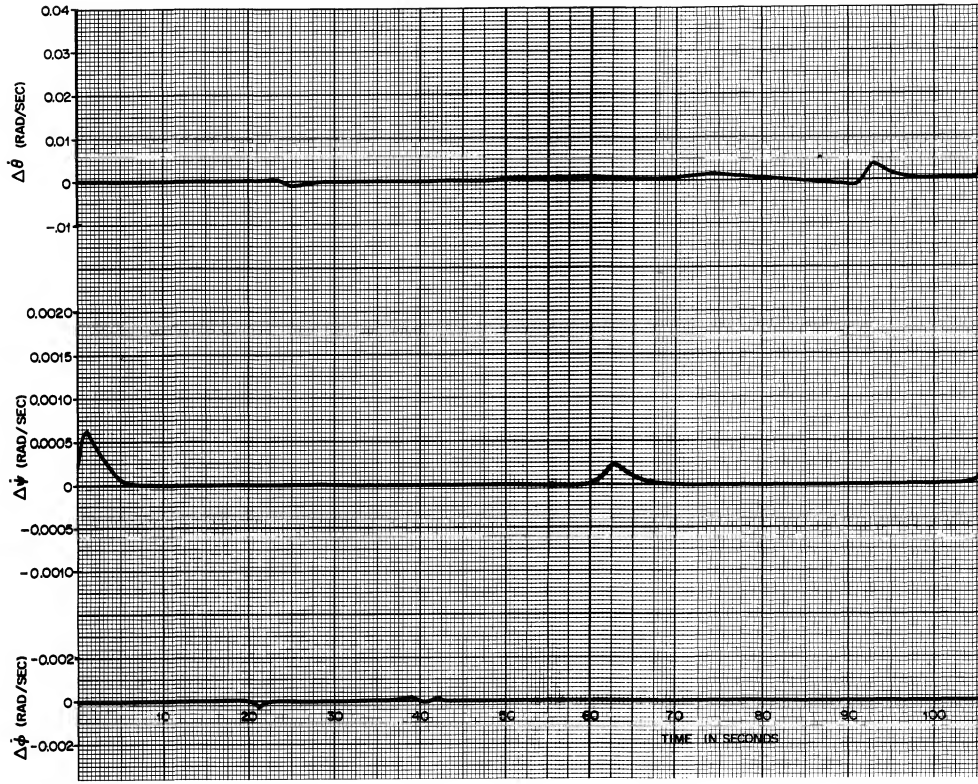
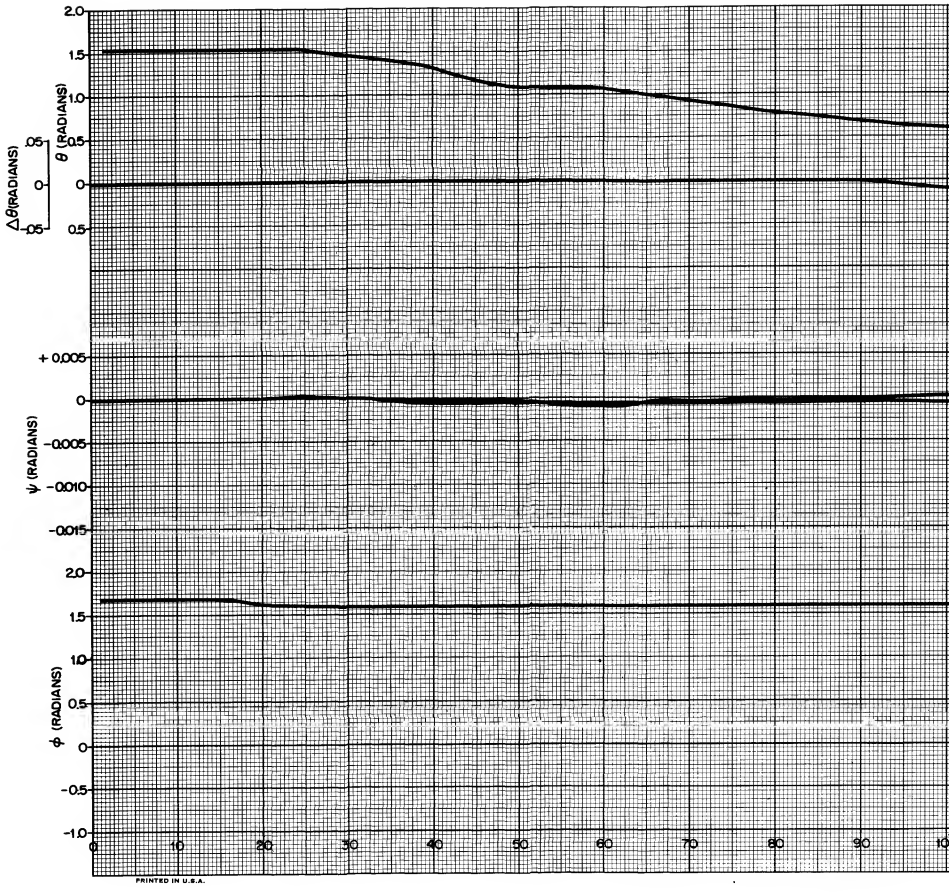


Figure III-6. Vel



PRINTED IN U.S.A.

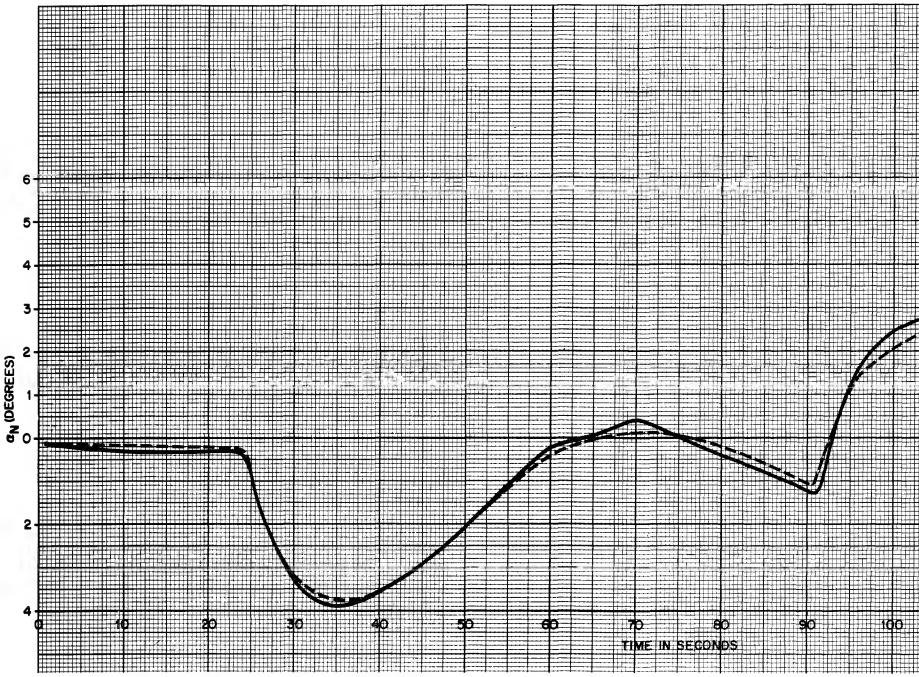
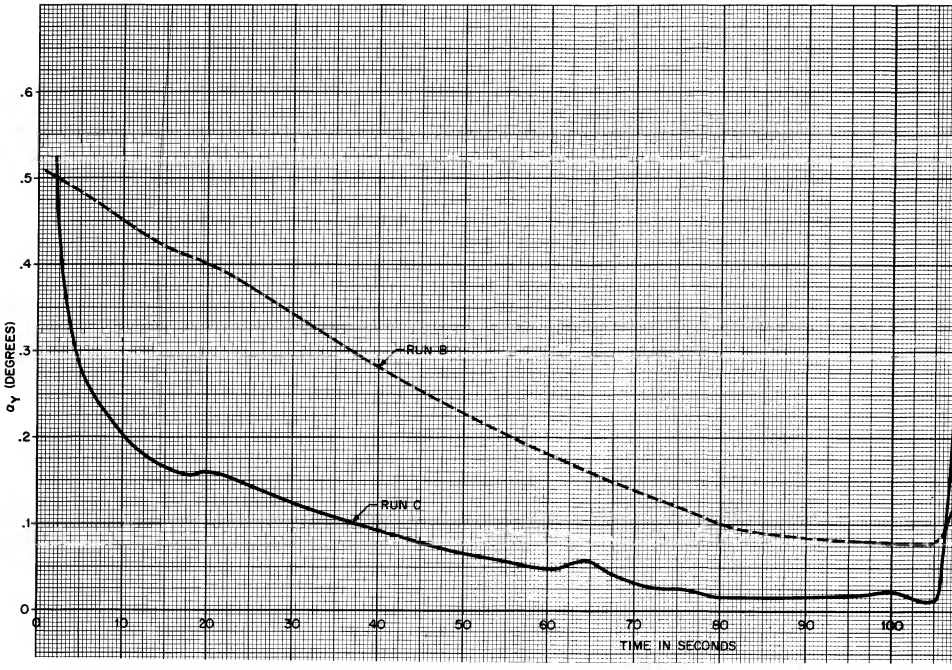


Figure III-8. Pitch P


.6
.5
.4
.3
.2
.1
0
α_Y (DEGREES)
RUN B
RUN C
0
10
20
30
40
50
60
70
80
90
100
TIME IN SECONDS

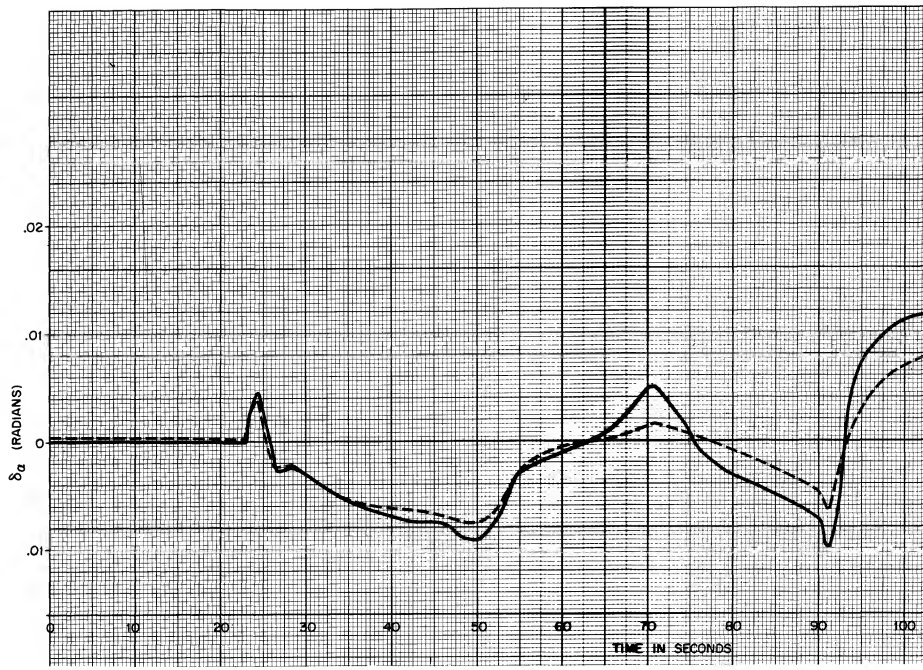
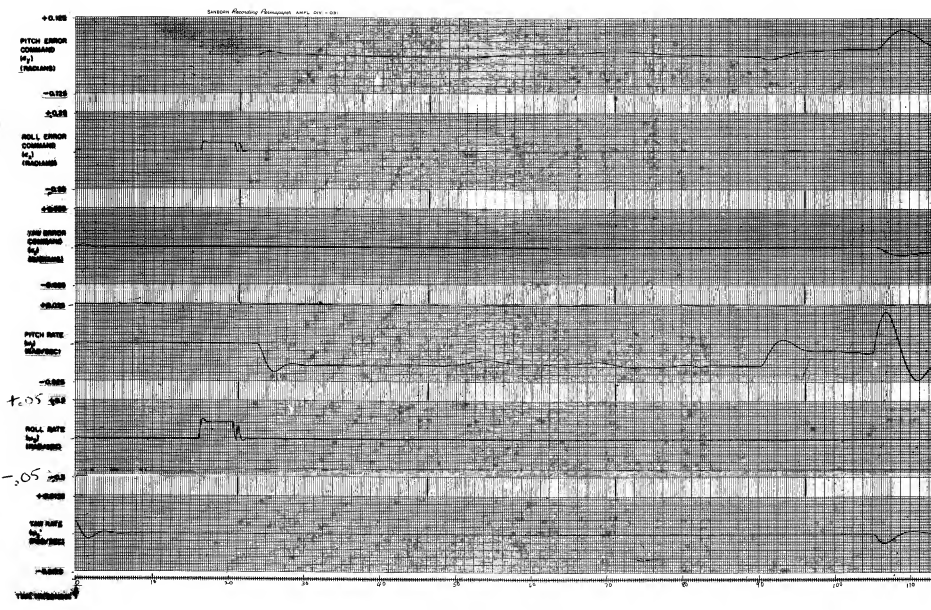


Figure III-10. Engine


PITCH ERROR COMMAND (e_p) (RADIANS)
ROLL ERROR COMMAND (e_r) (RADIANS)
YAW ERROR COMMAND (e_y) (RADIANS)
PITCH RATE (w_p) (RAD/SEC)
ROLL RATE (w_r) (RAD/SEC)
YAW RATE (w_y) (RAD/SEC)
TIME (SECONDS)

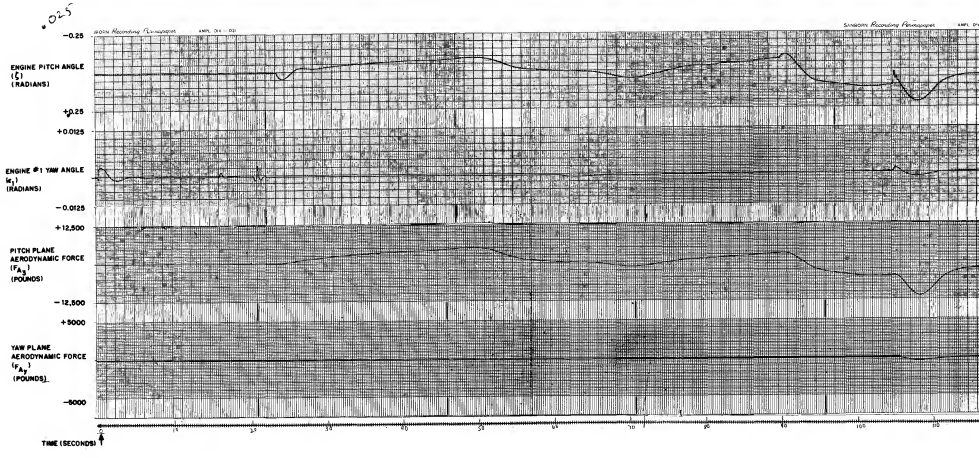
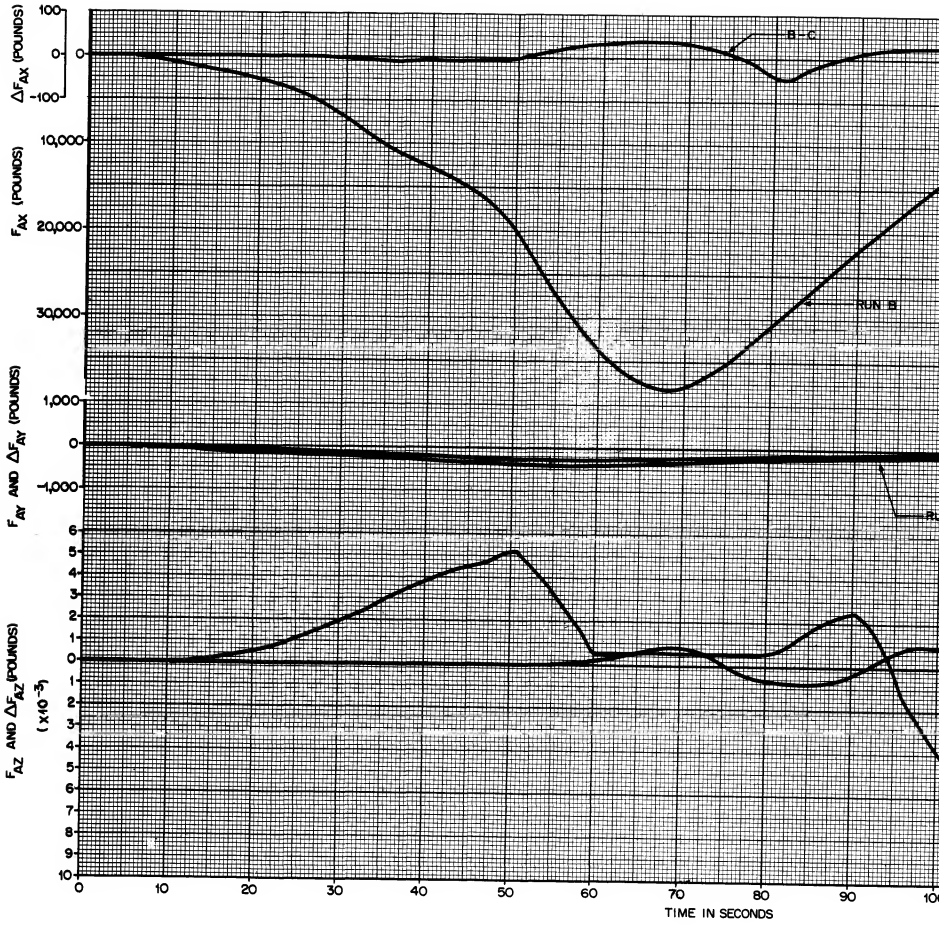


Figure III-12. E
Results


ΔF_{AX} (POUNDS)
100
0
-100
F_{AX} (POUNDS)
10,000
20,000
30,000
B - C
RUN B
F_{AY} AND ΔF_{AY} (POUNDS)
1,000
0
-1,000
F_{AZ} AND ΔF_{AZ} (POUNDS)
(x10^{-3})
6
5
4
3
2
1
0
1
2
3
4
5
6
7
8
9
10
0
10
20
30
40
50
60
70
80
90
TIME IN SECONDS

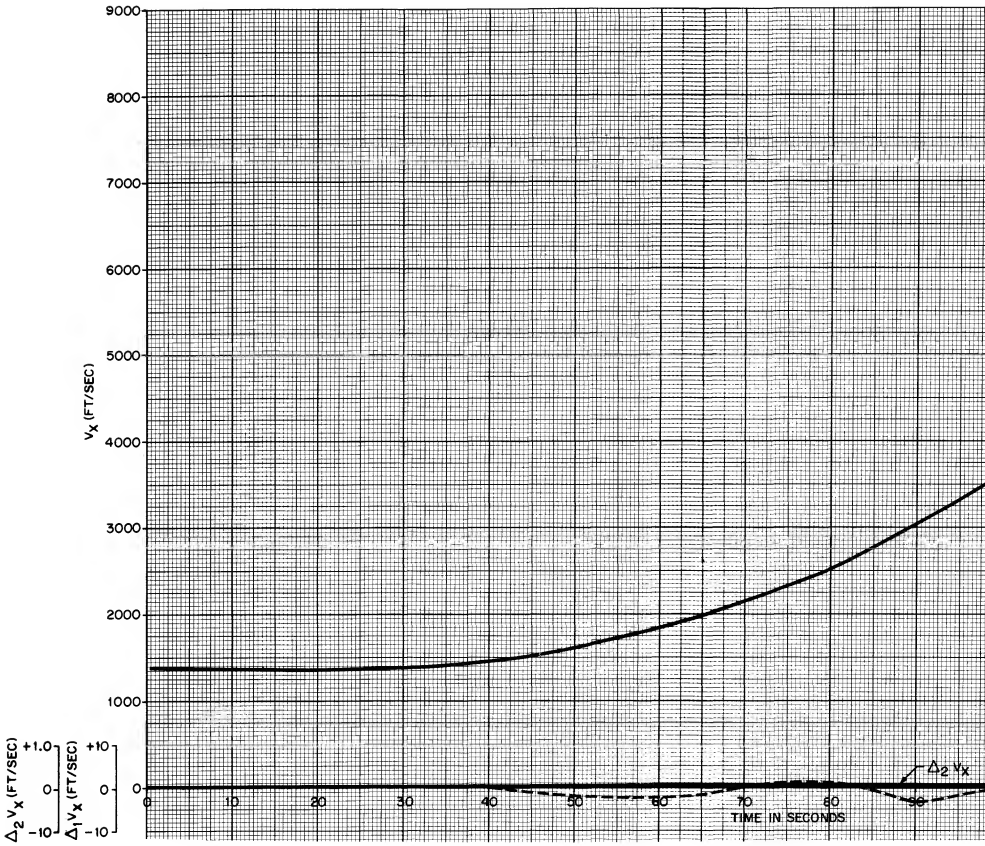
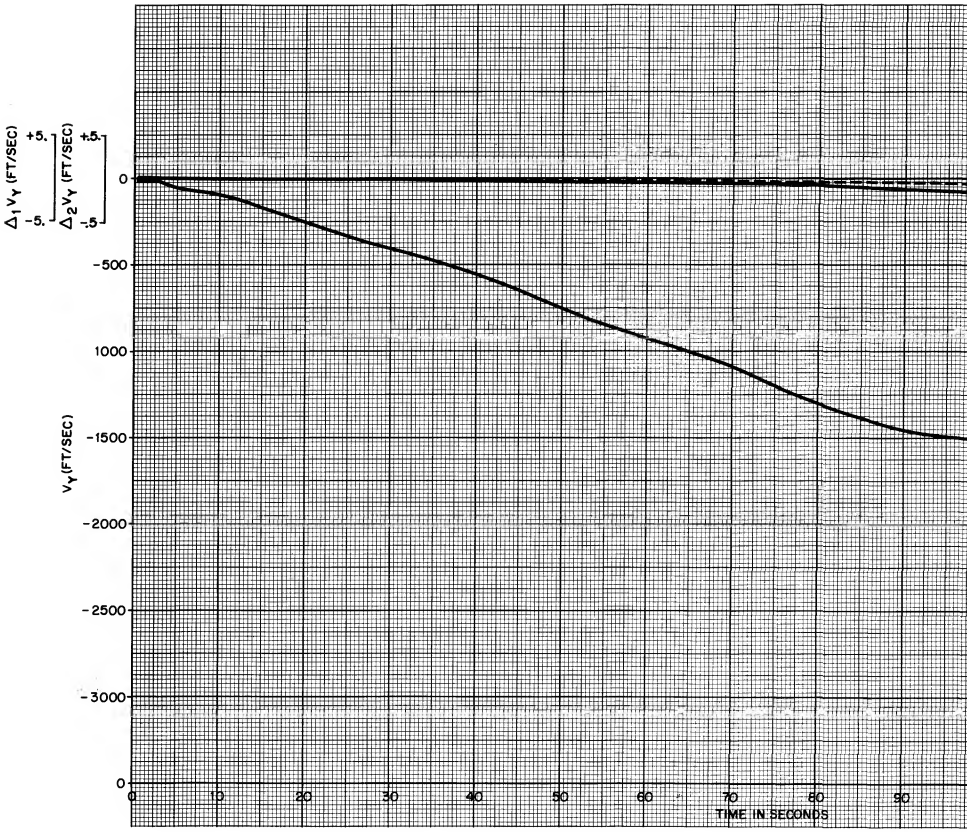


Figure III-14.


$\Delta_1 V_Y$ (FT/SEC)
+5.
-5.
$\Delta_2 V_Y$ (FT/SEC)
+5.
-5.
0
-500
1000
-1500
-2000
-2500
-3000
V_Y (FT/SEC)
0
0
10
20
30
40
50
60
70
80
90
TIME IN SECONDS

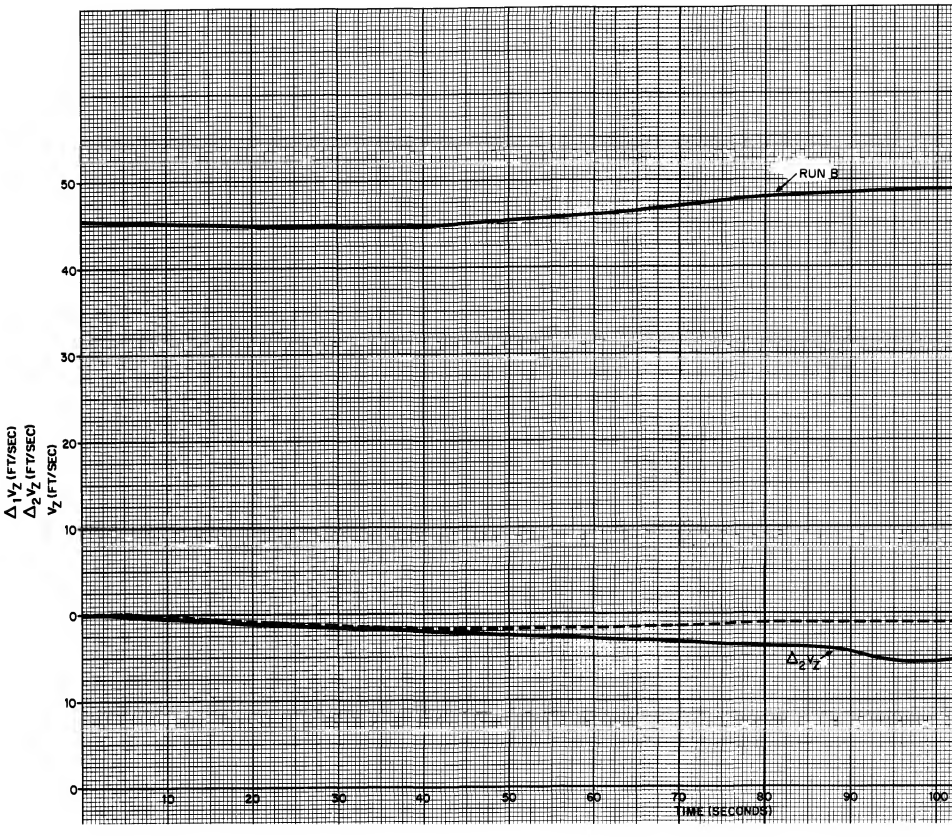


Figure III-16. La

500,000
400,000
300,000
200,000
100,000
0
0
10
20
30
40
50
60
70
80
90
TIME (SECONDS)
$\Delta_2 R_x$
$\Delta_2 R_x$ (FT)
10
0
−10
$\Delta_1 R_x$ (FT)
200
100
0
−100
R_x (FEET)
PRINTED IN U.S.A.

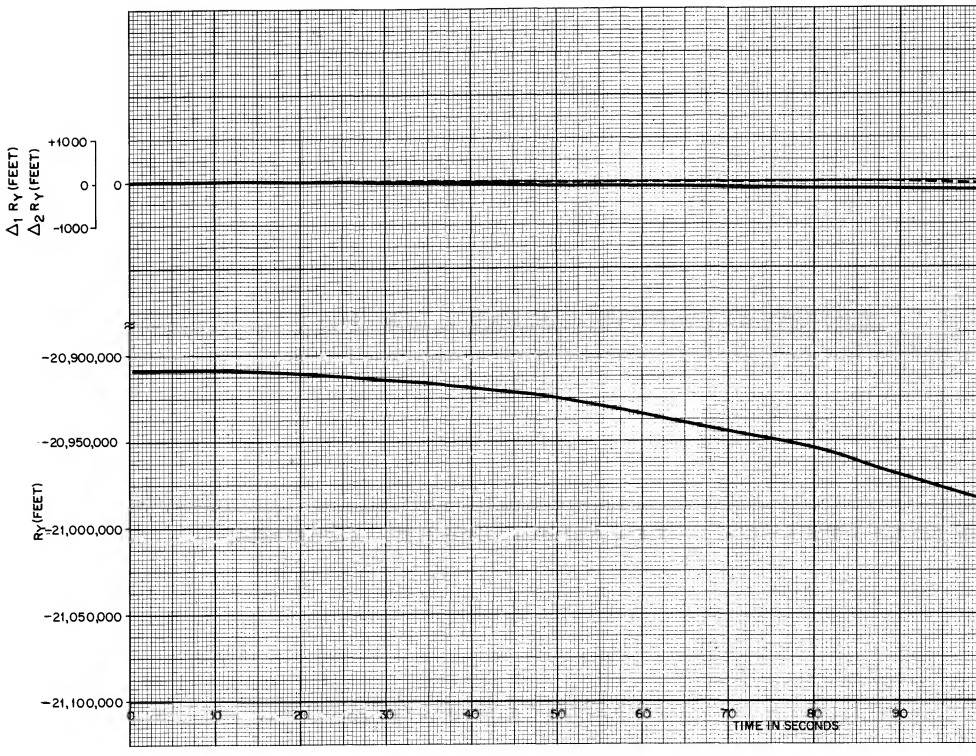
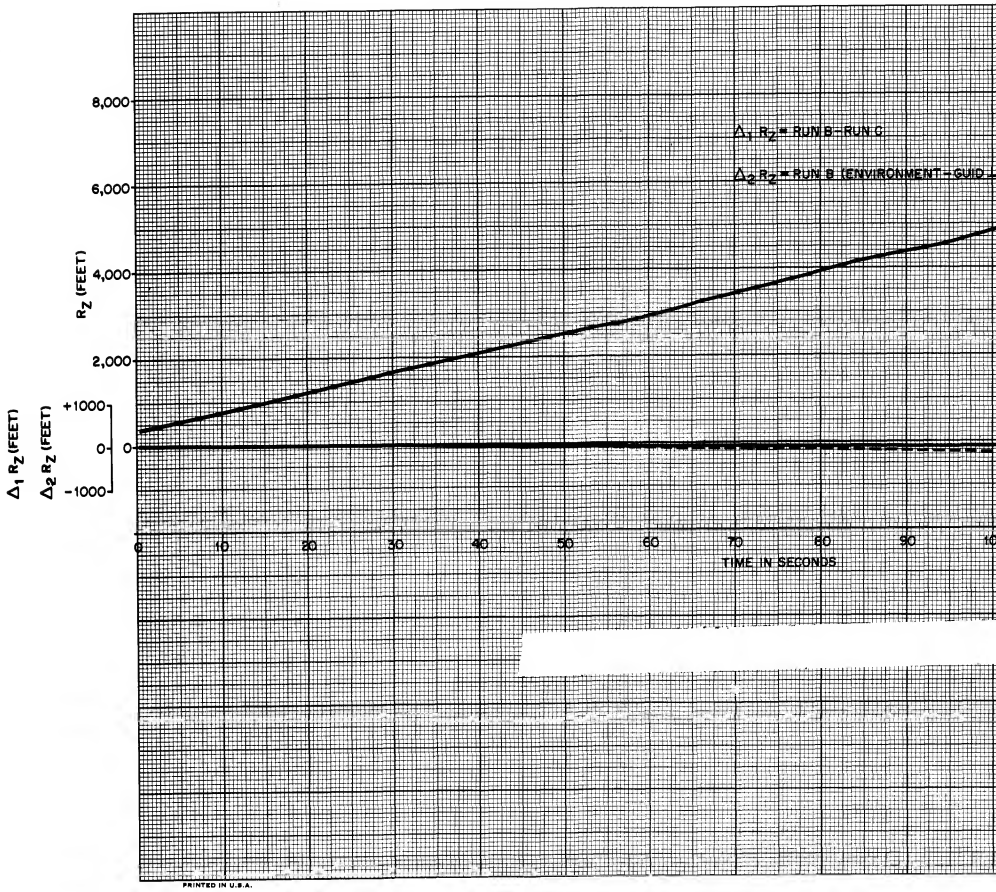


Figure III-18.



PRINTED IN U.S.A.

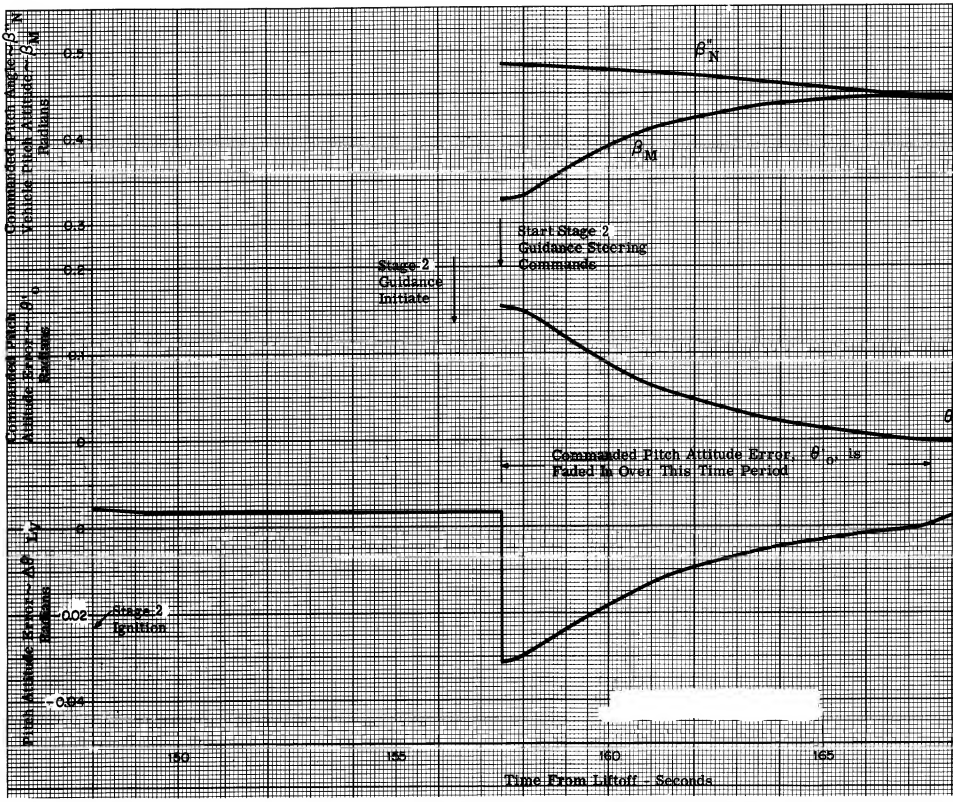
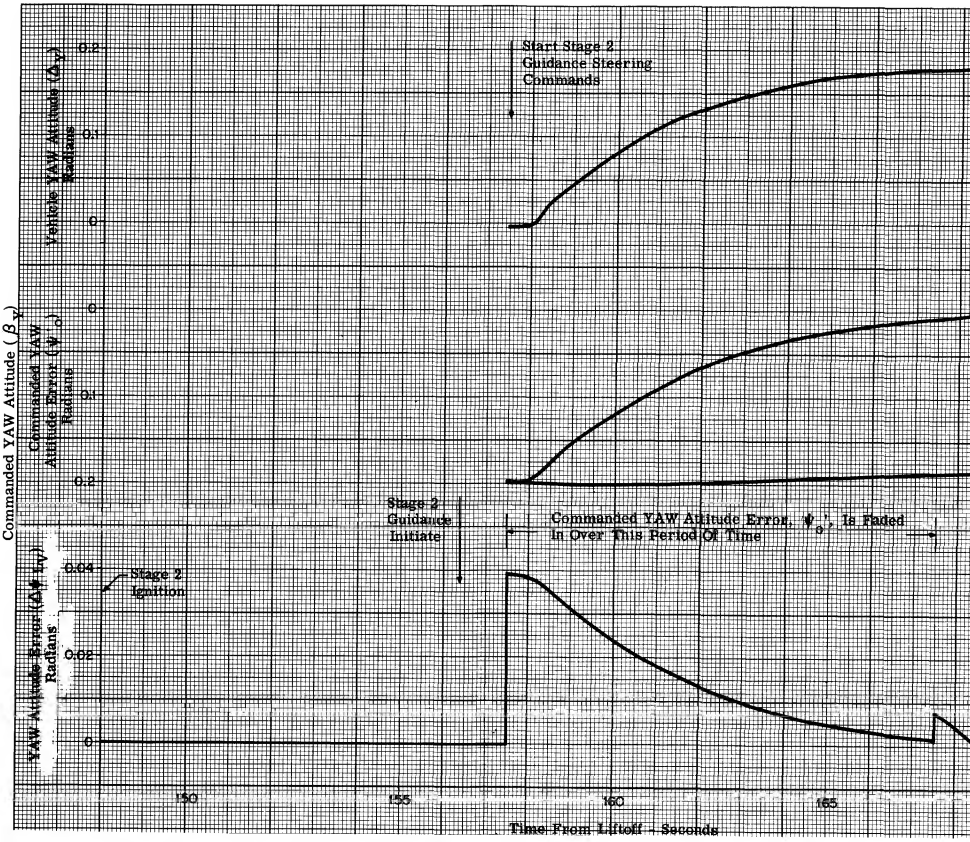


Figure III-20. Stage 2 Steering Co


Commanded YAW Attitude (β_{yc})
Vehicle YAW Attitude ($\Delta\psi_v$) Radians
0.2
0.1
0
Start Stage 2 Guidance Steering Commands
Commanded YAW Attitude Error (ψ_o') Radians
0.2
0.1
0
Stage 2 Guidance Initiate
Commanded YAW Attitude Error, ψ_o', Is Faded In Over This Period Of Time
YAW Attitude Error ($\Delta\psi_{Err}$) Radians
0.04
0.02
0
Stage 2 Ignition
150
155
160
165
Time From Liftoff - Seconds

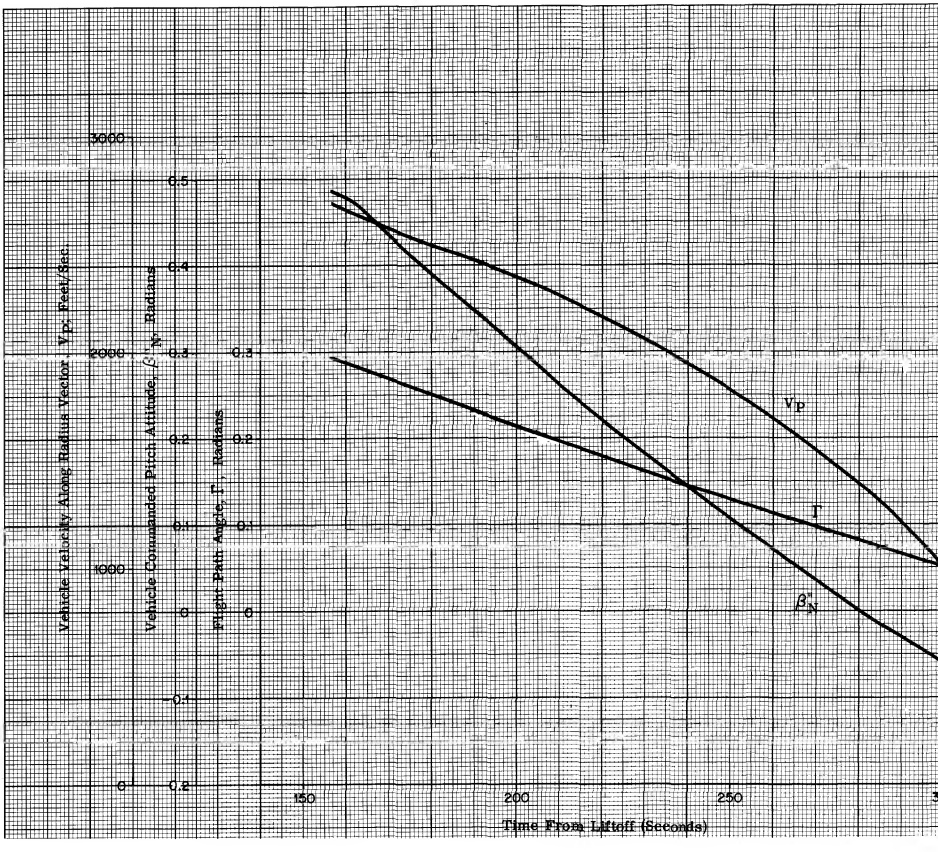
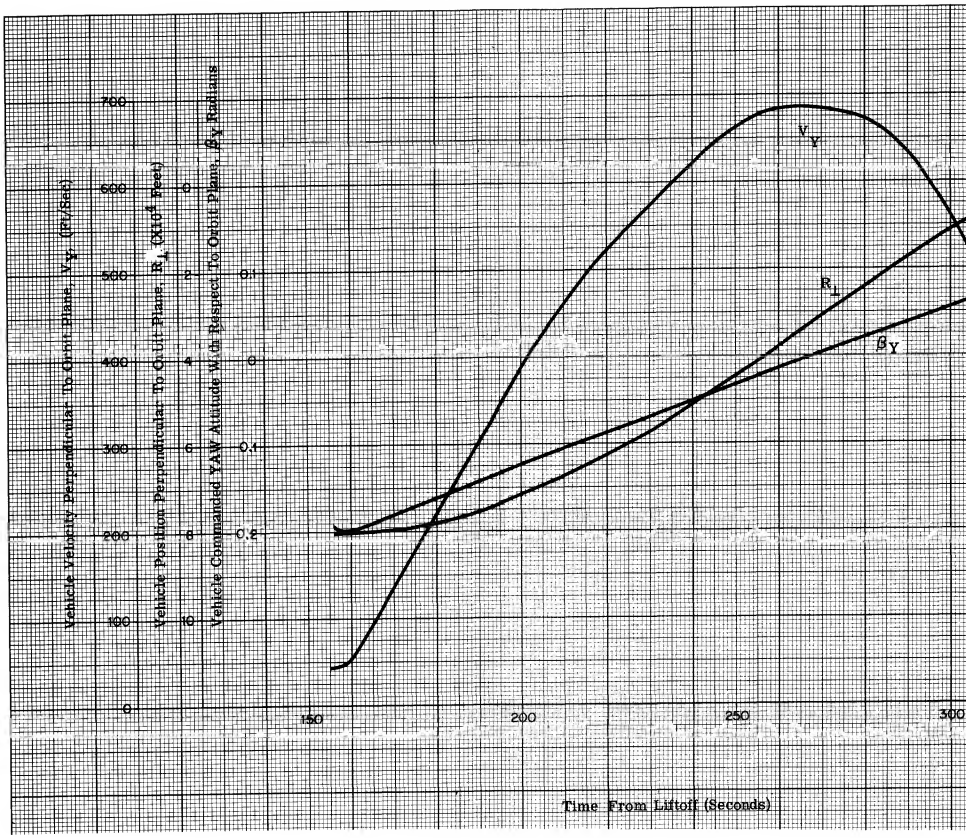


Figure III-22. Stage


Vehicle Velocity Perpendicular To Orbit Plane, V_Y (Ft/Sec)
Vehicle Position Perpendicular To Orbit Plane, $R_\perp$ ($\times 10^4$ Feet)
Vehicle Commanded YAW Attitude With Respect To Orbit Plane, β_Y Radians
700
600
500
400
300
200
100
0
8
6
4
2
0
0.8
0.1
0
0.1
V_Y
$R_\perp$
β_Y
150
200
250
300
Time From Liftoff (Seconds)

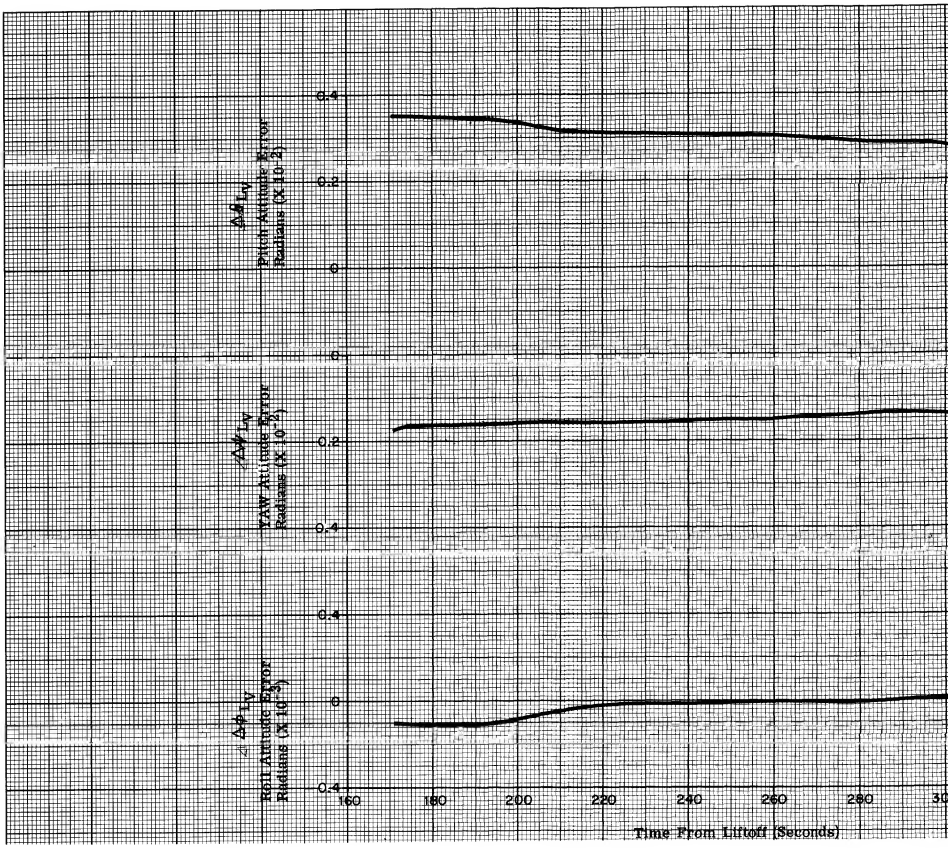


Figure III-24. Stage 2

Δθ = Dimensionless Computation Variable
Time To Go (Seconds)
T_G
Δθ
Velocity Loss Due To Gravity, V_{gL}
Velocity Loss Due To Steering, $V_{\lambda L}$
(Feet/Sec)
V_{gL}
$V_{\lambda L}$
ΔV_g = Velocity To Be Gained
W_N = Dimensionless Comp. Variable
ΔV_g
W_N
Final Velocity Loss Due To Gravity, V_{gf}
Final Velocity Loss Due To Steering, $V_{\lambda f}$
(Feet / Second)
Correction For Time To Go, δT_G
V_{gf}
$V_{\lambda f}$
Time From Liftoff (Seconds)
Time Fro

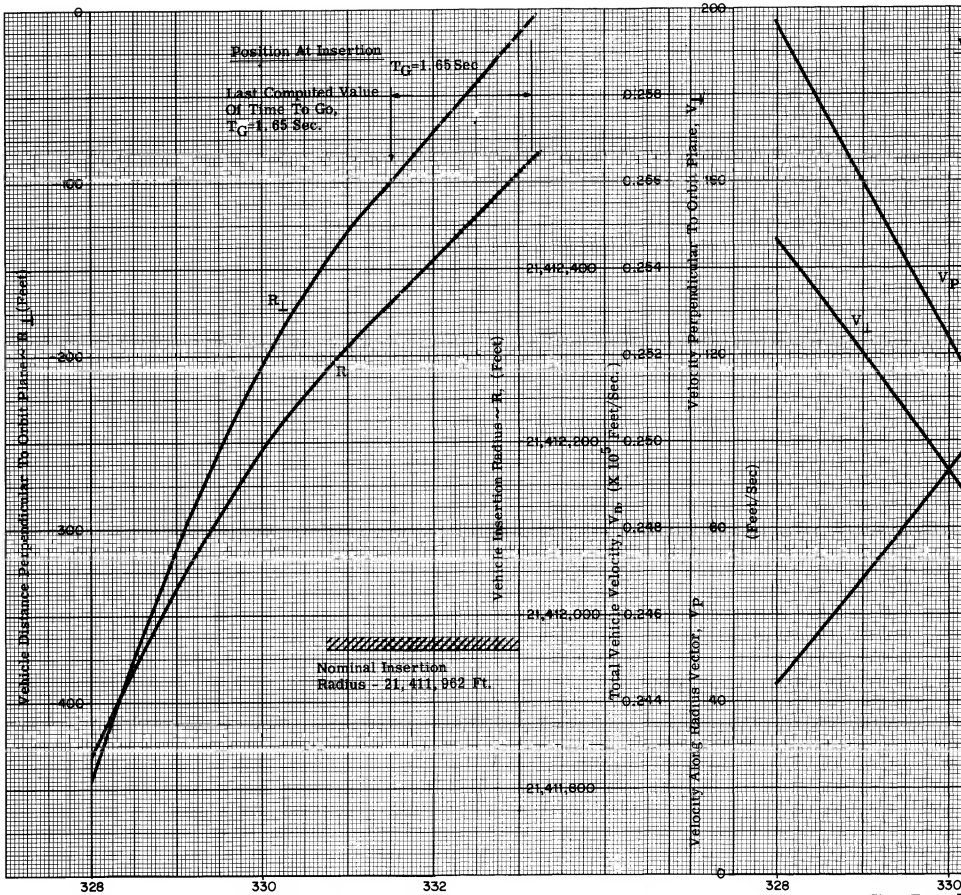
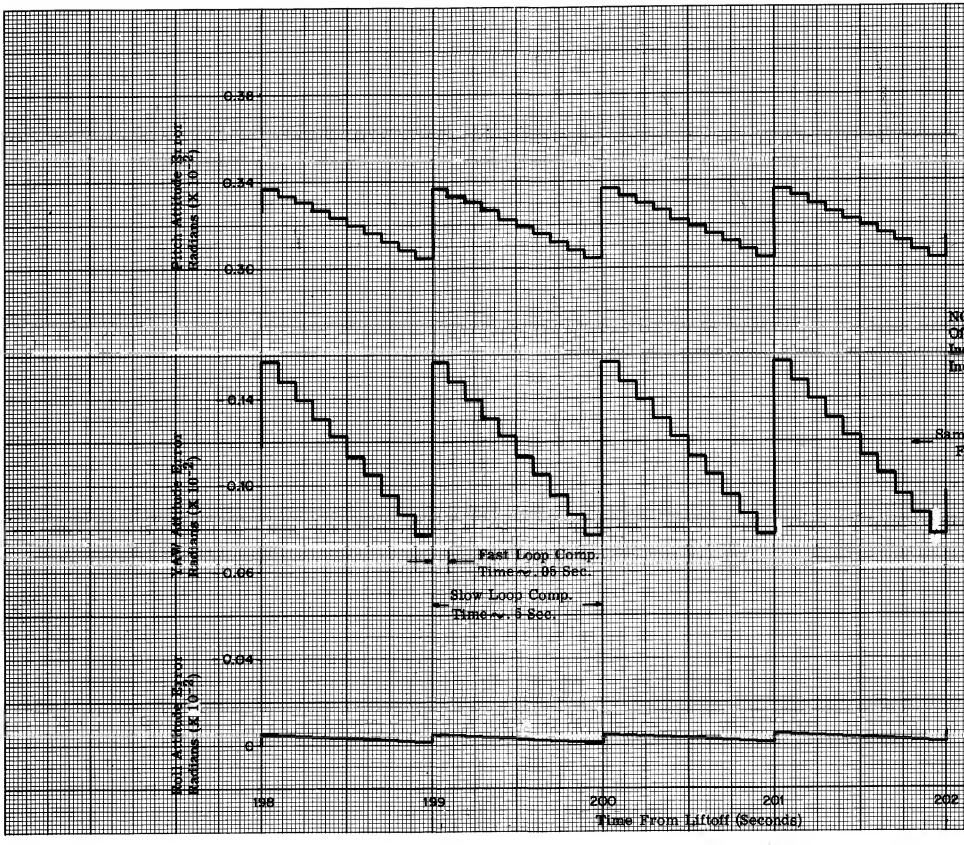


Figure III-26. Stage


0.38
0.34
0.30
Roll Attitude Error
Radians (3σ)
0.14
0.10
0.06
0.04
0
Yaw Attitude Error
Radians (3σ)
Fast Loop Comp.
Time ≈ .05 Sec.
Slow Loop Comp.
Time ≈ .5 Sec.
198
199
200
201
202
Time From Liftoff (Seconds)

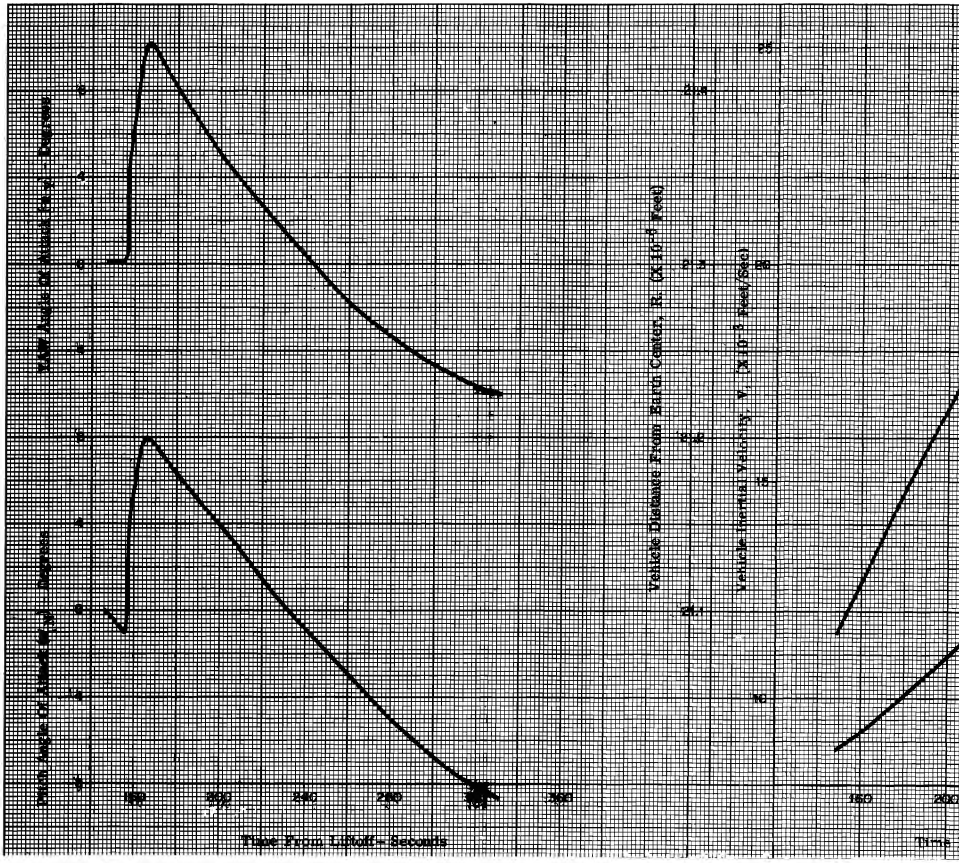


Figure III-28. Vehicle
and Radius to Earth Ce

Section IV

PLANS FOR CONTINUED SIMULATION STUDY

Section IV

PLANS FOR CONTINUED SIMULATION STUDY

A. OBJECTIVES

The primary objective of future ascent guidance studies is to obtain an over-all error model for the IGS ascent guidance system. The secondary objective is to study switch-over transients. In addition to these objectives, the various areas which required additional investigation and simulation will be considered. * Studies which are being planned are discussed in this section.

B. FIXED POINT PROGRAM

The simulation is programmed in 27 bit floating point binary. This implies that in many areas accuracy will exceed the capability of the 26-bit, fixed-point computations to be done in the actual GEMINI IGS. Errors due to this difference are difficult to assess; therefore, a fixed-point simulation of the IGS equations is now being completed. Plans are also being made to tie this system to the programs developed for this simulation to assess some of these errors.

C. EFFECTS OF UPDATES ON STAGE 2 EQUATION

The late updates during flight may tend to introduce slight discontinuities in the velocity data used in the steering equations. These effects along with the effects of poor updates on the solution to the steering equation will require evaluation. The total uncertainty in the update data will also be considered so that the accuracy of the required update data can be assessed.

D. INITIALIZATION OF GIMBAL ANGLES PRIOR TO LAUNCH

The reading of the platform gimbal angles prior to platform release will be considered along with the setting of these angles equal to the commanded angles following platform release, because the platform azimuth could be misaligned by as much as 45 min. This misalignment could result in a 45 min. attitude error as long as initial platform gimbal angles are obtained from

---

*Scheduling is not included in this discussion. Information on scheduling will be included in "Plans for Ascent Guidance Study," Feb. 1963, (to be published).

preset and computed values. Similarly, the initial commanded gimbal angles could be in error as much as 14 min. because of vehicle, spacecraft, and IMU misalignment as well as gimbal reading uncertainties in the platform. Since the resolution in gimbal reading and readout noise is much less than 14 min., accuracy can be gained by initializing as described.

E. LOADING DATA OR CONSTANTS INTO THE COMPUTER

From the operational standpoint, it may be desirable to load a number of constants into the computer within several weeks or possibly hours of a flight, e.g , R_f, V_f or capability to remove position steering computations. A thorough review of the equations and constants will be made to determine if this action is beneficial. Constants or variables requiring frequent or occasional changing will also be outlined. Procedures to be used to provide this update capability will have to be prepared should it become necessary.

F. GROUND CHECK-OUT AND SIMULATION OF ASCENT PROGRAM

During launch pad check-out, a capability must be provided to operationally exercise the ascent program. This provision may be difficult without additional instructions in the program or some sophistication in the AGE equipment. The major difficulty is in the Stage 2 equations where the nominal solution depends on an exponential thrust profile, and where the delivery of a SECO signal is required. A check-out procedure is being studied without additional instructions. However, additional instructions will be considered if required.

G. GIMBAL READING UNCERTAINTIES

Various errors can be introduced through the platform gimbal reading due to misalignment of the IMU, spacecraft, and GLV, or through the uncertainties in the gyro reading. In fact, the gimbal reading uncertainty could be as much as 10 min. A model of the gimbal reading resolution, uncertainty, and possibly noise will be introduced in the simulation to estimate these effects.

H. OUTPUT LADDER NETWORK SIMULATION

Figure III-27 indicates that the resolution of the IGS output ladder network is quite large compared to the accuracy to which the attitude errors have been computed and used in these programs. Consequently, these effects must be included in the simulation to assess their effect on vehicle stability and insertion error.

I. TIMING OF DISCRETES

Timing of the discretes in the Ascent program is generally controlled in the slow-computation loop. The accuracy of these discretes can be controlled better if approximately one-half computation cycle was subtracted from each nominal discrete time. For example, the present accuracy of a discrete is $t_{NOMINAL}$ +0.5, -0.0 sec. The above operation, however, would provide an accuracy of $t_{NOMINAL}$ ±0.25 sec.

J. DIFFERENCE IN R AND R_f AT SHUTDOWN

Reasons for the large discrepancy in R-R_f at insertion will require a more detailed study. The effects of extending position-steering computations closer to shutdown will be investigated in an attempt to improve these accuracies. Judgment, however, must be exercised because the payload penalties to do position steering late in flight could exceed the advantages gained.

K. FADE-IN OF INITIAL STAGE 2 ATTITUDE ERRORS

As previously discussed, the fade-in of initial Stage 2 attitude errors should be considered to provide a smooth transition from Stage 1 to Stage 2 flight. Advantages and disadvantages will be weighed before any decision is made.

L. PROVISION FOR STUDYING SWITCH-OVER TRANSIENT

Since the IGS is a back-up system to the primary Radio Guidance System to be used on GEMINI, it will be necessary to include the investigation of switch-over transient in the simulation. The following methods are being considered for this investigation:

1. Include radio guidance system in the simulation which will allow switch-over at any time in the trajectory.

2. Include capability in the present program which will allow entry into the ascent profile at any time along the trajectory with selected initial conditions.

M. ATTITUDE ERROR QUANTIZING

The attitude errors supplied to the autopilot from the guidance equations in this simulation are accurate to eight significant digits. In the actual GEMINI computer, these outputs will be supplied on the output ladder network where they are quantized to 0.12 deg. The effects of this quantization are discussed under Stage 2 results in Section III-C. This quantization will be incorporated in future computer runs to determine its effect on system stability.

N. ORBIT VELOCITY ADJUST

Simulation efforts to date have not included the orbit velocity adjust equations which are a part of the operational math flow (Figure E-1B). Two plans are being considered for the simulation of these equations.

The first plan is to use the present simulation and approximate the control system with a perfect autopilot. This might be pictured as a third Stage in the GEMINI ascent. An approach of this type would certainly provide an indication of the adequacy of the equations; however, parametric analysis would be somewhat difficult.

The second approach would use an entirely separate program. A forcing function would be provided and a perfect autopilot assumed. This would provide maximum flexibility in the study of these equations.

O. PROGRAM COMPATIBILITY PLANS

Specific items will be modified to reach a state of compatibility with other GEMINI studies regarding data used in the program. These changes will facilitate comparisons of results which might be available through the Martin, Aerospace, and McDonnell simulation efforts. The items are as follows:

1. Write and substitute into the program a subroutine describing gravitational acceleration according to approved GEMINI models.

2. Write and substitute into the program a subroutine describing atmospheric coefficients, based on NASA TND-595 "A Reference Atmosphere for Patrick AFB." At this time surface winds will also be introduced.

3. Update vehicle characteristics to be compatible with document LV-140. Specific characteristics are aerodynamic coefficients and location of center of pressure.

4. Provide for inclusion of engine pitch angle offset.

5. Revise weight, thrust, and mass flow rates to latest compatibility data.

6. Provide correction to account for initial orientation of the platform with respect to the orbit plane as described in Appendix A.

P. MATH FLOW CHANGES

The following list outlines areas in the math flow where changes have been incorporated or are currently being planned:

1. Add Third Pitch Step
2. Add Low-Level Sensor Discrete
3. Modify Test on ΔQ (A205)
4. Change Standard Update Increment to 40 sec. (A128)
5. Move platform release to engine ignition and delete ephemeris calculations.
6. Set commanded platform gimbal angles equal to actual platform gimbal angles prior to platform release
7. Modify Orbit Insertion Equations
8. Modify SECO sequence.

These changes will be added to the simulation as they are incorporated in the operational computer. Additional details concerning the above changes will be contained in the math flow change records as they appear.

Appendix A

EXPLANATION AND DERIVATION OF GEMINI ASCENT GUIDANCE SIMULATION

Appendix A

EXPLANATION AND DERIVATION OF GEMINI ASCENT GUIDANCE SIMULATION

I INTRODUCTION

This appendix defines the composite approach used initially in IBM studies on GEMINI ascent guidance simulation. The symbols and mathematics will not necessarily follow exactly in all programs and subroutines, but each interface will be compatible.

The task of writing the main environmental program was simplified by using existing vehicle guidance simulation programs where possible. However, coordinate systems and symbols were not always identical. Since the specific areas in which the differences exist are relatively unimportant, most of the differences will not be identified. One exception is the case of the coordinate systems chosen for programming.

Certain simplifications have been made in preparing this simulation. These can be classified under one of two categories. One category covers the use of existing data rather than the rewriting of subroutines. The other covers simplifications which were made by neglecting elements of the problem that were not under investigation or would not contribute to the final results.

The first category, which covers gravitational computations, determination of atmospheric coefficients, corrections for initial orientation of the platform, use of constant thrust, simplification of platform orientation and omission of zero offset of engine gimbal, are discussed in this report.

In the second category the following simplifications were made:

- The vehicle was considered to be a rigid body with no bending or fuel slosh modes.
- All body rates and angles were assumed to be measured around the center of gravity.
- Body rotational rates were considered constant during one computation cycle.

- Vehicle position and velocity were approximated by a quadratic for one complete cycle.
- Only first-order aerodynamic forces and moments were considered.
- Instrumentation errors were not considered.

II LIST OF SYMBOLS

A. COORDINATE SYSTEMS

X_E, Y_E, Z_E - Earth referenced frame (inertial)

I_X^*, I_Y^*, I_Z^* - Orbit plane reference frame (inertial)

I_X, I_Y, I_Z - Platform reference frame (inertial)

I_{xb}, I_{yb}, I_{zb} - Spacecraft referenced frame

I_x, I_y, I_z - Vehicle reference frame

B. MATRICES

$[a_{ij}]$ - Earth to platform transformation matrix

$[b_{ij}]$ - Platform to spacecraft transformation matrix

C. ALPHABETIC

t - Time

t_{pr} - Time of platform release

i - Inclination of desired orbit plane

C_o - Magnitude of eastward inertial velocity at launch point due to earths rotation

C_5 - Angle from east to the spacecraft I_{zb} axis

X_s - Station number of engine gimbal

Z_1, Z_2 - Distance from vehicle centerline to engines A and B respectively

X_{cp} - Station of the center of pressure of the vehicle

X_{cg} - Station of the center of gravity of the vehicle

W - Vehicle weight

M - Vehicle mass

q - Aerodynamic pressure

S_{π} - Vehicle frontal area

C_S - Speed of sound

$\mathcal{M}$ - Mach number

I_x, I_y, I_z - Moments of inertia about I_x, I_y, and I_z respectively (note that distinction in symbols must be made by reader where necessary)

G_o - Value of gravitational acceleration for conversion of fuel weight to mass

C_A - Axial force coefficient

C_Y - Side force coefficient

C_N - Normal force coefficient

F - Force

subscripts		
	X, Y, Z	- Along platform axes
	x, y, z	- Along vehicle axes
	xb, yb, zb	- Along spacecraft axes
	A_x, A_y, A_z	- Aerodynamic in vehicle frame
	A_{xb}, A_{yb}, A_{zb}	- Aerodynamic in spacecraft frame

R - Position vector

subscripts		
	X, Y, Z	- In platform frame
	X_E, Y_E, Z_E	- In earth-referenced frame

V - Velocity

subscripts

X, Y, Z - Platform frame

X_E, Y_E, Z_E - Earth referenced frame

w_X, w_Y, w_Z - Of wind in platform frame

w_x, w_y, w_z - Of wind in vehicle frame

w_{xb}, w_{yb}, w_{zb} - Of wind in spacecraft frame

D. GREEK LETTERS

θ - Platform pitch angle

ψ - Platform yaw angle

ϕ - Platform roll angle

λ - Launch point latitude

ϕ' - Launch point longitude

Ω - Longitude of the ascending node of desired orbit plane

δ - Angle between local geocentric vertical and orbit plane

γ - Angle from east to platform I_X axis

ϕ_o - Initial roll angle of platform

τ - Integration time period

ϵ_1, ϵ_2 - Vehicle engine yaw angles

δ_1, δ_2 - Vehicle engine pitch angles

ω_e - Earth's rotational rate

$\omega_x, \omega_y, \omega_z$ - Vehicle rotational rates

$\omega_{xb}, \omega_{yb}, \omega_{zb}$ - Spacecraft rotational rates

α_N - Pitch angle of attack

α_Y - Yaw angle of attack

ρ - Air density

III COORDINATE SYSTEMS

A. SYSTEM $\overline{X}_e$, $\overline{Y}_e$, $\overline{Z}_e$

This system (see Figure A-1) is an inertial reference with the $\overline{Z}_e$ axis through the north polar axis. The $\overline{X}_e$ axis is determined by the position of the Greenwich Meridian at the time t_{pr}. $\overline{Y}_e$ forms a right-handed orthogonal set.

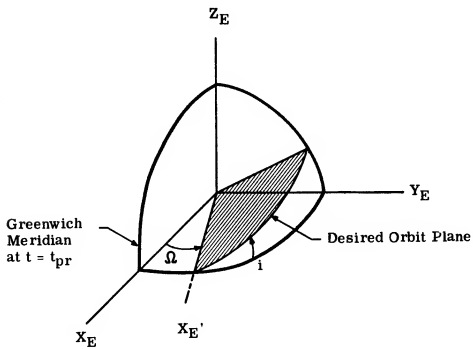


Figure A-1

Data in this system is required to initially determine and continuously provide geographic latitude which is equal to inertial latitude. Geographic longitude does not have to be determined because the earth is considered to be symmetrical about the equatorial plane. However, it could be determined by rotating the system at earth's rate.

The following angles are measured in this system:

- Ω, the longitude of the ascending node of the desired orbit plane at time $t = t_{pr}$. For positive angles, Ω is measured as a right-handed rotation about $\overline{Z}_e$.

- i, the inclination angle of the desired orbit plane. For positive angles, i is measured as a right handed rotation about the displaced $\overline{X}_e$ axis.

B. ORBIT PLANE SYSTEM $\overline{I}_X^*$, $\overline{I}_Y^*$, $\overline{I}_Z^*$

This system is defined primarily by the normal to the orbit plane, $\overline{I}_Z^*$. $\overline{I}_X^*$ is parallel to $\overline{I}_X$ of the platform reference frame, and $\overline{I}_Y^*$ forms a right-handed orthogonal set.

Figure A-1 shows the relationship between system $\overline{X}_E$, $\overline{Y}_E$, $\overline{Z}_E$ and the orbit plane.

C. PLATFORM REFERENCE SYSTEM $\overline{I}_X$, $\overline{I}_Y$, $\overline{I}_Z$

This system may not be the true platform reference. The frame is inertial after launch; but, due to platform drift, the actual platform which is also nominally inertial after launch may not coincide with this system. For the purposes of this simulation platform, drifts and initial misalignments will be ignored.

This system is defined with the $\overline{I}_X$ axis parallel to the desired orbit plane and horizontal with respect to earth at time of launch. The $\overline{I}_Y$ axis is along the geocentric radius vector towards the center of the earth. $\overline{I}_Z$ forms a right-handed orthogonal system. In practice, this system will be aligned to the local gravity vector. There is a basic simplification in this definition; see Section IV for explanation.

D. SPACECRAFT REFERENCE SYSTEM $\overline{I}_{xb}$, $\overline{I}_{yb}$, $\overline{I}_{zb}$

This system is defined as the x_b, y_b, z_b system used in the GEMINI capsule reference. $\overline{I}_{xb}$ is aligned with the longitudinal axis of the spacecraft; the positive direction is towards the nose of the capsule. $\overline{I}_{zb}$ is aligned to the vehicle yaw axis; the positive direction is down when referenced to the capsule during normal orbital flight attitude. $\overline{I}_{yb}$ is along the vehicle transverse axis and forms a right-handed orthogonal system.

E. LAUNCH VEHICLE SYSTEM $\bar{I}_x$, $\bar{I}_y$, $\bar{I}_z$

The capsule is mounted on the launch vehicle so there is a 90 deg. rotation about the longitudinal axis of the capsule, $\bar{I}_{xb}$. Thus, $\bar{I}_{xb} = \bar{I}_x$, and $\bar{I}_{yb}$ and $\bar{I}_{zb}$ are rotated 90 deg. from $\bar{I}_y$ and $\bar{I}_z$, respectively. This is a principal axis system located at the vehicle's center of gravity.

For the remainder of this Appendix, vector notation will be omitted although implied.

IV COORDINATE TRANSFORMATIONS

Throughout the simulation, two coordinate transformation matrices, $[a_{ij}]$ and $[b_{ij}]$, will be used extensively. The matrix $[a_{ij}]$ defines the relationship between the X_e, Y_e, Z_e system and the I_X, I_Y, I_Z system. The matrix $[b_{ij}]$ defines the relationship between the I_{xb}, I_{yb}, I_{zb} system and the I_X, I_Y, I_Z system. Thus, both earth and body angles are known with respect to the platform. As previously noted, both I_X, I_Y, I_Z and X_e, Y_e, Z_e are inertially fixed; therefore, $[a_{ij}]$ is a constant. Since the derivation of $[a_{ij}]$ leads to other subjects, it is convenient to discuss $[b_{ij}]$ first.

A. DERIVATION OF MATRIX $[b_{ij}]$

The matrix $[b_{ij}]$ is the transformation from capsule coordinates to platform coordinates.

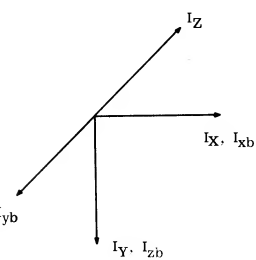


Figure A-2

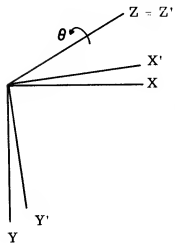
Given: a) Two right-handed coordinate systems I_{xb}, I_{yb}, I_{zb} and I_X, I_Y, I_Z aligned in a zeroed position (see Figure A-2).

b) θ, the angle between platform and innermost gimbal (1st platform rotation).

c) ψ, the angle between inner gimbal and outer gimbal (2nd platform rotation).

d) ϕ, the angle between outer gimbal and body (3rd platform rotation).

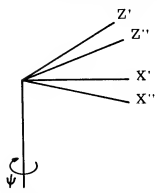
1. First Platform Rotation - About Z Through θ



$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & -\cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

θ is positive with X' above XZ plane.

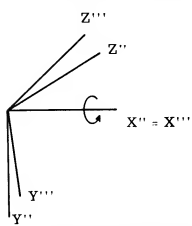
2. Second Platform Rotation - About Y'' Through ψ



$$\begin{bmatrix} X'' \\ Y'' \\ Z'' \end{bmatrix} = \begin{bmatrix} \cos\psi & 0 & -\sin\psi \\ 0 & 1 & 0 \\ \sin\psi & 0 & \cos\psi \end{bmatrix} \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix}$$

ψ is positive with X'' to the right of X'Y' plane.

3. Third Platform Rotation - About X'' Through ϕ



$$\begin{bmatrix} X''' \\ Y''' \\ Z''' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} X'' \\ Y'' \\ Z'' \end{bmatrix}$$

ϕ is positive with Z''' about X'' Z'' plane.

Now,

$$X''' = I_{xb} \qquad X = I_X$$

$$Y''' = I_{zb} \qquad Y = I_Y$$

$$Z''' = -I_{yb} \qquad Z = I_Z ,$$

and,

$$\begin{bmatrix} I_{xb} \\ I_{yb} \\ I_{zb} \end{bmatrix} = \begin{bmatrix} \cos\psi\cos\theta & -\cos\psi\sin\theta & -\sin\psi \\ -\cos\theta\cos\phi\sin\psi+\sin\phi\sin\theta & \cos\phi\sin\psi\sin\theta+\sin\phi\cos\theta & -\cos\phi\cos\psi \\ \sin\phi\sin\psi\cos\theta+\sin\theta\cos\phi & -\sin\theta\sin\phi\sin\psi+\cos\phi\cos\theta & \cos\psi\sin\phi \end{bmatrix} \begin{bmatrix} I_X \\ I_Y \\ I_Z \end{bmatrix}$$

B. RELATIONSHIP BETWEEN I_{xb}, I_{yb}, I_{zb} and I_x, I_y, I_z

Figure A-3 shows the relationship of the capsule and launch vehicle systems. This configuration results from having the astronaut's head pointed towards the south with I_{yb} nominally eastward. During the pitching maneuver, vehicle pitch is negative around I_y.

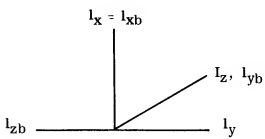


Figure A-3

The following equations, based on Figure A-3, will be frequently employed throughout the report when relating launch vehicle and spacecraft quantities:

$$I_x = I_{xb}$$

$$I_y = -I_{zb}$$

$$I_z = I_{yb}$$

C. DERIVATION OF MATRIX $\left[a_{ij}\right]$

Recall that the initial platform orientation depends on the desired orbit plane. Some relationships will be established between the orbit plane, earth reference system and platform (Figure A-4).

The following derivations are made which relate orbital, geocentric and platform quantities:

1. Derivation of unit vector normal to orbit plane (I_Z^*) in terms of earth-referenced axes, Ω and i.

2. Determination of unit vector I_Y in terms of earth-referenced axes, λ and ϕ'.

3. Determination of angle δ between I_Y and I_Y^* - - this angle is considered positive when orbit plane is above platform plane at launch site.

4. Determination of unit vector $I_X = I_X^*$.

5. Determination of angle γ between east and I_X axis - this angle is considered positive when I_X is north of east.

6. Determination of matrix $\left[a_{ij}\right]$.

The normal to the orbit plane can be derived by considering two rotations from the earth-referenced system: (1) A rotation about Z_E through the angle Ω, (2) about the displaced X_E axis through the angle i. The third rotation is not required since it would be about the displaced Z_E axis.

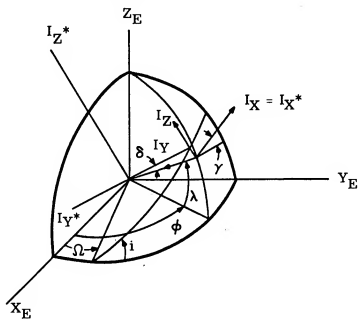
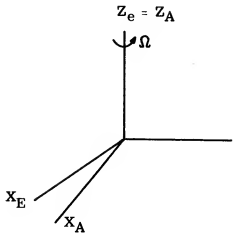


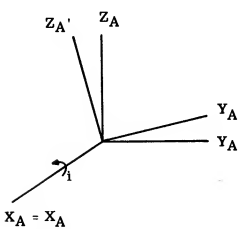
Figure A-4

1st Rotation Ω About Z_E



$$\begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix} = \begin{bmatrix} \cos\Omega & \sin\Omega & 0 \\ -\sin\Omega & \cos\Omega & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix}$$

2nd Rotation i About X_A'



$$\begin{bmatrix} X_A' \\ Y_A' \\ Z_A' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos i & \sin i \\ 0 & -\sin & \cos i \end{bmatrix} \begin{bmatrix} X_A \\ Y_A \\ Z_A \end{bmatrix}$$

From the above

$$Z_A' = \sin i \sin\Omega\, X_E - \sin i \cos\Omega\, Y_E + \cos i\, Z_E$$

and,

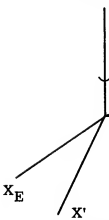
$$Z_A = I_Z^*$$

Thus, the normal to the orbit plane is given by

$$I_Z^* = I_{XE} \sin i \sin\Omega - I_{YE} \sin i \cos\Omega + I_{ZE} \cos i \qquad (1)$$

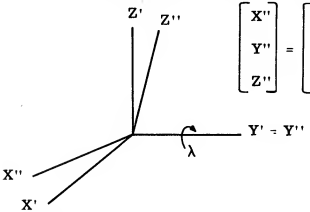
Since the platform I_Y axis has been defined as along the launch point radius vector, it can be determined by two rotations. These rotations are about Z_E through ϕ, and then about the displaced X_E axis through λ.

1. First Platform Rotation -- ϕ' About Z_E



$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \begin{bmatrix} \cos\phi' & \sin\phi' & 0 \\ -\sin\phi' & \cos\phi' & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix}$$

2. Second Platform Rotation -- λ About Y' (neg. rot.)



$$\begin{bmatrix} X'' \\ Y'' \\ Z'' \end{bmatrix} = \begin{bmatrix} \cos\lambda & 0 & \sin\lambda \\ 0 & 1 & 0 \\ -\sin\lambda & 0 & \cos\lambda \end{bmatrix} \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix}$$

From the above,

$$X'' = \cos\lambda \cos\phi' X_E + \cos\lambda \sin\phi' Y_E + \sin\lambda Z_E$$

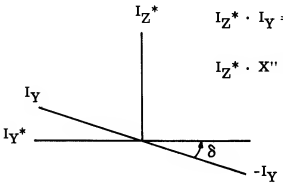
By previous definition,

$$X'' = -I_Y$$

Therefore,

$$-I_Y = I_{XE} \cos\lambda \cos\phi' + I_{YE} \cos\lambda \sin\phi' + I_{ZE} \sin\lambda \qquad (2)$$

Now since I_X and I_X^* have been defined as equal vectors, I_Y and I_Z^* must be in a common plane. The cosine of the angle between I_Y and I_Z^* is equal to the sine of the desired angle δ and can be determined by:



$$I_Z^* \cdot I_Y = \cos\left(\frac{\pi}{2} - \delta\right) = \sin\delta = -(I_Z^* - I_Y)$$

$$I_Z^* \cdot X'' = -\sin\delta$$

And,

$$\sin\delta = \sin i \cos\lambda \sin(\phi' - \Omega) - \cos i \sin\lambda \qquad (3)$$

The vector I_X can be evaluated in terms of the above quantities by evaluating the vector cross-product of I_Z^* and $-I_X$ as follows:

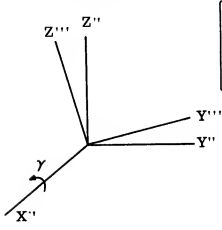
$$I_Z^* \times -I_Y = I_X \sin\left(\frac{\pi}{2} + \delta\right) = I_X \cos\delta$$

$$I_X \cos\delta = \begin{vmatrix} I_{XE} & I_{YE} & I_{ZE} \\ \sin\Omega \sin i & -\sin i \cos\Omega & \cos i \\ \cos\lambda \cos\phi' & \cos\lambda \sin\phi' & \sin\lambda \end{vmatrix}$$

$$\begin{aligned} I_X \cos\delta = {} & I_{XE}\left[-\sin i \cos \quad \sin\lambda - \cos\lambda \sin\phi' \cos i\right] \\ & + I_{YE}\left[\cos i \cos\lambda \cos\phi' - \sin\Omega \sin i \sin\lambda\right] \\ & + I_{ZE}\left[\sin\Omega \sin i \cos\lambda \sin\phi' + \sin i \cos\Omega \cos\lambda \cos\phi'\right] \end{aligned} \qquad (4)$$

This vector I_X is also defined by the third rotation which relates platform to earth values. This rotation is about X'' through the angle γ.

3. Third Platform Rotation -- γ About X''



$$\begin{bmatrix} X''' \\ Y''' \\ Z''' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & \sin\gamma \\ 0 & -\sin\gamma & \cos\gamma \end{bmatrix} \begin{bmatrix} X'' \\ Y'' \\ Z'' \end{bmatrix}$$

$$\begin{bmatrix} X''' \\ Y''' \\ Z''' \end{bmatrix} = \begin{bmatrix} \cos\lambda\cos\phi' & \cos\lambda\sin\phi' & \sin\lambda \\ -\sin\gamma\sin\lambda\cos\phi' - \cos\gamma\sin\phi' & -\sin\gamma\sin\lambda\sin\phi' + \cos\gamma\cos\phi' & \sin\gamma\cos\lambda \\ -\cos\gamma\sin\lambda\cos\phi' + \sin\gamma\sin\phi' & -\cos\gamma\sin\lambda\sin\phi' - \sin\gamma\cos\phi' & \cos\gamma\cos\lambda \end{bmatrix} \begin{bmatrix} X_E \\ Y_E \\ Z_E \end{bmatrix}$$

By previous definition,

$$X''' = -I_Y$$

$$Y''' = I_X$$

$$Z''' = I_Z \qquad (5)$$

Therefore,

$$I_X = I_{XE}\left[-\sin\gamma\sin\lambda\cos\phi' - \cos\gamma\sin\phi'\right] + I_{YE}\left[-\sin\gamma\sin\lambda\sin\phi' + \cos\gamma\cos\phi'\right] + I_{ZE}\left[\sin\gamma\cos\lambda\right] \qquad (6)$$

From equations (4) and (6),

$$I_X = I_{XE}\, A + I_{YE}\, B + I_{ZE} \sin\gamma \cos\lambda$$

$$I_X \cos\delta = I_{XE}\, C + I_{YE}\, D + I_{ZE} \left[\sin\Omega \sin i \cos\lambda \sin\phi' + \sin i \cos\Omega \cos\lambda \cos\phi'\right]$$

By equating I_{ZE} terms:

$$\sin\gamma \cos\lambda = \frac{1}{\cos\delta} \left[\sin\Omega \sin i \cos\lambda \sin\phi' + \sin i \cos\Omega \cos\lambda \cos\phi'\right]$$

This expression may be reduced to determine the sine of the angle between Y'' (east) and I_X

$$\sin\gamma = \sin i \cos(\phi' - \Omega)(\cos\delta)^{-1} \qquad (7)$$

The matrix $\left[a_{ij}\right]$ is defined by equations (4) and (5).

$$\begin{bmatrix} I_X \\ I_Y \\ I_Z \end{bmatrix} = \begin{bmatrix} -\cos\gamma\sin\phi' - \sin\gamma\sin\lambda\cos\phi' & \cos\gamma\cos\phi' - \sin\gamma\sin\lambda\sin\phi' & \sin\gamma\cos\lambda \\ -\cos\lambda\cos\phi' & -\cos\lambda\sin\phi' & -\sin\lambda \\ -\cos\gamma\sin\lambda\cos\phi' + \sin\gamma\sin\phi' & -\cos\gamma\sin\lambda\sin\phi' - \sin\gamma\cos\phi' & \cos\gamma\cos\lambda \end{bmatrix} \begin{bmatrix} I_{XE} \\ I_{YE} \\ I_{ZE} \end{bmatrix}$$

$$\begin{bmatrix} I_X \\ I_Y \\ I_Z \end{bmatrix} = \left[a_{ij}\right] \begin{bmatrix} I_{XE} \\ I_{YE} \\ I_{ZE} \end{bmatrix}$$

4. Difference Between Simulation and Actual Platform Alignment

A basic relationship was neglected in the derivation of the equations above to define the relative orientations of the several coordinate systems. In the physical application of the guidance equations, the inertial platform will be aligned such that the Y-axis is along the local gravity vector. The error associated with this simplification is, as expected, primarily noticed in out-of-phase position and velocity computations. This discussion defines the changes that will be required in order to simulate the IGS platform more closely.

The figure shows the actual relationship that will exist between the referenced inertial frame and the actual platform.

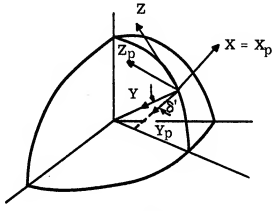


Figure A-5

Thus, the actual platform frame can be described in terms of one rotation, δ', about X. δ' is considered positive as shown, and is the angle between the local gravity vector and the geocentric radius vector. This relationship may be expressed by the following equations:

$$\begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\delta' & -\sin\delta' \\ 0 & \sin\delta' & \cos\delta' \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

Let $\sin\delta' = \epsilon_y$, then

$$\begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & -\epsilon_y \\ 0 & \epsilon_y & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

The following assumptions are implicit in this relationship:

1. $\cos\delta' = 1$ -- This is valid since $\delta' < 15\ \widehat{\text{min}}$

2. $X_p = X$ -- Again, since δ' is negligible, γ may be assumed equal to the actual rotation of the platform in azimuth. (error is less than 30 sec)

The relationship between the orbit frame and the actual platform may be established by neglecting products $\epsilon_y \eta_y$ (less than 10^{-5}):

$$\begin{bmatrix} X^* \\ Y^* \\ Z^* \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \epsilon_y - \eta_y \\ 0 & \eta_y - \epsilon_y & 1 \end{bmatrix} \begin{bmatrix} X_P \\ Y_P \\ Z_P \end{bmatrix}$$

Consider the initial conditions inserted into the computer ($X_C = 0$, $Y_C = R_o$, $Z_C = 0$), which are required to satisfy the gravity equation ($G = K/R2$), then the following relationship are valid:

$$X = 0,\ Y = R_o,\ Z = 0$$

$$X_{P_o} = 0,\ Y_{P_o} = R_o,\ Z_{P_o} = \epsilon_y R_o$$

now

$$X_P = \Delta X_P + X_{P_o} = \Delta X_P$$

$$Y_P = \Delta Y_P + Y_{P_o} = \Delta Y_P + R_o$$

$$Z_P = \Delta Z_P + Z_{P_o} = \Delta Z_P + \epsilon_y R_o$$

Where $\Delta X_P, \Delta Y_P, \Delta Z_P$ are the changes in X_P, Y_P, Z_P from time t_o to time t. Now the values in the computer can be identified:

$$X_C = \Delta X_P,\ Y_C = \Delta Y_P + R_o,\ Z_C = \Delta Z_P$$

Thus,

$$X^* = X_C$$

$$Y^* = Y_C + (\epsilon_y - \eta_y) Z_C + \epsilon_y (\epsilon_y - \eta_y) R_o$$

$$Z^* = Z_C + \epsilon_y R_o + (\eta_y - \epsilon_y) Y_C$$

This will be implemented in Stage 2 guidance equations in the following manner:

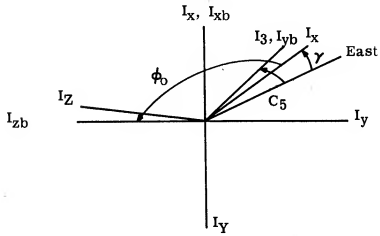
$$R_Z^* = Z + \epsilon_y R_o + (\eta_y - \epsilon_y) Y$$

$$V_\perp = \dot{Z} + (\eta_y - \epsilon_y) \dot{Y}$$

Changes in the simulation will be (1) establish the matrix between the inertial reference and the platform frame, and (2) use this matrix to transform position, velocity and accelerations from the inertial reference to the platform frame.

V. INITIAL MATRIX $\left[b_{ij}\right]$

Figure A-6 shows the relationship between the platform, capsule and launch vehicle prior to launch. We assume that the platform is aligned to the orbit plane and earth reference system as previously noted.



In Figure A-6, I_Y points toward the earth's center; I_x and I_{xb} point away from the earth's center (vertical). All other vectors are in a plane normal to I_Y.

γ = angle from east to I_X

C_5 = angle from east to I_{zb}

ϕ_o = angle from I_X to I_{zb}

Figure A-6

The following matrix may be written from Figure A-6.

$$\begin{bmatrix} I_{xb} \\ I_{yb} \\ I_{zb} \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 \\ \sin\phi_o & 0 & -\cos\phi_o \\ \cos\phi_o & 0 & \sin\phi_o \end{bmatrix} \begin{bmatrix} I_X \\ I_Y \\ I_Z \end{bmatrix}$$

In matrix $\left[b_{ij}\right]$,

$b_{13} = -\sin\psi = 0$

$b_{11} = \cos\psi \cos\theta = \cos\theta = 0$

$b_{12} = -\cos\psi \sin\theta = -\sin\theta = -1$

$b_{33} = \cos\psi \sin\phi = \sin\phi = \sin\phi_o$

$b_{23} = -\cos\psi \cos\phi = -\cos\phi = -\cos\phi_o$

From the above,

$\psi = 0$

$\theta = 90^o$

$\phi = \phi_o = C_5 - \gamma + 90^o$

VI. DESCRIPTION OF LAUNCH VEHICLE MODEL (FIGURE A-7)

The six parameters described in this section have been included in the description of the vehicle. Other parameters not defined have been omitted because (1) we want a simplified model and (2) they do not contribute appreciably to an analysis of navigation equations.

A. REFERENCE SYSTEM

The capsule, vehicle and platform systems have been defined in Section III, Coordinate Systems.

B. THRUST APPLICATION

Thrust is assumed to be applied at the engine gimbal. During Stage 1, two engines are in operation. They are symmetrically located about the vehicle longitudinal axis in the I_X, I_Z plane. The longitudinal station of the engines is denoted as X_S; the distance from the vehicle axis to the engines is Z_1 to engine A and Z_2 to engine B. During Stage 2 a single engine is used. This engine is located on the vehicle longitudinal axis. All engines are gimballed in 2-degrees-of-freedom, ϵ and δ. Positive directions of these angles are shown in Figure A-9.

C. AERODYNAMIC COEFFICIENTS

The three coefficients, axial C_A, side C_Y and normal C_N, have been determined from preliminary GEMINI Launch Vehicle (GLV) wind tunnel data. Fixed data points, which show the coefficients versus mach number, are stored for various angles of attack, α_N and α_Y. Linear interpolation is used to determine the required values. The angles of attack are defined by Figure A-7. Aerodynamic forces act through the center of pressure X_{cp}. The location of X_{cp} is also defined on the basis of mach number and angle of attack in a stored table.

D. MOMENTS OF INERTIA

The moments of inertia -- roll I_X, pitch I_Y and yaw I_Z about the center of gravity -- have been determined from preliminary GLV data sheets. These values are stored as polynomials in the computer and determined as functions of time. Location of the center of gravity, X_{cg}, is also stored as a polynomial in time.

E. VEHICLE WEIGHT

The vehicle weight is a linear function of time determined by initial weight and fuel flow rate which were determined from preliminary data sheets. Mass is determined by converting weight through use of standard gravity value, G_o.

F. THRUST

Thrust is considered constant at the values specified in Appendix F.

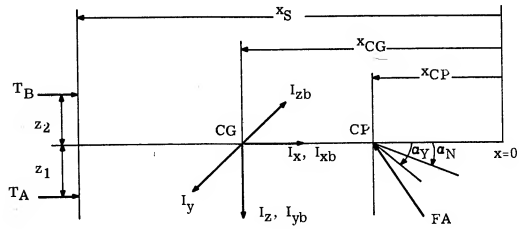


Figure A-7. Vehicle Model Used in the Simulation

VII DETERMINATION OF AERODYNAMIC FORCES

A. RELATIVE WIND

Relative wind is defined as the velocity of the vehicle with respect to the air mass. Vectorally, $\overline{V}_w = \overline{V} - \overline{V}_a$. For initial studies, we assume that the air mass has zero velocity with respect to earth. Under this assumption, the inertial velocity of the air mass is given by $\overline{\omega}_e \times \overline{R}$, where $\overline{\omega}_e$ is earth's rotation and $\overline{R}$ is the position vector of the vehicle with respect to the earth's center.

Since $\overline{\omega}_e$ can be defined as a vector in inertial space, this vector can be transformed into platform coordinates through matrix $[a_{ij}]$. The position vector, $\overline{R}$, is updated continuously and is available in the platform frame.

Thus,

$$\overline{V}_{e\ platform} = \overline{\omega}_{e\ platform} \times \overline{R}_{platform}$$

$$\overline{\omega}_{e\ platform} = [a_{ij}]\ \overline{\omega}_e = [a_{ij}]\ |\omega_e|\ \overline{Z}_E$$

$$= a_{13}\,\omega_e\,I_X + a_{23}\,\omega_e\,I_Y + a_{33}\,\omega_e\,I_Z$$

$$= \omega_{eX}\,I_X + \omega_{eY}\,I_Y + \omega_{eZ}\,I_Z$$

$$\overline{R}_{platform} = R_X\,I_X + R_Y\,I_Y + R_Z\,I_Z$$

and

$$\overline{V}_{e\ platform} = V_{eX}\,I_X + V_{eY}\,I_Y + V_{eZ}\,I_Z,$$

where

$$V_{eX} = \omega_{eY}\,R_Z - \omega_{eZ}\,R_Y$$

$$V_{eY} = \omega_{eZ}\,R_X - \omega_{eX}\,R_Z$$

$$V_{eZ} = \omega_{eX}\,R_Y - \omega_{eY}\,R_X$$

and

$$\overline{V}_w = \overline{V} - \overline{V}_e$$

$$V_{wX} = V_X - V_{eX}$$

$$V_{wY} = V_Y - V_{eY}$$

$$V_{wZ} = V_Z - V_{eZ}$$

Now, the relative wind velocity must be transformed into vehicle coordinates (Figure A-8):

$$\overline{V}_{w\ spacecraft} = [b_{ij}]\ \overline{V}_{w\ platform}$$

Explicitly,

$$V_{wxb} = b_{11} V_{wX} + b_{12} V_{wY} + b_{13} V_{wZ}$$

$$V_{wyb} = b_{21} V_{wX} + b_{22} V_{wY} + b_{23} V_{wZ}$$

$$V_{wzb} = b_{31} V_{wX} + b_{32} V_{wY} + b_{33} V_{wZ}$$

Convert into vehicle coordinates:

$$V_{wx} = V_{wxb}$$

$$V_{wy} = -V_{wzb}$$

$$V_{wz} = V_{wyb}$$

The magnitude of wind velocity, V_w, is found by root-sum squaring the components in spacecraft coordinates.

$$V_w = \sqrt{V_{wxb}^2 + V_{wyb}^2 + V_{wzb}^2}$$

B. ATMOSPHERIC COEFFICIENTS

The air density, ρ , and speed of sound, C_s, are determined by using a previously developed IBM 7090 subroutine, based on the 1959 ARDC Standard Atmosphere. Future plans provide for incorporation of data from NASA TND 595 Reference Atmosphere for Patrick AFB. At this time, surface wind conditions may also be introduced.

C. COMPUTATION OF AERODYNAMIC FORCES

Three components of aerodynamic forces -- axial F_{Ax}, side F_{Ay} and normal F_{Az} -- are defined as follows:

$$F_{Ax} = -C_A \, q \, S \, \pi$$

$$F_{Ay} = -C_Y \, q \, S \, \pi$$

$$F_{Az} = -C_N \, q \, S \pi,$$

where q is aerodynamic pressure and S_T is effective frontal area. The negative sign accounts for the direction of the force according to the definition of relative wind.

$$q = 1/2\,\rho\,V_w^2,$$

and S_T is a constant for the vehicle.

C_A, C_N and C_Y depend on mach number M and angle of attack, and are determined from stored data as previously described.

$$M = \frac{V_w}{C_S}$$

$$\alpha_Y = \tan^{-1}\frac{V_{wy}}{V_{wx}}$$

$$\alpha_N = \tan^{-1}\frac{V_{wz}}{V_{wx}}$$

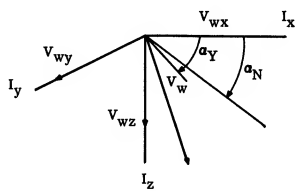


Figure A-8

Note that,

$$\alpha_Y = \tan^{-1} \frac{-V_{wzb}}{V_{wxb}}$$

and

$$\alpha_N = \tan^{-1} \frac{V_{wyb}}{V_{wxb}}$$

and

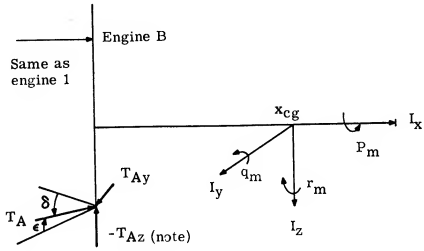
$$F_{Axb} = F_{Ax}$$

$$F_{Ayb} = F_{Az}$$

$$F_{Azb} = -F_{Ay}$$

VIII. THRUST RESOLUTION

As has been previously mentioned, the engines are gimballed in two degrees of freedom. This is shown in Figure A-9 along with the resolution of thrust into vehicle coordinates.



(Note) forces are defined in a positive direction. Therefore, a positive δ angle yields a negative force.

Figure A-9

$T_{Ax} \approx T_A \cos\delta_1 \cos\epsilon_1$ $\quad T_{Bx} \approx T_B \cos\delta_2 \cos\epsilon_2$

$T_{Ay} = T_A \sin\epsilon_1$ $\quad T_{By} = T_B \sin\epsilon_2$

$T_{Az} = -T_A \sin\delta_1$ $\quad T_{Bz} = -T_B \sin\delta_2$

IX. RESOLUTION OF PERTURBATIVE FORCES

Forces in vehicle coordinates may be summed as follows:

$$F_x = F_{Ax} + T_{Ax} + T_{Bx}$$

$$F_y = F_{Ay} + T_{Ay} + T_{By}$$

$$F_z = F_{Az} + T_{Az} + T_{Bz}$$

By transforming these forces through capsule coordinates and into the platform frame, the following equations are obtained:

$$\overline{F}_{platform} = \left[b_{ij}\right]^{-1} \overline{F}_{capsule}$$

$$F_X = b_{11} F_x + b_{21} F_z - b_{31} F_y$$

$$F_Y = b_{12} F_x + b_{22} F_z - b_{32} F_y$$

$$F_Z = b_{13} F_x + b_{23} F_z - b_{33} F_y$$

X. CENTRAL FORCE FIELD

Gravitational acceleration is determined by using a previously written subroutine which includes the first two oblateness teams. As previously specified, position of the vehicle may be transformed into an earth-referenced system, X_E, Y_E, Z_E, through $\left[a_{ij}\right]$. After computation of gravity in this system, G is transformed back into platform coordinates.

$$\overline{R}_{earth} = \left[a_{ij}\right]^{-1} \overline{R}_{platform}$$

$$\overline{G}_{platform} = \left[a_{ij}\right] \overline{G}_{earth}$$

XI UPDATING POSITION AND VELOCITY

The total acceleration acting on the vehicle may be determined from the equations in Sections IX and X:

$$\dot{\bar{V}} = \frac{\bar{F}}{M} + \bar{G}$$

or, in terms of platform coordinates,

$$\dot{V}_X = \frac{F_X}{M} + G_X$$

$$\dot{V}_Y = \frac{F_Y}{M} + G_Y$$

$$\dot{V}_Z = \frac{F_Z}{M} + G_Z$$

Velocity and position are updated by numerical integration by using Simpson's Rule.

$$\bar{V}_i = \bar{V}_{i-2} + \frac{\Delta t}{3}\left(\dot{\bar{V}}_i + 4\,\dot{\bar{V}}_{i-1} + \dot{\bar{V}}_{i-2}\right)$$

$$\bar{R}_i = \bar{R}_{i-2} + \frac{\Delta t}{3}\left(\bar{V}_i + 4\,\bar{V}_{i-1} + \bar{V}_{i-2}\right)$$

XII. VEHICLE ATTITUDE AND ANGULAR ROTATION

Two cases are discussed in this section. The first case is a perfect autopilot, i.e., the attitude is governed solely by the guidance equations. The second is the tie-in of an autopilot model coupled with solving moment equations to determine angular rotation.

A. "PERFECT" AUTOPILOT (ATTITUDE GOVERNED BY GUIDANCE EQUATIONS)

This simulation is designed to work in conjunction with simulated inputs determined by IBM guidance equations.* The primary outputs of these equations are vehicle errors ϵ_x, ϵ_y, ϵ_z about the three vehicle axes. Since the computation cycle is short and the expected values of the orientation errors are small, these errors are considered commutative, i.e., order of rotation about vehicle axes is not important. In this case, the following derivation demonstrates the method used to update the matrix $[b_{ij}]$.

Given:

$$\begin{bmatrix} I_{xb} \\ I_{yb} \\ I_{zb} \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \begin{bmatrix} I_X \\ I_Y \\ I_Z \end{bmatrix} \tag{1}$$

Differentiate with respect to time in X Y Z frame:

$$\begin{bmatrix} \dot{I}_{xb} \\ \dot{I}_{yb} \\ \dot{I}_{zb} \end{bmatrix}_{XYZ} = \frac{d}{dt}\begin{bmatrix} b_{ij} \end{bmatrix} \begin{bmatrix} I_X \\ I_Y \\ I_Z \end{bmatrix} \tag{2}$$

From the theorem of Coriolis,

$$\left.\frac{d\bar{R}}{dt}\right|_{XYZ} = \left.\frac{d\bar{R}}{dt}\right|_{xyz} + \bar{\omega} \times \bar{R} \tag{3}$$

---

*Appendix E

and, letting $\overline{R} = I_x + I_y + I_z$,

$$\left.\frac{dI_x}{dt}\right|_{XYZ} = \left.\frac{dI_N}{dt}\right|_{xyz} + \omega \cdot I_x$$

$$\left.\frac{dI_y}{dt}\right|_{XYZ} = \left.\frac{dI_y}{dt}\right|_{xyz} + \omega \cdot I_y$$

$$\left.\frac{dI_z}{dt}\right|_{XYZ} = \left.\frac{dI_z}{dt}\right|_{xyz} + \omega \cdot I_z \qquad (4)$$

Now

$$\left.\frac{dI_x}{dt}\right|_{xyz} = 0; \quad \left.\frac{dI_y}{dt}\right|_{xyz} = 0; \quad \left.\frac{dI_z}{dt}\right|_{xyz} = 0,$$

and

$$\omega \cdot I_x = \omega_z I_y - \omega_y I_z$$

$$\omega \cdot I_y = \omega_x I_z - \omega_z I_x$$

$$\omega \cdot I_z = \omega_y I_x - \omega_x I_y \qquad (5)$$

In matrix notation,

$$\begin{bmatrix} \dot{I}_{xb} \\ \dot{I}_{yb} \\ \dot{I}_{zb} \end{bmatrix}_{XYZ} = \begin{bmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{bmatrix} \begin{bmatrix} I_x \\ I_y \\ I_z \end{bmatrix}$$

$$= \begin{bmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{bmatrix} \begin{bmatrix} \quad \\ b_{ij} \\ \quad \end{bmatrix} \begin{bmatrix} I_X \\ I_Y \\ I_Z \end{bmatrix} \qquad (6)$$

Equate (6) and (2):

$$\left[\dot{b}_{ij}\right] = \begin{bmatrix} 0 & \omega_z & -\omega_y \\ -\omega_z & 0 & \omega_x \\ \omega_y & -\omega_x & 0 \end{bmatrix} \left[b_{ij}\right] \tag{7}$$

The nine differential equations for $\dot{b}_{ij}$ can now be written:

$\dot{b}_{11} = \omega_{zb}\, b_{21} - \omega_{yb}\, b_{31}$

$\dot{b}_{12} = \omega_{zb}\, b_{22} - \omega_{yb}\, b_{32}$

$\dot{b}_{13} = \omega_{zb}\, b_{23} - \omega_{yb}\, b_{33}$

$\dot{b}_{21} = \omega_{xb}\, b_{31} - \omega_{zb}\, b_{11}$

$\dot{b}_{22} = \omega_{xb}\, b_{32} - \omega_{zb}\, b_{12}$

$\dot{b}_{23} = \omega_{xb}\, b_{33} - \omega_{zb}\, b_{13}$

$\dot{b}_{31} = \omega_{yb}\, b_{11} - \omega_{xb}\, b_{21}$

$\dot{b}_{32} = \omega_{yb}\, b_{12} - \omega_{xb}\, b_{22}$

$\dot{b}_{33} = \omega_{yb}\, b_{13} - \omega_{zb}\, b_{23}$

Use the first two terms of a Taylor series expansion; then

$b_{ij\,t} = b_{ij\,t-1} + \dot{b}_{ij}\,\Delta t$

Let $\omega\Delta t = \epsilon$ and simplify:

$b11_i = b11_{i-1} + \epsilon_{zb}\, b21_{i-1} - \epsilon_{yb}\, b31_{i-1}$

$b12_i = b12_{i-1} + \epsilon_{zb}\, b22_{i-1} - \epsilon_{yb}\, b32_{i-1}$

$b13_i = b13_{i-1} + \epsilon_{zb}\, b23_{i-1} - \epsilon_{yb}\, b33_{i-1}$

$b21_i = b21_{i-1} + \epsilon_{xb}\, b31_{i-1} - \epsilon_{zb}\, b11_{i-1}$

$$b22_i = b22_{i-1} + \epsilon_{xb}\, b32_{i-1} - \epsilon_{zb}\, b12_{i-1}$$

$$b23_i = b23_{i-1} + \epsilon_{xb}\, b33_{i-1} - \epsilon_{zb}\, b13_{i-1}$$

$$b31_i = b31_{i-1} + \epsilon_{yb}\, b11_{i-1} - \epsilon_{xb}\, b21_{i-1}$$

$$b32_i = b32_{i-1} + \epsilon_{yb}\, b12_{i-1} - \epsilon_{xb}\, b22_{i-1}$$

$$b33_i = b33_{i-1} + \epsilon_{yb}\, b13_{i-1} - \epsilon_{xb}\, b23_{i-1}$$

where ϵ_{xb}, ϵ_{yb}, ϵ_{zb} are the attitude error outputs of the guidance equations in capsule coordinates.

This detailed derivation need not be presented since the assumption was made previously that the rotations were commutative. However, equation (7) will be used in subsequent discussions. Note that the relationship between spacecraft and launch vehicle quantities has been previously stated.

$$\epsilon_{xb} = \epsilon_x$$

$$\epsilon_{yb} = \epsilon_z$$

$$\epsilon_{zb} = -\epsilon_y$$

B. ANGULAR ROTATION (AUTOPILOT TIE-IN COUPLED WITH SOLVING OF MOMENT EQUATIONS)

In the general simulation where the inclusion of an autopilot model is desirable, additional computations are made. This simulation uses combined analog and digital computers. In this discussion, we do not intend to define the interface or organization of computations between analog or digital equipment but to define the equations which must be solved.

When we solve the moment equations, we must determine some physical properties of the vehicle specifically, moments of inertia, location of applied forces, and locations of the center of gravity. Recall from past discussion that functions of moments of inertia I_x, I_y, I_z, location of center of pressure X_{cp}, and location of the center of gravity are stored in the computer. To determine moment arms, the distance from vehicle longitudinal axis to engine gimbal (Z_i) and engine gimbal station (X_s) are given.

Thus, the moment arms of interest are

Z_1

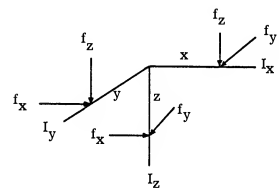
Z_2

$X_1 = X_{cg} - X_s$

$X_2 = X_{cg} - X_{cp}$

The sign is changed because stations are measured from the vehicle nose aft while the x-axis is towards the vehicle nose.

From the generalized diagram below, we can derive the moments generated by applied forces.



Consider a positive moment which would rotate the vehicle in a positive direction about the axis in question according to the right-hand screw rule:

$M_x = \Sigma f_z\, y - \Sigma f_y\, z$

$M_y = \Sigma f_x\, z - \Sigma f_z\, x$

$M_z = \Sigma f_y\, x - \Sigma f_x\, y$

From previous discussions, we have expressions for the three force components due to engines A and B and aerodynamic forces.

Thus,

$$M_x = -T_{Ay} Z_1 - T_{By} Z_2$$

$$M_y = T_{Ax} Z_1 + T_{Bx} Z_2 - T_{Az} X_1 - T_{Bz} X_1 - F_{Az} X_2$$

$$M_z = T_{Ay} X_1 + T_{By} X_1 + F_{ay} X_2$$

From Eulers' dynamical equations,

$$M_x = I_x \dot{\omega}_x - \omega_y \omega_z (I_z - I_y)$$

$$M_y = I_y \dot{\omega}_y - \omega_z \omega_x (I_x - I_z)$$

$$M_z = I_z \dot{\omega}_z - \omega_x \omega_y (I_y - I_x)$$

or

$$\dot{\omega}_x = \frac{1}{I_x} \left[M_x - \omega_y \omega_z (I_z - I_y) \right]$$

$$\dot{\omega}_y = \frac{1}{I_y} \left[M_y - \omega_z \omega_x (I_x - I_z) \right]$$

$$\dot{\omega}_z = \frac{1}{I_3} \left[M_z - \omega_x \omega_y (I_y - I_x) \right]$$

Vehicle rate, ω_i, is generated by integrating $\dot{\omega}_i$. The results of this integration provide angular rotation rates about the body coordinate system axes. From the previous derivation [equation (3-7)] we can find the change in body-platform orientation defined by matrix $[b_{ij}]$:

$$\left[\dot{b}_{ij} \right] = \begin{bmatrix} 0 & \omega_{zb} & -\omega_{yb} \\ -\omega_{zb} & 0 & \omega_{xb} \\ \omega_{yb} & -\omega_{xb} & 0 \end{bmatrix} \left[b_{ij} \right]$$

and

$$\left[\dot{b}_{ij}\right] = \begin{bmatrix} 0 & \omega_{zb} & -\omega_{yb} \\ -\omega_{zb} & 0 & \omega_{xb} \\ \omega_{yb} & -\omega_{xb} & 0 \end{bmatrix} \left[b_{ij}\right]$$

To update matrix $\left[b_{ij}\right]$ we take the first three terms of the Taylor series expansion:

$$b_{ij}{}_{t} = b_{ij}{}_{t-1} + \dot{b}_{ij}\Delta t + \ddot{b}_{ij}\frac{\Delta t^2}{2}$$

The differential equations for angular rotation are solved in vehicle coordinates, while matrix $\left[b_{ij}\right]$ relates capsule to platform orientation. This requires that vehicle angular rotation be transformed to capsule rotations:

$$\omega_{xb} = \omega_x$$

$$\omega_{yb} = \omega_z$$

$$\omega_{zb} = -\omega_y$$

XIII DETERMINATION OF PLATFORM GIMBAL ANGLES

We can assume that yaw angle ψ is small -- nominally zero - during launch. Hence, values of gimbal angles may be obtained from matrix $\left[b_{ij}\right]$ and the terms defined in Section IV-A of this appendix.

$$b_{13} = -\sin\psi$$

$$\psi = \sin^{-1}(-b_{13}) \quad -90^{o} \angle \psi \angle \ 90^{o}$$

$$\frac{b_{12}}{b_{11}} = \frac{-\sin\theta\cos\psi}{\cos\theta\cos\psi} .$$

$$\frac{-\sin\theta}{\cos\theta} = \frac{b_{12}}{b_{11}}$$

$$\frac{b_{33}}{b_{23}} = \frac{\sin\phi\cos\psi}{-\cos\phi\cos\psi}$$

$$\frac{\sin\phi}{-\cos\phi} = \frac{b_{33}}{b_{23}}$$

Since ϕ and θ will not be 180^{o} during launch, $\theta = 0$ if $b_{12} = 0$ and $\phi = 0$ if $b_{33} = 0$.

Appendix B

AUTOPILOT AND RIGID BODY SIMULATION -- ANALOG PROGRAM

Appendix B

AUTOPILOT AND RIGID BODY SIMULATION -- ANALOG PROGRAM

A. ASCENT GUIDANCE COMBINED SIMULATION

In the hybrid simulation, the analog computer was used to simulate the autopilots, engine servos, and the rigid body model of the GEMINI launch vehicles in pitch, roll, and yaw channels. The operations performed are shown in the block diagram in Figure B-1, which indicates all the quantities used and transferred by each block. The contents of each block in Figure B-1 are described in this appendix.

The autopilots, engine servos, rate feedback filters, and associated gains have similar characteristics from channel to channel. A typical channel (roll, pitch, or yaw) is illustrated in Figure B-2; all gain factors, time constants, and transfer functions representing these items are given.

In order to check the operation of each of the closed loops, each loop was independently subjected to a unit step input. For the purposes of the test, the rigid body model was assumed to be a constant gain factor. The response of the pitch and yaw channels is presented for times before and after the gain change (at 105 sec. of flight time) in Figures B-3 and B-4. The gain of the roll channel is constant, and step response is presented as in Figure B-5. The areas on the analog computer diagram (Figure B-6), where the autopilot, engine servo, and rate filter transfer functions were simulated, are labeled and enclosed with dotted lines. The attitude error inputs $\Delta\theta$, $\Delta\psi$, and $\Delta\Psi$ in launch vehicle coordinates, are indicated in Figures B-1 and B-6.

The equations representing the rigid body model of the launch vehicle have been derived in Appendix A and are as follows:

1. Moment arms for thrust and aerodynamic components of force, respectively:

$$X_1 = X_{cg} - X_S$$

$$X_2 = X_{cg} - X_{cp}$$

2. Thrust components along the x, y, and z axes for both engines:*

$$T_{ax} = T_a \cos \epsilon_1 \cos \delta$$

$$T_{ay} = T_a \sin \epsilon_1$$

$$T_{az} = -T_a \sin \delta$$

$$T_{bx} = T_b \cos \epsilon_2 \cos \delta$$

$$T_{by} = T_b \sin \epsilon_2$$

$$T_{bz} = -T_b \sin \delta$$

3. Euler's dynamical equations:

$$\dot{\omega}_x = \frac{M_x}{I_x}$$

$$\dot{\omega}_y = I_y^{-1} \left[M_y - \omega_z \omega_x (I_x - I_z) \right]$$

$$\dot{\omega}_z = I_z^{-1} \left[M_z - \omega_y \omega_x (I_y - I_x) \right]$$

The effective thrust moment arm, X_1, is a function of time because the motion of the center of gravity is dependent upon mass flow rate which was assumed constant. Therefore, X_1 was programmed on diode function generator F61. Time was received from the digital program on trunk 41. For similar reasons, the moment of inertia about the y-axis was set up on function generator F71. The actual functions programmed are illustrated in Figure B-7. These functions were derived from data in Appendix G. The aerodynamic moment arm, x_2, was formed by summing x_1, x_S, and x_{cp} with the proper signs at the input to amplifier 33. x_{cp} was received from the digital program on trunk 45.

The thrust components are somewhat disguised in the analog computer diagram, but a check of the equations will provide the reason. First, in the equation for the moment about the y-axis, the terms $T_{Ax} Z_1$ and $T_{Bx} Z_2$ appear.

But,

$$T_{ax} Z_1 + T_{bx} Z_2 = T_a \cos \delta (\cos \epsilon_1 - \cos \epsilon_2).$$

---

* δ angles are assumed equal for engines A and B

Since the engine gimbal angles, ϵ_1 and ϵ_2 are limited to 4.5 deg. ($\cos \epsilon_1 - \cos \epsilon_2$ will be very small. Typically, neglecting this term altogether would cause an error less than 0.01 percent according to actual runs, so the term was not included in the computation. By eliminating T_{Ax} and T_{Bx}, we also neglect all cosine terms. Since δ, ϵ_1, and ϵ_2 are less than 4.5 deg. the sine of the angle can be assumed equal to the angle in radians which could cause an error of no more than 0.1 percent. With these simplifications plus the constant thrust level, we can obtain scaled thrust components merely by setting a potentiometer following the engine servo output. Potentiometers P06, P26, and Q96 were used for this purpose.

One further simplification was made in the simulation regarding the summing and differencing operations performed at the output of the autopilots as seen in Figure B-1. Since the engine servos are linear, the summing and differencing can be moved to the servo output. If we assign values to the quantities at the roll and yaw engine servo outputs and trace the flow through to the formation of the moment about the x-axis, the summing and differencing is unnecessary since a similar operation is performed at this point which cancels the first ($z_1 = -z_2$). The net effect is that twice the quantity in the roll channel appears to be used in computing the x component of moment, and twice the quantity in the yaw channel appears to contribute to the moment about the z-axis.

With these simplifications, it is relatively easy to find the total moments about each axis. The only additional quantities needed are the aerodynamic forces which are supplied on trunks 46 and 47. The scaled moments about the x, y, and z axes appear at the outputs of potentiometer 26 and amplifiers 21 and M52, respectively, in Figure B-6.

The moments and moments of inertia are next inserted in Euler's dynamical equations to yield angular rates about the launch vehicle axes. The moment of inertia about the x-axis is constant, and the moments of inertia about the y- and z-axes are assumed equal. Rates about x-, y-, and z-axes appear at the outputs of integrators 30, 00, and 01, respectively, and then are sent with engine gimbal angles to the digital program on the trunks indicated in Figure B-6.

The gain change at 105 sec. was implemented using comparator relays, which have 105 V and time as inputs. The method in which information was exchanged between computers is described in the following section.

B. USE OF SIMULATION LABORATORY INTERCONNECTION EQUIPMENT

The interface between the analog computer and the IBM 7090 DPS consists of four units:

(1) D/A and A/D converters (digital-to-analog and analog-to-digital)

(2) IBM 737 core memory

(3) Data Controller (DACON),

(4) Real-Time Control Unit (RTCU)

Data is transferred between computers by feeding information from the 7090 through the RTCU to specified locations in the 737 memory (Figure B-8). This information is taken from memory by DACON through the RTCU and sent to the D/A converters, one channel at a time. When the transfers to the analog computer are complete, DACON takes information from the A/D converters and relocates it in other memory locations, one channel at a time. This operation completes one cycle of the DACON program. The order in which the data transfers are made can be programmed on the DACON control board.

The memory now contains information and is prepared to transmit to the 7090. When the 7090 requests information from the memory, data transfer between the 737 memory and the 7090 is initiated by an interrupt signal, which comes from the RTCU at fixed time intervals. This signal produces an unconditional GO TO statement in the 7090 program which causes data to be transferred. When the 7090 has received and transmitted information to the 737 memory, it uses this information in the computation of new program results. The digital program must complete its computation cycle using the new information before another interrupt signal is transmitted by the RTCU. The analog computer was time-scaled by a factor of 4 to 1 because of the length of the digital program cycle (1 sec. analog time equals 4 sec. real time). The length of the digital program computation cycle was dictated by the large printout requirements.

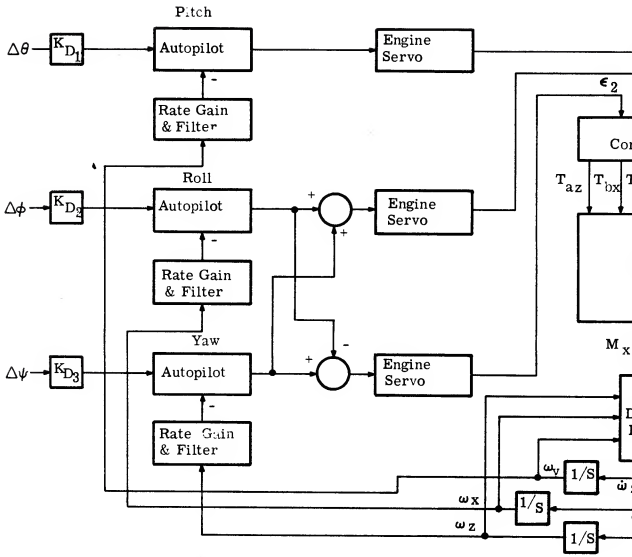
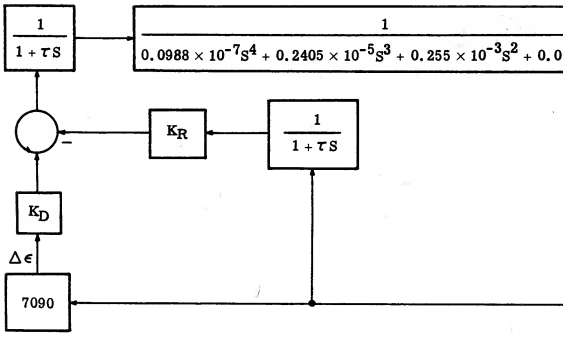


Figure B-1. Analog Portion of Ascent Guidance Combined Simulati



	K_D	K_R	τ in sec.
Pitch, Yaw 0-105 sec.	0.65%	0.4%/sec.	0.05882
Pitch, Yaw 105-150 sec.	0.21%	0.165%/sec.	0.05882
Roll 0-150 sec.	0.31%	0.07%/sec.	0.008196

Figure B-2. Autopilot Transfer Function

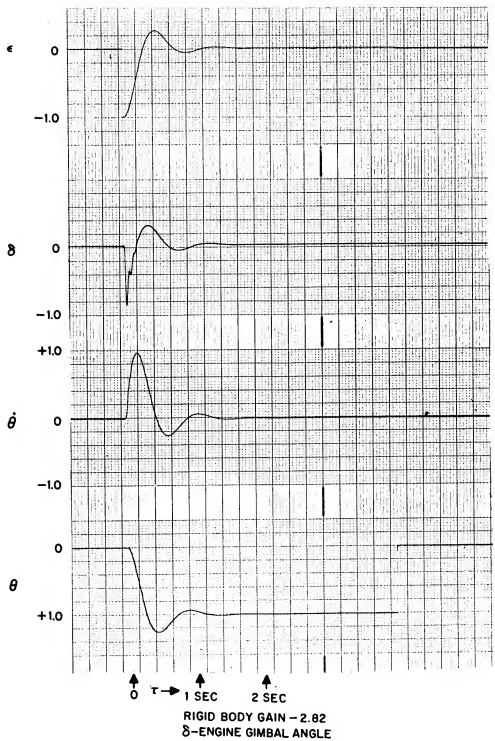


Figure B-3. Pitch Channel Closed Loop Response Before Gain Change (Continuous)

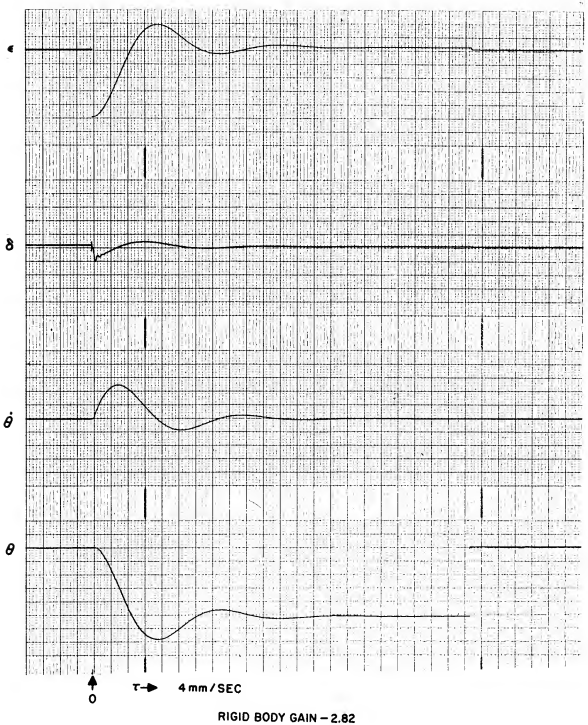


Figure B-4. Pitch Channel Closed Loop Response
After Gain Change (Continuous)

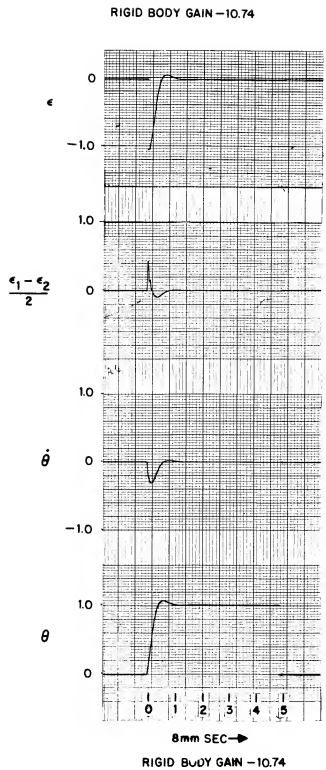


Figure B-5. Roll Channel Closed Loop Response (Continuous)

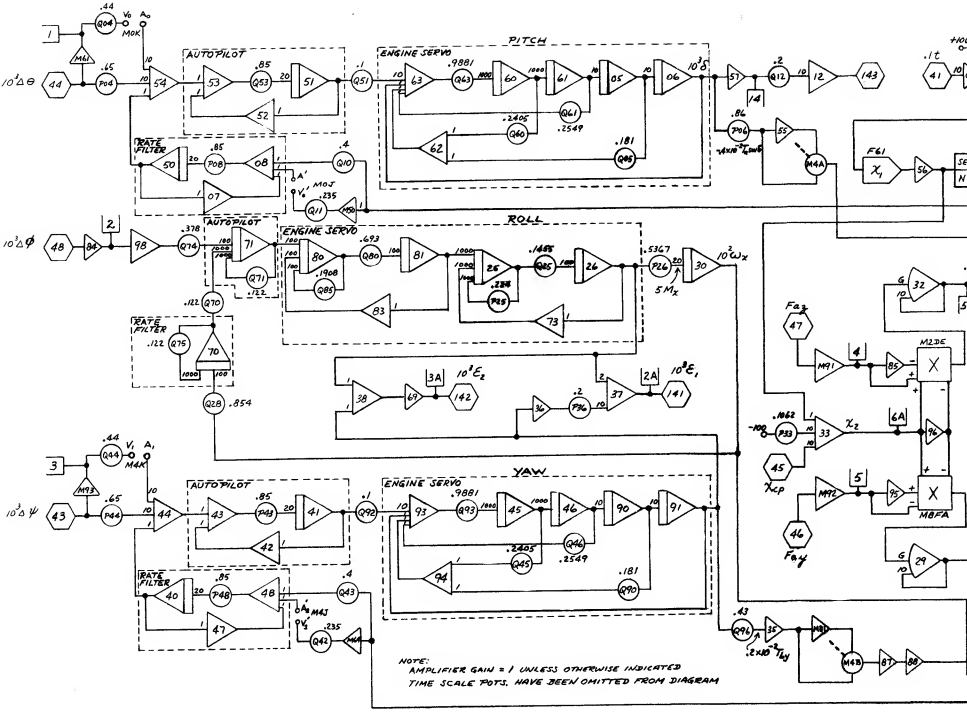


Figure B-6. First Stage Ascent Guidance Analog Computer Diagram

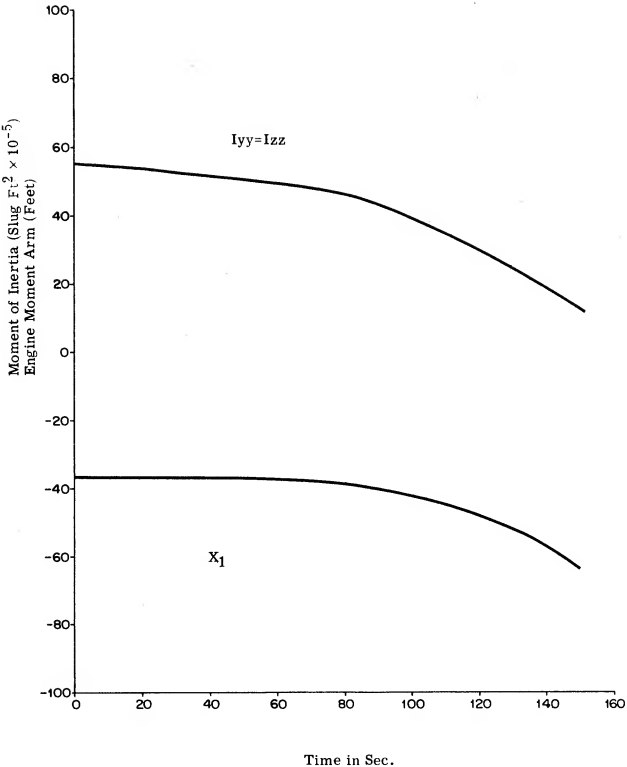


Figure B-7. Moment of Inertia, Engine Moment Arm

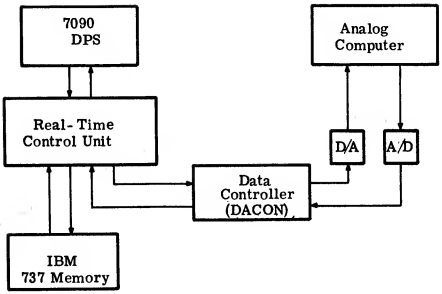


Figure B-8. Simplified Diagram of Hybrid Simulation

Appendix C

AUTOPILOT AND RIGID BODY SIMULATION--DIGITAL PROGRAM

Appendix C

AUTOPILOT AND RIGID BODY SIMULATION--DIGITAL PROGRAM

I. INTRODUCTION

This appendix describes the digital simulation of the GEMINI launch vehicle autopilot and rigid body dynamics. The program consists of a control program and eight subroutines for generating the required equations.

The autopilot model used for simulation is shown in Figure C-1. This model also applies to the pitch, yaw and roll channels for both Stage 1 and Stage 2. Since the analysis of Stages 1 and 2 are similar, only Stage 1 is discussed in detail.

The autopilot equations are developed in this appendix. The derivation of the equations used for rigid body dynamics is included in Appendix A. Certain effects -- platform gimbal and rate gyro dynamics, and the gain amplifier lags in the autopilot -- have not been included because they are believed insignificant and beyond the scope and purpose of the simulation.

The symbols used in this program are consistent with those in Appendix A. However, a glossary has been attached which defines the "A" quantities not included elsewhere.

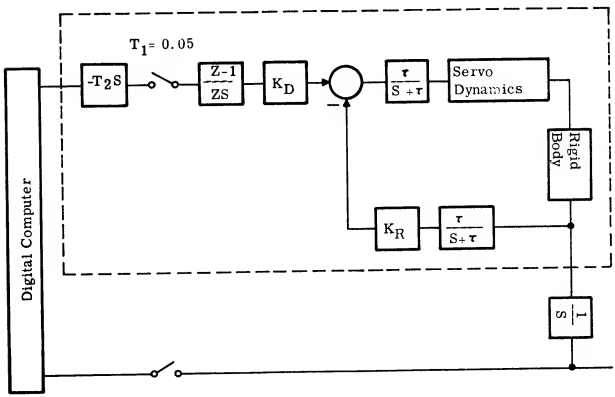


Figure C-1. GEMINI Launch Vehicle Model

II. DETERMINATION OF DIFFERENCE EQUATIONS

The servo dynamics in the pitch channel of Stage 1 (Figure C-2) have been reduced from compatibility data, supplied by Martin Aircraft Co., to fourth order by ignoring poles at (S + 220550.0) and (S + 511.67), and a zero at (S + 9918.7). For the present time, the rigid body dynamics will be represented by a first order integrator. The root locus plot of the continuous system (Figure C-2) is shown in Figure C-3 and is a function of $(K_R K_B)$.[1] To obtain difference equations which accurately approximate the continuous system, a sampling rate was chosen in which the root loci of the sampled system and the continuous system agreed to within a reasonable tolerance. The z transforms of Figure C-4 are for a sampling period of 0.01 sec. Note that a one-period time lag has been introduced in the feedback loop. This lag is necessary to offset the inherent phase lead of the z transformations. If you compare the root loci of the sampled system (Figure C-5) and the continuous system (Figure C-3), you will observe that the two systems are essentially identical in phase and gain over the specified frequency range. Since the operating point $(K_R K_B)$ is not expected to exceed about 5.0, the control poles of the z-transformed system will be accurate to better than 1 percent. This accuracy is believed to be considerably higher than the data on which the equations are based.

The next step is to program the difference equations resulting from the z-transformed system. A convenient and desirable programming check requires only the comparison of the response for a specific gain to that predicted by the root loci. A good choice of gain from Figure C-5 is 21.05, which causes a conditionally stable system. The programmed system response to a step input for this value of gain is shown in Figure C-6 and oscillates with no damping at 13.3 radians/sec. as was indicated by the root locus plot.

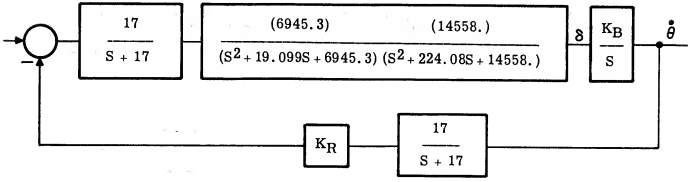


Figure C-2. Pitch Channel (Continuous)

1. All root loci and z transforms were obtained from an IBM 7090 program. See "Numerical Methods for the Synthesis of Linear Control Systems" by M.E. Fowler, IBM No. 62-907-417.

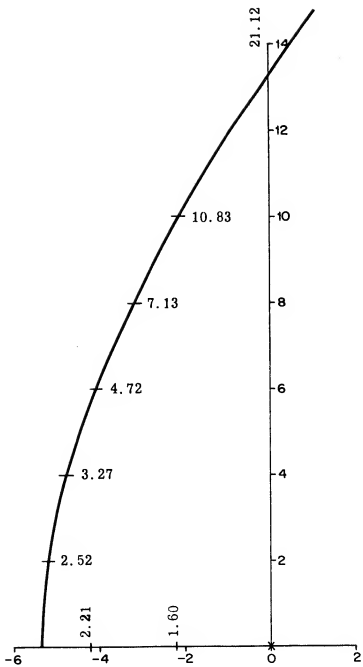


Figure C-3. Continuous System Root Locus Diagram

The remaining pitch and yaw channels are treated in a similar manner. For the vehicle in question, the servo dynamics are identical for all three channels, and the filter constant (τ) is the same for the pitch and yaw channels. Thus, the yaw channel difference equations are the same as those for pitch. For the roll channel, the filter constant is 122.0 rather 17.0; therefore; different coefficients for the z transforms must be obtained. With the same sampling period of 0.01 sec., the sampled system for roll is shown in Figure C-7.

Now that the difference equations for the three channels have been obtained, programmed, and checked; the rigid body approximations are replaced by the true rigid body dynamics which include coupling. Figure C-8 shows the response of the final program to a step input at 100 sec.

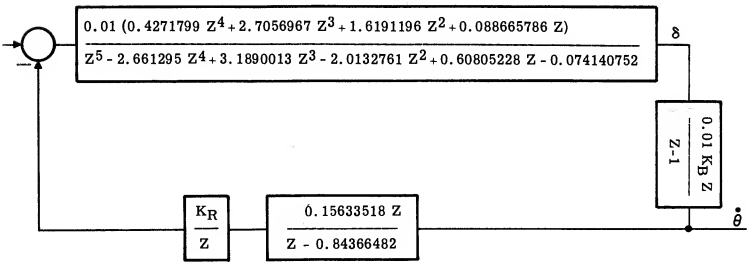


Figure C-4. Pitch Channel (Sampled at T = 0.01)

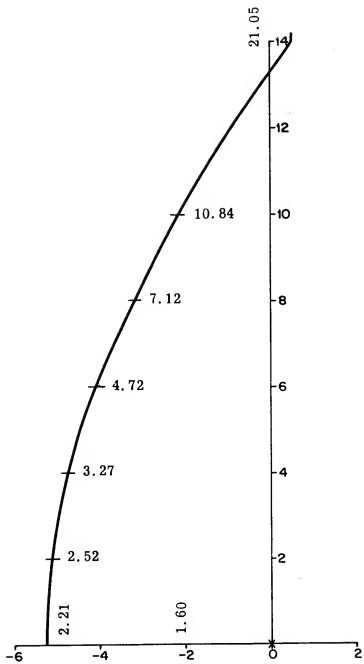


Figure C-5. Sampled System (T = 0.01) Root Locus Diagram

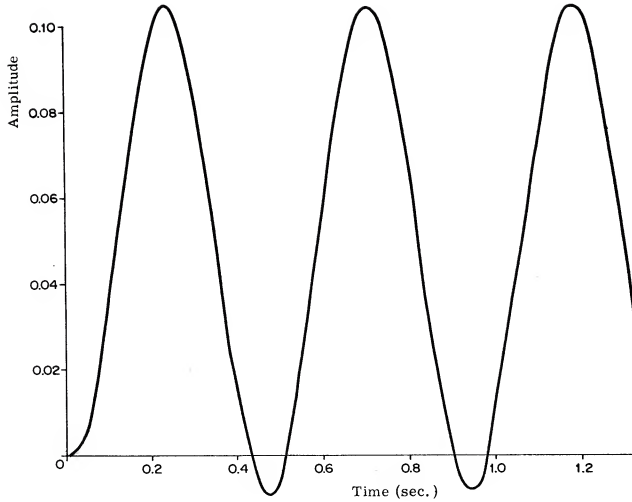


Figure C-6. Transient Response of Sampled System

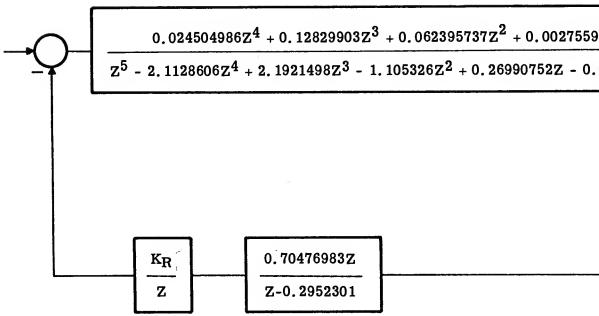
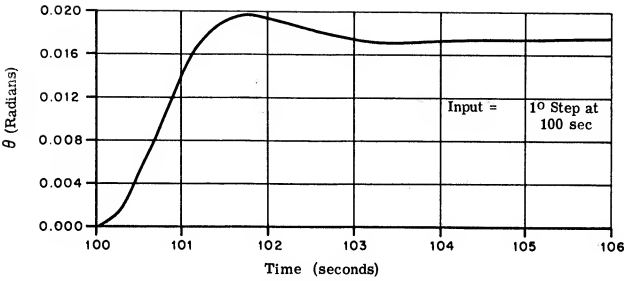
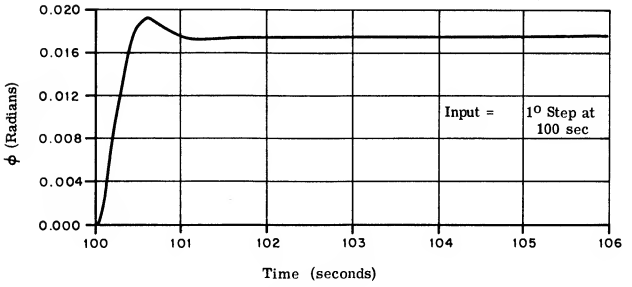


Figure C-7. Sampled Roll Channel (T = 0.01)



A. Pitch (Yaw) Channel



B. Roll Channel

Figure C-8. Transient Response in Final Program

III. COMPUTER PROGRAM

The nine subroutines, which will be referred to collectively as the vehicle program, are listed below, and the primary functions of each are discussed. Because the vehicle program difference equations are for a sampling period of 0.01 sec. and the main program sampling period is 0.05 sec., the vehicle program completes five cycles for every cycle of the main program.

A. SUBROUTINE MC

This subroutine provides communication between the main program and the remaining subroutines of the vehicle program. Since the main program and the vehicle program were not written by the same individual, this appeared to be the most convenient means of joining the two programs. The main program calls subroutine MC every 0.05 sec. which, in turn, calls the remaining subroutines of the vehicle program. Received from the main program are the three error commands for pitch, yaw, and roll — the location of the center of pressure and the launch vehicle y and z components of the aerodynamic force. The three rates and the three engine gimbal angles are returned to the main program 0.05 sec. later in flight time. These quantities are the averaged values over the 0.05-sec. sampling period. This subroutine also switches from Stage 1 to Stage 2 vehicle dynamics through sensing of main environment conditions.

B. SUBROUTINE CON1

Entry to this subroutine is made only once — the first time the vehicle program is called. It's function is to define the values of constants used in remaining subroutines for the Stage 1 equations.

C. SUBROUTINE SD1

SD1 computes the engine gimbal angles by solving the forward loop difference equations supplied with the error command signals. When necessary, it limits them to the maximum specified value of 4.5 deg. The computation time lag (T_2), shown in Figure C-1, is also handled in this subroutine. According to whether the read-in value of the constant A(10) is 0, 1, or 2, the time delay will be 0, 10, or 20 msec, respectively. The final duty of SD1 is to change the values of autopilot gain constants K_R and K_D at the end of 105 sec. of flight time.

D. SUBROUTINE RB1

The rigid body moment equations are computed in RB1. The moment of inertia and the control moment arm (Distance from C.G. to point of thrust application) are both obtained from polynomials which were fitted by the method of least squares with weight as the independent variable. For the Stage 1 both polynomials are fifth degree; they are fourth degree for Stage 2. Plots of these polynomials as a function of time are shown in Figures C-9 and C-10 for Stages 1 and 2, respectively. These plots are obtained directly from data presented in Appendix G.

The moment equations for Stage 1 are as follows:

$$Mx = -TZ_c \left(\sin \epsilon_1 - \sin \epsilon_2 \right)$$

$$My = T \left[-2X_c \sin \delta + Z_c \left(\cos \epsilon_1 - \cos \epsilon_2 \right) \right] - F_{az} X_a$$

$$Mz = T X_c \left(\sin \epsilon_1 + \sin \epsilon_2 \right) + F_{ay} X_a$$

Included in these equations is the approximation that $(\cos^2 \epsilon - \sin^2 \delta) \cong (\cos \epsilon \cos \delta)$ for the small angles involved. Fay and Faz are the aerodynamic forces and Xa is the aerodynamic moment arm; all of which are supplied by the main program.

The moment equations for Stage 2 are as follows:

$$Mx = T_1 Y_c \sin \eta$$

$$My = -T X_c \sin \delta + T_1 X_c \sin \eta$$

$$Mz = T X_c \sin \epsilon$$

T_1 is the roll nozzle thrust which is located on the y-axis, Y_c feet from the vehicle center line. The aerodynamic forces are considered to be zero for Stage 2.

The rates are then obtained for both stages from the equations:

$$\dot{\omega}_x = Mx/Ix$$

$$\dot{\omega}_y = (My + \omega_z \omega_x (Iz - Ix)) / Iy$$

$$\dot{\omega}_z = (Mz + \omega_x \omega_y (Ix - Iy)) / Iz$$

E. SUBROUTINE RL1

RL1 simply solves the rate loop difference equations.

F. SUBROUTINES CON2, SD2, RB2, RL2

These subroutines perform in a similar manner to their Stage 1 counterparts when the main environment indicates initiation of Stage 2.

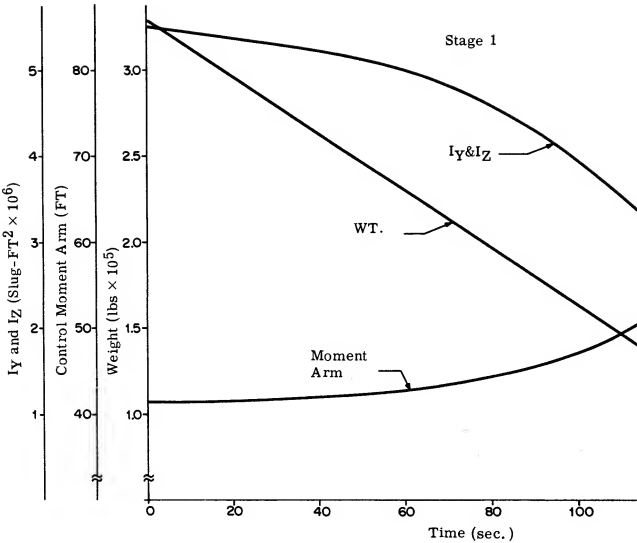


Figure C-9. Stage 1 Vehicle Characteristics

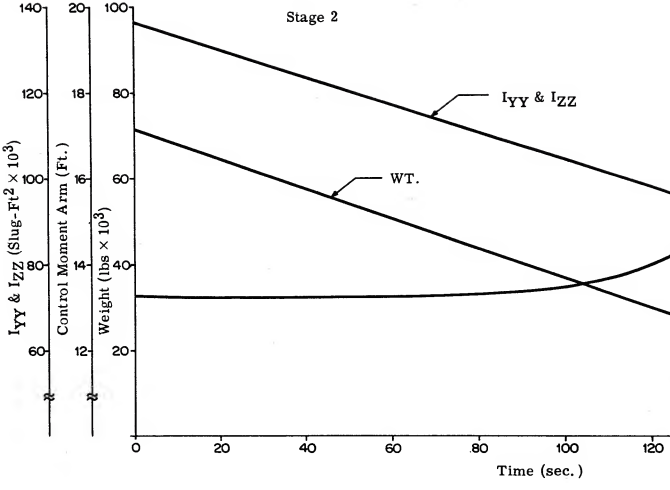


Figure C-10. Stage 2 Vehicle Characteristics

GLOSSARY OF "A" DEFINITIONS

1. Time
2. T (0.01 sec)
3. ——
4. ——
5. ——
6. ——
7. K_D ($\Delta\theta$)
8. K_D ($\Delta\psi$)
9. K_D ($\Delta\phi$)
10. Delay Control (0 = No Delay, 1 = 0.01 sec Delay, 2 = 0.02 sec Delay)
11. Z_c (Distance from thruster ℄ to vehicle ℄; Stage 1)
    Y_c (Distance from vehicle ℄ to T_1 thruster; Stage 2)
12. T (Engine Thrust)
13. ——
14. WT.
15. X_c (Control moment arm - Distance from STA 1274 to C.G. (Ft.))
16. T_1 (Roll nozzle thrust - Stage 2)
17. K_R (Roll Channel)
18. K_R (Pitch and Yaw Channels)
19. K_D (Roll Channel)
20. K_D (Pitch and Yaw Channels)
21. Z^0 num. coeff. (Pitch and Yaw forward loop)
22. Z^1 num. coeff. (Pitch and Yaw forward loop)
23. Z^2 num. coeff. (Pitch and Yaw forward loop)
24. Z^3 num. coeff. (Pitch and Yaw forward loop)
25. Z^4 num. coeff. (Pitch and Yaw forward loop)
26. ——
27. ——
28. $\Delta\theta$ (Pitch error command)
29. $\Delta\psi$ (Yaw error command)
30. $\Delta\phi$ (Roll error command)
31. Z^0 Denom. coeff. (Pitch and Yaw forward loop)
32. Z^1 Denom. coeff. (Pitch and Yaw forward loop)
33. Z^2 Denom. coeff. (Pitch and Yaw forward loop)
34. Z^3 Denom. coeff. (Pitch and Yaw forward loop)
35. Z^4 Denom. coeff. (Pitch and Yaw forward loop)
36. ——
37. Iyy and Izz
38. Ixx
39. ——
40. θ_F (Pitch feedback loop value)
41. δ (T-5) (Pitch channel)

42. δ (T-4) (Pitch channel)
43. δ (T-3) (Pitch channel)
44. δ (T-2) (Pitch channel)
45. δ (T-1) (Pitch channel)
46. δ (T) (Pitch channel)
47. ϵ_1 (Yaw angle of engine No. 1 Stage 1)
48. ϵ_2 (Yaw angle of engine No. 2 Stage 2)
49. $\dot{\omega}_y$ (Pitch channel)
50. ω_y or θ or q
51. $\epsilon^{\dagger}$ (T-5) (Pitch channel)
52. $\epsilon^{\dagger}$ (T-4) (Pitch channel)
53. $\epsilon^{\dagger}$ (T-3) (Pitch channel)
54. $\epsilon^{\dagger}$ (T-2) (Pitch channel)
55. $\epsilon^{\dagger}$ (T-1) (Pitch channel)
56. $\epsilon^{\dagger}$ (T) (Pitch channel)
57. ——
58. ——
59. ——
60. $\dot{\psi}_F$ (Yaw feedback loop value)
61. δ (T-5) (Yaw channel)
62. δ (T-4) (Yaw channel)
63. δ (T-3) (Yaw channel)
64. δ (T-2) (Yaw channel)
65. δ (T-1) (Yaw channel)
66. δ (T) (Yaw channel)
67. ——
68. ——
69. $\dot{\omega}_z$ (Yaw channel)
70. ω_z or ψ or r
71. $\epsilon^{\dagger}$ (T-5) (Yaw channel)
72. $\epsilon^{\dagger}$ (T-4) (Yaw channel)
73. $\epsilon^{\dagger}$ (T-3) (Yaw channel)
74. $\epsilon^{\dagger}$ (T-2) (Yaw channel)
75. $\epsilon^{\dagger}$ (T-1) (Yaw channel)
76. $\epsilon^{\dagger}$ (T) (Yaw channel)
77. Xa (Aerodynamic Moment arm (Ft.))
78. Fay (Aerodynamic force in Y direction)
79. Faz (Aerodynamic force in Z direction)
80. Location of C.P. (Ft.) from Aft End
81. Z^0 num. coeff. (Roll channel forward loop)
82. Z^1 num. coeff. (Roll channel forward loop)
83. Z^2 num. coeff. (Roll channel forward loop)
84. Z^3 num. coeff. (Roll channel forward loop)

85. Z^4 num. coeff. (Roll channel forward loop - Stage 1)
86. ——
87. Mx (Moment about X axis)
88. My (Moment about Y axis)
89. Mz (Moment about Z axis)
90. ——
91. Z^0 denom. coeff. (Roll channel forward loop)
92. Z^1 denom. coeff. (Roll channel forward loop)
93. Z^2 denom. coeff. (Roll channel forward loop)
94. Z^3 denom. coeff. (Roll channel forward loop)
95. Z^4 denom. coeff. (Roll channel forward loop - Stage 1)
96. ——
97. ——
98. ——
99. ——
100. Φ_F (Roll feedback loop value)
101. δ (T-5) (Roll channel) — Stage 1
102. δ (T-4) (Roll channel) — Stage 1
103. δ (T-3) (Roll channel) — Stage 1
104. δ (T-2) (Roll channel) — Stage 1
105. δ (T-1) (Roll channel) — Stage 1
106. δ (T) (Roll channel) — Stage 1
δ (T-4) — Stage 2
δ (T-3) — Stage 2
δ (T-2) — Stage 2
δ (T-1) — Stage 2
δ (T) — Stage 2
107. ——
108. ——
109. $\dot{\omega}_X$ (Roll channel)
110. ω_X or $\dot{\Phi}$ or P
111. ϵ' (T-5) (Roll channel) — Stage 1
112. ϵ' (T-4) (Roll channel) — Stage 1
113. ϵ' (T-3) (Roll channel) — Stage 1
114. ϵ' (T-2) (Roll channel) — Stage 1
115. ϵ' (T-1) (Roll channel) — Stage 1
116. ϵ' (T) (Roll channel) — Stage 1
ϵ' (T-4) — Stage 2
ϵ' (T-3) — Stage 2
ϵ' (T-2) — Stage 2
ϵ' (T-1) — Stage 2
ϵ' (T) — Stage 2
117. ——
118. Sum of ω_y (A(50)) over one sample period (0.05 sec)
119. Sum of ω_z (A(70)) over one sample period (0.05 sec)
120. Sum of ω_X (A(110)) over one sample period (0.05 sec)
121. Sum of δ (A(46)) over one sample period (0.05 sec)
122. Sum of ϵ_1 (A(47)) over one sample period (0.05 sec) - Stage 1; ϵ (A(66)) - Stage 2
123. Sum of ϵ_2 (A(48)) over one sample period (0.05 sec) - Stage 1; η (A(105)) - Stage 2
124. ——
125. ——

```
*           PARK BEDFORD
*      LABEL
CMC
       SUBROUTINE MC (NOT USED)
       DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
      C         ,DV5(100)
       COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
      C        ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
       IF (TIME-.05) 1,1,2
   1   CALL CON1
   2   A(28) =-EYY
       A(29) =-EZZ
       A(30) =-EXX
       A(78) = FA(2)
       A(79) = FA(3)
       A(80) = (1274.-XCP)/12.
       CALL SD1
       YAWD  = A(118)/5.
       ROLLD = A(119)/5.
       THETD = A(120)/5.
        DELTA2= -A(121)/5.
        EPSIL1= -A(122)/5.
        EPSIL2= -A(123)/5.
       RETURN
       END
*      LABEL
CSD1
       SUBROUTINE SD1
       DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
      C         ,DV5(100)
       COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
      C        ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
       DO 20 I=1,6
   20  A(I+117) = 0.
       N = A(10)+1.
       DO 13 K=1,5
       A(1) = A(1)+A(2)
       DO 1 I=1,5
       A(I+40) = A(I+41)
       A(I+50) = A(I+51)
       A(I+60) = A(I+61)
       A(I+70) = A(I+71)
```

```
      A(I+100) = A(I+101)
    1 A(I+110) = A(I+111)
      GO TO (2,3,4,5),N
    2 A(7) = A(20)*A(28)
      A(8) = A(20)*A(29)
      A(9) = A(19)*A(30)
      N = 4
      GO TO 5
    3 N = 1
      GO TO 5
    4 N = N-1
    5 A(56) = A(7)-A(40)
      A(76) = A(8)-A(60)
      A(116) = A(9)-A(100)
      A(46) = 0.
      A(66) = 0.
      A(106) = 0.
      DO 6 I=1,5
      J = 6-I
      A(46) = A(46)+A(J+20)*A(J+50)-A(J+30)*A(J+40)
      A(66) = A(66)+A(J+20)*A(J+70)-A(J+30)*A(J+60)
    6 A(106) = A(106)+A(J+80)*A(J+110)-A(J+90)*A(J+100)
      A(47) = A(66)+A(106)
      A(48) = A(66)-A(106)
      IF (ABSF(A(46))-.078539816) 8,8,7
    7 A(46) = SIGNF(.078539816,A(46))
    8 IF (ABSF(A(47))-.078539816) 10,10,9
    9 A(47) = SIGNF(.078539816,A(47))
   10 IF (ABSF(A(48))-.078539816) 12,12,11
   11 A(48) = SIGNF(.078539816,A(48))
   12 CALL RB1
   13 CALL RL1
      IF (A(1)-105.) 15,14,14
   14 A(20) = .21
      A(18) = .165
   15 RETURN
      END
*     LABEL
CRB1
      SUBROUTINE RB1
      DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
     C          ,DV5(100)
```

```
      COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
     C        ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
      A(14) = 329000.-1662.1622*A(1)
      WT2 = A(14)*A(14)
      WT3 = WT2*A(14)
      WT4 = WT3*A(14)
      WT5 = WT4*A(14)
      A(15) = 154.86-.0017201*A(14)+.10734E-7*WT2-.3329E-13*WT3+.493255
     C         E-19*WT4-.26337E-25*WT5
      A(37) = -999960.+11.074*A(14)+.0003861*WT2-.23705E-8*WT3+.50698E-
     C         14*WT4-.36014E-20*WT5
      A(77) = A(80)-A(15)
      A(87) = A(12)*A(11)*(SINF(A(47))-SINF(A(48)))
      A(88) = A(12)*(2.*A(15)*SINF(A(46))+(COSF(A(47))-COSF(A(48)))*A(11
     C         ))-A(79)*A(77)
      A(89) = A(12)*A(15)*(SINF(A(47))+SINF(A(48)))+A(78)*A(77)
      A(109)= A(87)/A(38)
      A(49) = (A(88)+A(70)*A(110)*(A(37)-A(38)))/A(37)
      A(69) = (A(89)+A(110)*A(50)*(A(38)-A(37)))/A(37)
      A(50) = A(2)*A(49)+A(50)
      A(70) = A(2)*A(69)+A(70)
      A(110)= A(2)*A(109)+A(110)
      A(118) = A(118)+A(50)
      A(119) = A(119)+A(70)
      A(120) = A(120)+A(110)
      A(121) = A(121)+A(46)
      A(122) = A(122)+A(47)
      A(123) = A(123)+A(48)
      RETURN
      END
*     LABEL
CRL1
      SUBROUTINE RL1
      DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
     C         ,DV5(100)
      COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
     C        ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
      A(40) = .15633518*A(50)*A(18)+.84366482*A(40)
      A(60) = .15633518*A(70)*A(18)+.84366482*A(60)
      A(100)= .70476983*A(110)*A(17)+.2952301*A(100)
      RETURN
      END
```

```
*           PARK BEDFORD
*     LABEL
CMC
      SUBROUTINE MC(NOT USED)
      DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
     C          ,DV5(100)
      COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
     C       ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
       IF(LC1) 1,1,2
   1   CALL CON2
       LC1 = 2
   2   A(28) =-EYY
       A(29) =-EZZ
       A(30) =-EXX
       CALL SD2
       YAWD  = A(118)/5.
       ROLLD = A(119)/5.
       THETD = A(120)/5.
       DELTA2= A(121)/5.
       EPSIL1= A(122)/5.
       EPSIL2= A(123)/5.
       RETURN
       END
*     LABEL
CCON2
       SUBROUTINE CON2
       DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
      C          ,DV5(100)
       COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
      C       ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
       DO 1 I=1,9
   1   A(I) = 0.
       DO 2 I=11,125
   2   A(I) = 0.
       A(2) = .01
       A(11) = 3.67
       A(12) = 100000.
       A(16) = 565.
       A(17)= 5.
       A(18)= .144
       A(19)= 10.
       A(20)= .216
```

```
      A(21)=-.16298145E-8
      A(22)= .00022354559
      A(23)= .0048435752
      A(24)= .01058024?
      A(25)= .002042806
      A(31)=-.026487565
      A(32)= .11564639
      A(33)=-.91904442
      A(34)= 2.4805997
      A(35)=-2.633024?
      A(38) = 5159.
      A(81)= .13299286E-5
      A(82)= .055928213
      A(83)= .10959722
      A(84)= .20813586
      A(91)= .066116963
      A(92)=-.12566
      A(93)=-.35719627
      A(94)=-.21303089
      A(50) = -THETD
      A(70) = YAWD
      A(110) = ROLLD
      A(118) =5.*A(50)
      A(119) =5.*A(70)
      A(120) =5.*A(110)
      RETURN
      END
*     LABEL
CSD2
      SUBROUTINE SD2
      DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
     C         ,DV5(100)
      COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
     C       ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
      DO 20 I=1,6
   20 A(I+117) = 0.
      N = A(10)+1.
      DO 15 K=1,5
      A(1) = A(1)+A(2)
      DO 1 I=1,5
      A(I+40) = A(I+41)
      A(I+50) = A(I+51)
```

```
      A(I+60) = A(I+61)
    1 A(I+70) = A(I+71)
      DO 2 I=1,4
      A(I+100) = A(I+101)
    2 A(I+110) = A(I+111)
      GO TO (3,4,5,6),N
    3 A(7) = A(20)*A(28)
      A(8) = A(20)*A(29)
      A(9) = A(19)*A(30)
      N = 4
        GO TO 6
    4 N = 1
      GO TO 6
    5 N = N-1
    6 A(56) = A(7)-A(40)
      A(76) = A(8)-A(60)
       A(115) = A(9) - A(100)
      A(46) = 0.
      A(66) = 0.
      A(105) = 0.
      DO 7 I=1,5
      J = 6-I
      A(46) = A(46)+A(J+20)*A(J+50)-A(J+30)*A(J+40)
    7 A(66) = A(66)+A(J+20)*A(J+70)-A(J+30)*A(J+60)
      DO 8 I=1,4
      J = 5-I
    8 A(105) = A(105)+A(J+80)*A(J+110)-A(J+90)*A(J+100)
      IF (ABSF(A(46))-.0389208421 10,10,9
    9 A(46) = SIGNF(.0389208421,A(46))
   10 IF (ABSF(A(66))-.0389208421 12,12,11
   11 A(66) = SIGNF(.0389208421,A(66))
   12 IF (ABSF(A(105))-.5759586531 14,14,13
   13 A(105) = SIGNF(.5759586531,A(105))
   14 CALL RB2
   15 CALL RL2
      RETURN
      END
*     LABEL
CRB2
      SUBROUTINE RB2
      DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
     C          ,DV5(100)
```

```
      COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
     C       ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
      A(14) = 71500.-344.118*A(1)
      WT2 = A(14)*A(14)
      WT3 = WT2*A(14)
      WT4 = WT3*A(14)
      A(15) = 29.555-.0011695*A(14)+.31521E-7*WT2-.37686E-12*WT3+.16829
     C        E-17*WT4
      IF (A(14)-35000.) 2,1,1
    1 A(37) = 103000.+.90411*(A(14)-35000.)
      GO TO 3
    2 A(37) = -86553.+21.493*A(14)-.0010424*WT2+.23863E-7*WT3-.20586E-12
     C        *WT4
    3 A(87) = A(16)*A(11)*SINF(A(105))
      A(88) = A(15)*(A(12)*SINF(A(46))+A(16)*SINF(A(105)))
      A(89) = A(12)*A(15)*SINF(A(66))
      A(109)= A(87)/A(38)
      A(49) = (A(88)+A(70)*A(110)*(A(37)-A(38)))/A(37)
      A(69) = (A(89)+A(110)*A(50)*(A(38)-A(37)))/A(37)
      A(50) = A(2)*A(49)+A(50)
      A(70) = A(2)*A(69)+A(70)
      A(110)= A(2)*A(109)+A(110)
      A(118) = A(118)+A(50)
      A(119) = A(119)+A(70)
      A(120) = A(120)+A(110)
      A(121) = A(121)+A(46)
      A(122) = A(122)+A(47)
      A(123) = A(123)+A(48)
      RETURN
      END
*     LABEL
CRL2
      SUBROUTINE RL2
      DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
     C        ,DV5(100)
      COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
     C       ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
      A(40) = .15633518*A(50)*A(18)+.8436648 2*A(40)
      A(60) = .15633518*A(70)*A(18)+.8436648 2*A(60)
      A(100)= .70476983*A(110)*A(17)+.2952301*A(100)
      RETURN
      END
```

```
C       JAN.  ,1962
CMC
      SUBRDUTINE MC
      DIMENSION  DV1(24),DV2(20),DV3(106),FA(3),SF1(3),SF(2),A(125)
     C           ,DV5(100),DV6(24)
      CDMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,JUP,DV6,EXX,EYY,EZZ
     C,EPSIL1,EPSIL2,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
      GD TO(1,4,4),JUP
    1 IF(LC1) 2,2,3
    2 CALL CON1
      LC1 = 2.
      GO TO 3
    4 IF(LC2) 5,5,3
    5 CALL CON2
      LC2 = 2.
      LC3 = 2.
    3 A(28) = -EYY
      A(29) = -EZZ
      A(30) = -EXX
      A(78) = FA(2)
      A(79) = FA(3)
      A(80) = (1274.-XCP)/12.
      IF(LC3) 6,6,7
    6 CALL SD1
      GD TO 8
    7 CALL SD2
    8 YAWD   = A(118)/5.
      RDLLD  = A(119)/5.
      THETD  = A(120)/5.
      DELTA2 =-A(121)/5.
      EPSIL1 =-A(122)/5.
      EPSIL2 =-A(123)/5.
      RETURN
      END
*     LABEL
CCDN1
      SUBROUTINE CDN1
      DIMENSIDN DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2),A(125)
     C          ,DV5(100)
```

```
      COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,EPSIL1,EPSIL2
     C       ,DELTA2,XCP,FA,SF1,SF,SFT,DV5,A
      DO 1 I=1,9
    1 A(I) = 0.
      DO 2 I=11,125
    2 A(I) = 0.
      A(2) = .01
       A(10)= 2.
      A(11) = 2.4225
      A(12) = 215000.
      A(17)= .07
      A(18)= .4
      A(19)= .31
      A(20)= .65
      A(21)=-.23283064E-9
      A(22)= .00088665786
      A(23)= .016191196
      A(24)= .027056967
      A(25)= .0042071799
      A(31)=-.074140752
      A(32)= .60805228
      A(33)=-2.0132761
      A(34)= 3.1890013
      A(35)=-2.6612953
      A(38) = 9730.
      A(81)= .18626451E-8
      A(82)= .002755933
      A(83)= .062395737
      A(84)= .12829903
      A(85)= .024504986
      A(91)=-.025944648
      A(92)= .26990752
      A(93)=-1.105326
      A(94)= 2.1921498
      A(95)=-2.1128606
      RETURN
      END
```

Appendix D

GEMINI ASCENT GUIDANCE SIMULATION – COMPUTER PROGRAM

Appendix D

GEMINI ASCENT GUIDANCE SIMULATION – COMPUTER PROGRAM

I. INTRODUCTION

This appendix presents the math flow and the FORTRAN listings used for the simulation, which has been previously described and derived in Section II and Appendix A of this report. In addition, the specific coordinate systems and symbols are defined in this appendix. Figure D-1 illustrates the complete program flow and will be referenced frequently in this discussion. Also, the various printout formats, used at various points in the program, are included as Tables D-I through D-IV, symbols as Table D-V and FORTRAN listings for the main environment and subroutines as Table D-VI.

II. VARIATIONS BETWEEN THE PROGRAM AND APPENDIX A

Several variations exist between the actual program and the description in Appendix A:

- Coordinate Systems – Since the actual program was an adaptation of a more general ascent guidance simulation program, the coordinate systems had to be changed. These coordinate systems are described in Section IV and Figure D-2 of this appendix.
- Symbols – As a result of other changes, the symbols vary significantly. Symbols used in the computer program are listed in Table D-V.
- Matrices – Transformation matrices between coordinate systems are identified in Section IV.
- Position and Velocity Update – Basically, Simpson's integration is employed, but present gravity components are estimated, thus providing for more accurate computation than described in Appendix A.

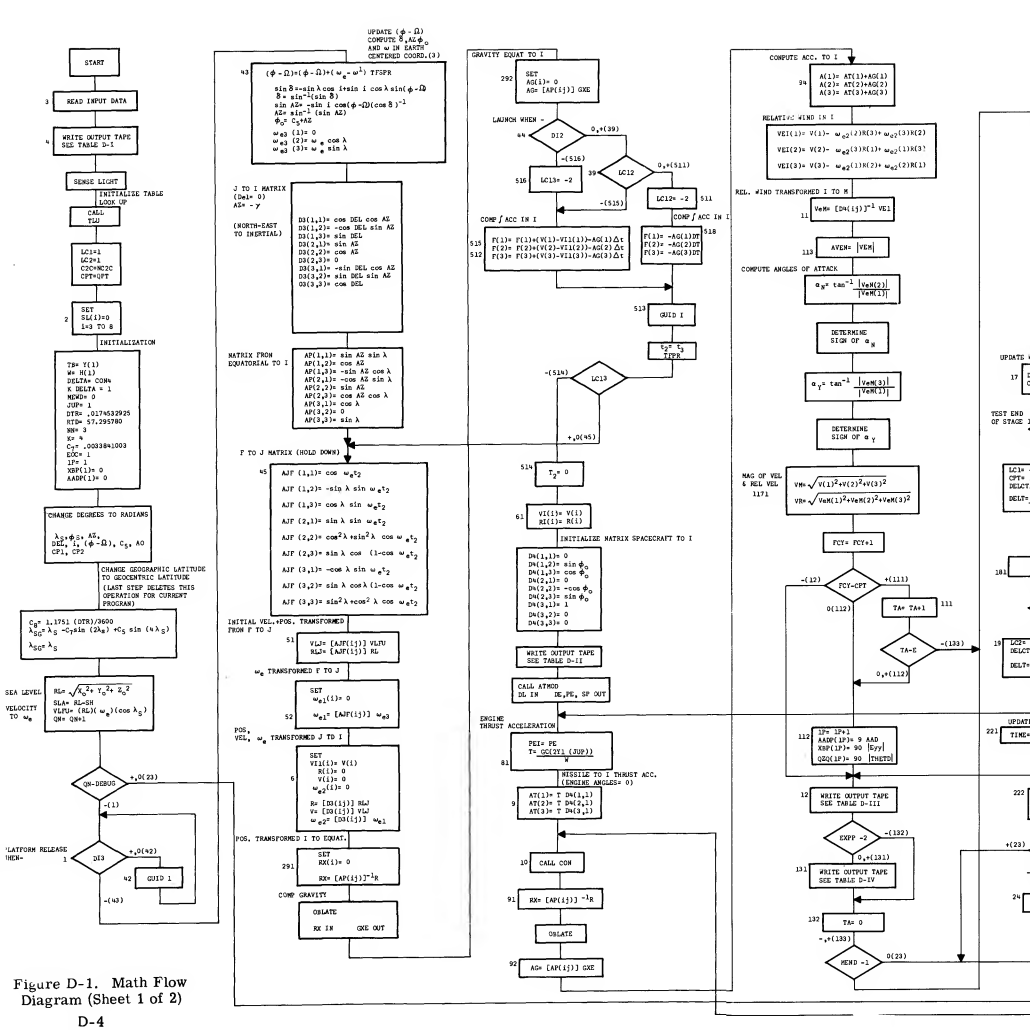


Figure D-1. Math Flow Diagram (Sheet 1 of 2)

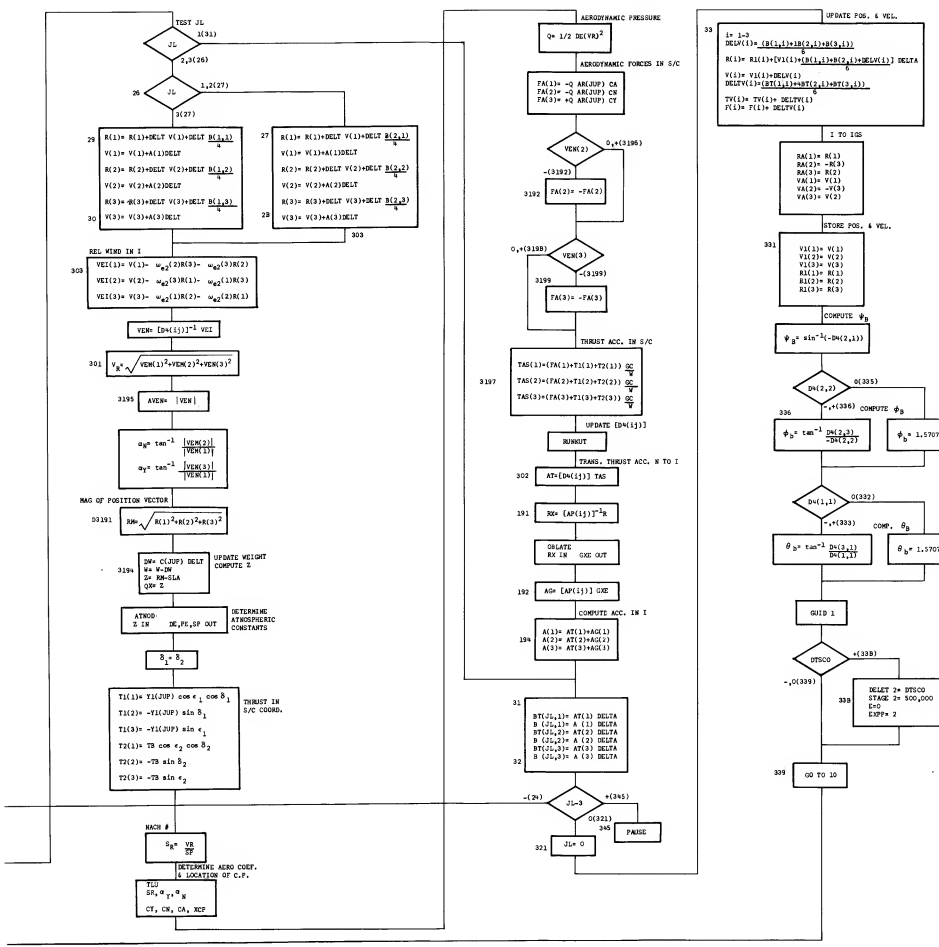


Figure D-1. Math Flow Diagram (Sheet 2 of 2)

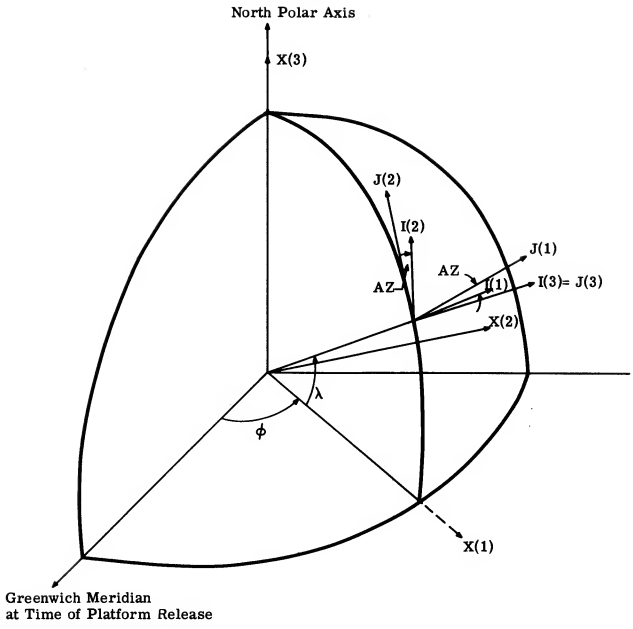


Figure D-2. Relationships of I, J and X Coordinate Frames

- Vehicle Attitude – The program uses a method of updating the vehicle platform matrix which is more accurate than the method described in Appendix A. Runge-Kutta integration with double precision is employed for this purpose.

- Hold Down – Provisions are incorporated into the math flow to permit updating of the position and velocity components during hold-down conditions. This provision is an addition to the program described in Appendix A.

III. PROGRAM SUBROUTINES

Basically, the simulation consists of the environmental or main program, which controls the program and calls the required subroutine as shown in Figure D-1. These subroutines, listed below, are accompanied by a description of their respective functions.

1. TLU (Table Look-Up)..... accepts inputs of Mach number and angles-of-attack (α_N and α_Y) and provides outputs of aerodynamic coefficients (C_A, C_N and C_Y) and location of the vehicle center of pressure (X_{CP}).

2. OBLATE..... accepts inputs of position components in an earth-reference coordinate system, RX(1), RX(2), RX(3), and provides outputs of gravitational acceleration components in the same coordinate system GXE(1), GXE(2), GXE(3) .

3. ATMOD..... accepts an input of altitude above sea level (z) and provides outputs of air density (DE) and the speed of sound (SP). Pressure (PE) is also provided but is not currently being used in thrust computations.

4. RUN-KUT..... accepts inputs of body rates (Thet D, Roll D, Psi D) and vehicle-guidance frame matrix, $[D4(i,j)]$ and provides output of updated matrix, $[D4(i,j)]$.

5. COM..... provides interface between the main program and the autopilot subroutines. Two programs were used; one for the digital autopilot simulation, and a second for the analog autopilot simulation.

Other subroutines used are the FIXER-FLOATER, DATA, PLOT, STOP and EXIT.

IV. DEFINITION OF COORDINATE SYSTEMS (Figure D-2)

A. 1 OR J FRAME

An intermediate north-east oriented inertial reference frame established at time of platform release:

1(1) = east 1(2) = north 1(3) = vertical (+up)

B. 2 OR I FRAME

The primary guidance frame, inertial reference defined at time of platform release. Basically, this frame is related to the J frame by a single rotation about 1(3). Relationship of this frame to the platform frame is given below:

$2(1) = 1_X$ $2(2) = 1_Z$ $2(3) = -1_Y$

C. 3 OR F FRAME

An earth-fixed geometric north-east frame with one axis through the launch point. Prior to platform release this frame is identical to the J frame.

D. M FRAME

Spacecraft-oriented frame:

$M(1) = 1_{Xb}$ $M(2) = 1_{Yb}$ $M(3) = 1_{Zb}$

E. X FRAME

An inertial frame described by two rotations from the I frame such that X(3) is through the polar axis. This frame is used for gravity computations and is established at platform release. X(1) and X(2) are in the equatorial plane such that X(1) has the same longitude as 2(3) prior to platform release.

Launch vehicle coordinates are not directly defined in this simulation although implicit in several equations. The relationship between launch vehicle and spacecraft coordinates is described in Appendix A.

V. TRANSFORMATION MATRICES

A. D3, J-to-I transformation matrix — Defined by angle AZ:

$$\bar{I} = \left[D3(i,j)\right] \bar{J}$$

B. D4, Vehicle-to-I transformation matrix — Defined by platform gimbal angles $\theta b, \psi b, \phi b$:

$$\bar{I} = \left[D4\ (i,j)\right] \bar{M}$$

C. AP, X-to-I transformation matrix — Defined by angles λ and AZ:

$$\bar{I} = \left[AP\ (i,j)\right] \bar{X}$$

D. AJF, F-to-J transformation matrix — Defined by λ and angle corresponding to angular rotation of the earth since platform release:

$$\bar{J} = \left[AJF(i,j)\right] \bar{F}$$

Table D-I

INPUT DATA FORMAT

Card Number	Position Number				
	1	2	3	4	5
1	SLAT	SLONG	—	—	—
2	AZ	DEL = 0	—	X0	Z0
3	SF1 (1)	SF1 (2)	SF1 (3)	ω_A	C (1)
4	C (2)	—	—	Debug	SF (1)
5	SF (2)	CON 4	H (1)	H (2)	—
6	—	—	—	—	—
7	—	—	—	—	—
8	—	—	—	—	—
9	—	E	C3	C4	D5
10	SWT	GC	EZ	EXP P	QPT
11	TIMI	Y0	—	—	—
12	—	—	—	—	—
13	—	Y1 (1)	Y1 (2)	—	—
14	—	—	—	—	X_{INCL}
15	ω_I	ω P	—	—	—
16	—	—	—	—	—
17	—	—	—	—	—
18	SH	Stage 1	—	—	—
19	—	—	—	—	—
20	DFL	DT	ω_E	C5	TU
21	TBECO	TRF	TR	TP1	TP2
22	TGC	GK	CP1	CP2	C1
23	C2	CT (1)	CT (2)	CT (3)	D Time
24	GIP	FP	AR (1)	AR (2)	SFT
25	A3	A4	A5	A6	A7
26	A8	A9	ω_N	—	—
27	—	Q_N	V_F	$V\lambda$	V_g
28	$\bar{q}_F$	C^*	K_Q	V_A	T_E
29	$V\lambda_F$	K_1^2	K_1	t_{cc}	R_F
30	t_K	t_{s1}	G_{epf}	MS2D	—

Table D-II

INITIAL ENVIRONMENT PRINTOUT

Line Number	Position Number					
	1	2	3	4	5	6
1	DL	DDV	Sin AZ	XPHIO	V(1)	V(2)
2	V(3)	Ome Z	D4(1, 1)	D4(2, 1)	D4(3, 1)	D4(1, 2)
3	D4(2, 2)	D4(3, 2)	D4(1, 3)	D4(2, 3)	D4(3, 3)	TFSPR
4	TFPR	R(1)	R(2)	R(3)		

Table D-III

MAIN ENVIRONMENT PRINTOUT

Line Number	Position Number					
	1	2	3	4	5	6
1	Time	W	QX	Thet B	Psi B	Phi B
2	Epsil 1	Epsil 2	Delta 2	Thet D	Yaw D	Roll D
3	TI(1)	TI(2)	TI(3)	VR	ANP	AYP
4	A(1)	A(2)	A(3)	DE	SP	Q
5	F(1)	F(2)	F(3)	FA(1)	FA(2)	FA(3)
6	RA(1)	RA(2)	RA(3)	VA(1)	VA(2)	VA(3)
7	RP(1)	RP(2)	RP(3)	VP(1)	VP(2)	VP(3)
8	SR	CA	CN	CY	XCP	—
*	ϵ R	ϵY	ϵP	Psi B	Phi B	Thet B

*Fast-loop printout not shown in Figure D-1.

Table D-IV

DETAILED MAIN PRINTOUT

Line Number	Position Number					
	1	2	3	4	5	6
1	AT(1)	AT(2)	AT(3)	AG(1)	AG(2)	AG(3)
2	TV(1)	TV(2)	TV(3)	B(1, 1)	B(2, 1)	B(3, 1)
3	B(1, 2)	B(2, 2)	B(3, 2)	B(1, 3)	B(2, 3)	B(3, 3)
4	BT(1, 1)	BT(2, 1)	BT(3, 1)	BT(1, 2)	BT(2, 2)	BT(3, 2)
5	BT(1, 3)	BT(2, 3)	BT(3, 3)	RE(1)	RE(2)	RE(3)
6	R_L	D4(1, 1)	D4(2, 1)	D4(3, 1)	D4(1, 2)	D4(2, 2)
7	D4(2, 3)	D4(3, 3)	D4(2, 3)	D4(3, 3)	D4I(1, 1)	D4I(2, 1)
8	D4I(3, 1)	D4I(1, 2)	D4I(2, 2)	D4I(3, 2)	D4I(1, 3)	D4I(2, 3)
9	D4I(3, 3)	VEI(1)	VEI(2)	VEI(3)	VEM(1)	VEM(2)
10	VEM(3)	TA	F(1)	F(2)	F(3)	D6(1, 1)
11	D6(2, 1)	D6(3, 1)	D6(1, 2)	D6(2, 2)	D6(3, 2)	D6(1, 3)
12	D6(2, 3)	D6(3, 3)	D3(1, 1)	D3(2, 1)	D3(3, 1)	D3(1, 2)
13	D3(2, 2)	D3(2, 3)	D3(1, 3)	D3(2, 3)	D3(3, 3)	D2(1, 1)
14	D2(2, 1)	D2(3, 1)	D2(1, 2)	D2(2, 2)	D2(3, 2)	D2(1, 3)
15	D2(2, 3)	D2(3, 3)	Thet D	Yaw_d	$Roll_d$	Alpha
16	Beta	Theta	PE	EP	DELV(1)	DELV(1)
17	DELV(3)	V1(1)	V1(2)	V1(3)	R1(1)	R1(2)
18	R1(3)	V(1)	V(2)	V(3)	AP(1, 1)	AP(2, 1)
19	AP(3, 1)	AP(1, 2)	AP(2, 2)	AP(3, 2)	AP(1, 3)	AP(2, 3)
20	AP(3, 3)	RX(1)	RX(2)	RX(3)	GXE(1)	GXE(2)
21	GXE(3)	—	—	—	—	—

Table D-V

SYMBOLS FOR GEMINI ASCENT GUIDANCE ENVIRONMENT PROGRAM

(Parentheses indicate maximum dimensions)

Symbol	Nearest Reference Number (Figure D-2)	Definition
A(3)	94 (194)	Total acceleration in I
AADP(800)		Angle of attack for digital plot
AG(3)	0	Gravity acceleration in I
AJF(3, 3)	45	Transformation matrix from F to J
AP(3, 3)	51	Transformation matrix from Equatorial plane to I
AR(4)	3194	Vehicle area constants for drag
AT(3)	9	Thrust acceleration in I
ATT	3195	Angle of attack
ATTY	3195	Angle of attack (side)
AZ	52	J to I azimuth angle (equal $-\gamma$)
B(3, 3)	32	Intermediate terms in Simpson's Integration
BT(3,3)	32	Intermediate terms in Simpson's Integration
C(3)	17	Fuel flow rate
CC2	17	Predicted weight for stage testing
CON4	2	Time increment
CP1	GUID1	Pitch rate from 23.04 to 89.6 sec

Table D-V. Symbols for GEMINI Ascent Guidance Environment Program (cont)

Symbol	Nearest Reference Number (Figure D-2)	Definition
CP2	GUID1	Pitch rate from 89.6 sec to end of 1st stage
CPT	⊥⊥⊥	Limit on initial small interval printouts
CY,CY,CA	3197	Aerodynamic coefficients (side, normal, axial)
C2C	21	No. of increments near cutoff
C5	43	Angle from east to $\bar{I}zb$
C7	4⊥	Constant in geodetic to geocentric conversion
C8	4⊥	Constant in geodetic to geocentric conversion
DDV	43	Longitude difference between launch point and ascending node of orbit plane
DE	8⊥	Air density
DEL	52	J to I elevation angle
DELC	20	Smaller increments initiated (near cutoff)
DELCT⊥	18	Time increment to Stage 1 cutoff
DELCT2	19	Time increment to Stage 2 cutoff
DELO	20	Smaller increment test parameter
DELT	22	One half of the basic time increment
DELTA	221	Basic env. time increment
DELV(3)	33	Intermediate quantity for Simpson's Integration
DL	8	Intermediate guidance angle

Table D-V. Symbols for GEMINI Ascent Guidance Environment Program (cont)

Symbol	Nearest Reference Number (Figure D-2)	Definition
DTR	2	Degrees to radians constant
DW	17	Weight increment
D3(3, 3)	52	J to I transformation matrix
D4(3, 3)	7	Spacecraft to I transformation matrix
E	111	Determines interval between printouts
EXPP	12	For detailed printouts
EZ	23	For early termination of program
FA(3)	3197	Aerodynamic Force components in S/C
F(3)	515 & 33	Thrust acceleration in I
FCY	111	Counter for initial printouts
G(7)	10	Constants in gravity series (not used in simulation)
GC	3197	Gravity constant
H(3)	2	Beginning of stage weights
Ip	112	Index for graph outputs
J	24	Integer for proper cycling in integration loop
JUP	16	Present stage
KDELTA	22	Time Updating integer (to prevent roundoff)
LDELTA	22	Summation of time updating integers

Table D-V. Symbols for GEMINI Ascent Guidance Environment Program (cont)

Symbol	Nearest Reference Number (Figure D-2)	Definition
LC1	133	Used in staging
LC2	136	For initializing Stage 3 quantities
MEND	132	End program parameter
MJF	41	F to J transformation matrix
NC	212	Cutoff counter
NR	214	Counter for new data read in
NSD	214	No. of new sets of data
NC2C	212	Integer value of C2C
PE	8	Atmospheric pressure
PHIB	336	Roll gimbal angle
PSIB	336	Yaw gimbal angle
Q	3194	Dynamic pressure
QN	1	Counter for early stop for initial debugging
QPT	132	Constant for small interval printouts
R(3)	33	Position vector in I
RA(3)	33	Position vector in GEMINI inertial coordinates
RL	51	Launch site radius magnitude
RLJ(3)	51	Launch site vector in J
RM	3191	Magnitude of position vector

Table D-V. Symbols for GEMINI Ascent Guidance Environment Program (cont)

Symbol	Nearest Reference Number (Figure D-2)	Definition
RTD	11	Radians to degrees conversion
RX(3)	291	Position components in equatorial frame
R1(3)	331	Launch position values in I
SH	1	Launch site height above sea level
SLA	3194	Sea level altitude
SLAT		Launch latitude
SLONG	43	Launch longitude
SP	81	Velocity of sound
SR	3194	Velocity ratio (mach no.)
Stage 1	17	Stage 1 cutoff time
Stage 2	181	Stage 2 cutoff weight
t	221	Time from launch
T	81	Thrust
TAS(3)	3197	Thrust acceleration in S/C
TA	111	Main printout counter
TB	3194	Second engine thrust (Stage 1)
THEB	336	Pitch gimbal angle
TIMI	221	Initial holddown time
TTT	22	Floating point value of time parameter

Table D-V. Symbols for GEMINI Ascent Guidance Environment Program (cont)

Symbol	Nearest Reference Number (Figure D-2)	Definition
TV(3)	33	Thrust velocity in I
T1(3)	3194	1st engine thrust components in S/C
T2(3)	3194	2nd engine thrust components in S/C
V(3)	33	Velocity vector in I
VA(3)	33	Velocity vector in GEMINI coordinates
VEI(3)	11	Velocity of air mass in I
VEM(3)	11	Velocity of air mass in spacecraft frame
VI	515	Previous value of velocity
VLFU	51	Initial earth rate velocity in F
VM	11	Velocity magnitude
VR	113	Relative velocity magnitude
V1	33	Previous value of velocity
W	3194	Total present weight of missile
XBP(I)		Pitch attitude error for digital plot
XO,YO,ZO	10	Components of launch site vector
XINC	213	Cutoff time increment
XINCL	43	Inclination angle of desired orbit plane
XPHIO	43	Initial platform roll angle
YI(8)	81	Thrust values for each stage
Z	3194	Height above sea level
λL, ϕL	41	Launch latitude and longitude
ω	51	Earth's rate
Ω	43	Longitude of ascending node of desired orbit plane

Table D-VI

A. Main Program

B. Analog Com.

C. Digital Com.

D. Table Look-Up

E. ATMOD

F. OBLATE

G. RUNKUT

Table D-VI

A. Main Program

```
C      R. E. COFER
C      JAN.  ,1963
       RTIM = CLOCKF(QQQ)
       DIMENSION    V(3),R(3),D3(3,3),D4(3,3),D2(3,3),OME1(3),OME2(3),
C                               AT(3),D4I(3,3),VEI(3),VEM(3),BT(3,3),
C                   B(3,3),TV(3),D6(3,3),TVA(3),VBI(3),RBI(3),DV1(8),
C                   T2(3),C(4),                    G(7),TI(3),A(3) ,RE(3),
C                   TS(3,3),AG(3),DV4(12),BET(3,3),AJF(3,3),
C                   VLJ(3),RLJ(3),OME3(3),TI(3,3),R1(3),DELV(3),
C                    AD(2),YI(8),FF(8),SL(8),TTC(3),H(3),AR(4),
C                   VE(3),RA(3),VA(3),ZP(33),BC(34),DV5(1),TC(6)
       DIMENSION  QZQ(800),XBP(800),AADP(800),V1(3)   ,AVEM(3)
C             ,DV6(3),DV7(15),F(3),VG(3),CT(3),VII(3),DELTV(3),SF(2),SF1
C             (3),FA(3),TAS(3)
C                    ,RX(3),AP(3,3),GX(3),GXE(3)
C             ,DV8(3),S2K(11),S2C(17),VP(3),RP(3),DV9(14)
       COMMON SLAT,SLONG,RL,TGLAT,TGLONG,RT,TB,VBI,RBI,AZ,DEL,SWT,DV1,
C           TIME,ALPHAI,BETTAI,THETAI,TVA,THETE,YAWE,ROLLE,R,AT,A,XO,ZO,
C             THETD,YAWD,ROLLD,UT,WT,S2K,MSC,MVC,S2C,DELT,D4,G,DV4,AG,YO
C           ,VT,  DL,V,CT,TFSPR,TFPR,DTIME,DII,DI2,DI3,
C                 XINCL,DFL,DT,OMEGAE,OMEGAP,VLFU,C5,YO,TU,SS,TT,GK,TBECO
C,TRF,TR,TP1,TP2,TGC,CP1,CP2,C1,C2,G1P,FP,    DV6,JUP,F,VG,
C THEB,PSIB,PHIB,DV7,EXX,EYY,EZZ,
C             EPSIL1,EPSIL2,DELTA2,XCP,FA,SF1,SF,SFT,DELTA,DV8,VP,RP
C             ,DV9,MS2D,DTSCO
       READ INPUT TAPE 5,3,SLAT,SLONG,TGLAT,TGLONG,RT,AZ,DEL,AO  ,XO,ZO,
C                          SF1,OMEGA,C,DEBUG ,SF,CON4,H,G,TI,E,C3,C4,D5,
C                          SWT,GC,EZ,EXPP,QPT,TIMI,YO,BET
C                          ,YI,XINCL,OMEG1,OMEGAP,DELTL,C9,XG,XJ,XK
C                                ,SL(1),CD,TTC,EO,AD,SH,STAGE1,NSD,EOK,
C                            TC,DFL,DT,OMEGAE,C5,TU,TBECO,TRF,TR,TP1,
C                            TP2,TGC,GK,CP1,CP2,C1,C2, CT,DTIME,G1P,FP
C             ,AR(1),AR(2),SFT,S2K,S2C,MS2D
       WRITE OUTPUT TAPE 9,4,SLAT,SLONG,TGLAT,TGLONG,RT,AZ,DEL,AO  ,XO,
C                          ZO,SF1,OMEGA,C,DEBUG ,SF,CON4,H,G,TI,E,C3,C4
C                          ,D5,SWT,GC,EZ,EXPP,QPT,TIMI,YO,BET
C                          ,YI,XINCL,OMEG1,OMEGAP,DELTL,C9,XG,XJ,XK
C                                ,SL(1),CD,TTC,EO,AD,SH,STAGE1,NSD,EOK,
C                            TC,DFL,DT,OMEGAE,C5,TU,TBECO,TRF,TR,TP1,
C                            TP2,TGC,GK,CP1,CP2,C1,C2, CT,DTIME,G1P,FP
C             ,AR(1),AR(2),SFT,S2K,S2C,MS2D
    3  FORMAT (5E14.8)
    4  FORMAT (5E18.8)
       SENSE LIGHT 1
```

```
      CALL TLU (SR,ATTY,ATT,CY,CN,CA,XCP)
      LC1=1
      LC2=1
      C2C =NC2C
      CPT = QPT
      DO 2 I = 3,8
2     SL(I) = 0.
      TB = YI(1)
      W = H(1)
      DELTA= CON4
      KDELTA = 1
      MEND = 0
      JUP = 1
      DTR = .0174532925
      RTD = 57.295780
      NN = 3
      K = 4
      C7 = .0033841003
      EOC = 1.
     IP = 1
     XBP(1) = 0.
     AADP(1) = 0.
      RAD = DTR*SLAT
      SLONG = DTR*SLONG
      TGLAT = DTR*TGLAT
      TGLONG = DTR*TGLONG
      AZ = DTR*AZ
      DEL =DTR*DEL
      AO = DTR*AO
      XINCL = DTR * XINCL
      C8 = (1.1731*DTR)/3600.
      DFL = DTR * DFL
      DDV = DFL
      C5 = DTR*C5
      CP1 = DTR*CP1
      CP2 = DTR*CP2
      SLAT = RAD - C7 * SINF(2. * RAD) + C8 * SINF(4. * RAD)
      SLAT = RAD
      TT = COSF(SLAT)
      SS = SINF(SLAT)
      RL = SQRTF(X0*X0+Y0*Y0+Z0*Z0)
      SLA= RL - SH
      VLFU = RL*OMEGA*TT
      QN = QN +1.
```

```
      IF (QN - DEBUG) 1,23,23
    1 IF (DI3) 43,42,42
   42 CALL GUID1
      GO TO 1
   43 DDV = DDV +     (OMEGAE- OMEGAP)* TFSPR
      AA1 = SINF(XINCL)
      SINDL =-SS*COSF(XINCL) + AA1*TT*SINF(DDV)
      DL = ARSINF(SINDL)
      COSDL = COSF(DL)
      SINAZ = -AA1*COSF(DDV)/COSDL
      AZ = ARSINF(SINAZ)
      XPHIO = C5 +AZ
      VVV = SINF(XPHIO)
      WWW = COSF(XPHIO)
      OME3(1) = 0.
      OME3(2) = OMEGA*TT
      OME3(3) = OMEGA*SS
      Z1= COSF(DEL)
      Z2= SINF(DEL)
      X = COSF(AZ)
      COSAZ = X
      Y = SINAZ
      D3(1,1) = Z1*X
      D3(2,1) = Y
      D3(3,1) =-Z2*X
      D3(1,2) =-Z1*Y
      D3(2,2) = X
      D3(3,2) = Z2*Y
      D3(1,3) = Z2
      D3(2,3) = 0.
      D3(3,3) = Z1
      AP(1,1) = SINAZ*SS
      AP(1,2) =                    COSAZ
      AP(1,3) = -SINAZ*TT
      AP(3,1) = +TT
      AP(3,3) = +SS
      AP(2,1) = -COSAZ*SS
      AP(2,2) =                    SINAZ
      AP(2,3) = COSAZ*TT
   45 VV = COSF(OMEGA*TIM1)
      WW = SINF(OMEGA*TIM1)
      AA = SS*TT
      BB = SS*WW
      DD = TT*WW
```

```
      GG = SS*SS
      HH = TT*TT
      AJF(1,1) = VV
      AJF(2,1) = BB
      AJF(3,1) =-DD
      AJF(1,2) =-BB
      AJF(2,2) = HH+VV*GG
      AJF(3,2) = AA*(1. - VV)
      AJF(1,3) = DD
      AJF(2,3) = AA*(1. - VV)
      AJF(3,3) = GG+HH*VV
      DO 51 I=1,3
      VLJ(I) = AJF(I,1)*VLFU
   51 RLJ(I) = AJF(I,3)*RL
      DO 52 I=1,3
      OME1(I) = 0.
      DO 52 J=1,3
   52 OME1(I) = OME1(I) + AJF(I,J)*OME3(J)
      DO 6  I=1,3
      V11(I) = V(I)
      R(I)=0.
      V(I)=0.
      OME2(I)=0.
      DO 6  J=1,3
      R(I) = R(I) + D3(I,J)*RLJ(J)
      V(I) = V(I) + D3(I,J)*VLJ(J)
    6 OME2(I) = OME2(I)+D3(I,J)*OME1(J)
      DO291  I = 1,3
      RX(I) = 0.
      DO291  J = 1,3
  291 RX(I) = RX(I) + AP(J,I)*R(J)
      CALL OBLATE (RX,GXE,1,1)
      DO292  I = 1,3
      AG(I) = 0.
      DO292  J = 1,3
  292 AG(I) = AG(I) + AP(I,J)*GXE(J)
   44 IF (DI2) 516,39,39
   39 IF (LC12) 515,511,511
  511 LC12 = -2
      DO 518 I = 1,3
  518 F(I) =-AG(I)*DT
      GO TO 513
  516 LC13 = - 2
  515 DO 512 I = 1,3
```

```
512  F(I) = F(I) + (V(I) - VII(I))   -AG(I)*DELTA
513  CALL GUID1
     TIMI = TFPR
     IF (LG13) 514,45,45
514  TIMI = 0
     DO 61 I =1,3
     RI(I) = R(I)
 61  VI(I) = V(I)
     D4(1,1) =  0
     D4(2,1) = 0
     D4(3,1) = 1
     D4(1,2) = + VVV
     D4(2,2) =-WWW
     D4(3,2) =  0
     D4(1,3) = WWW
     D4(2,3) = VVV
     D4(3,3) = 0
     WRITE OUTPUT TAPE 9,13,DL  ,DDV  ,SINAZ,XPHIO,V,OMEZ,D4,TFSPR,
    C TFPR,R
     CALL ATMOD (SH,DE,PE,SP)
     PE1 = PE
 81  T = GC *(YI(JUP)         *2.)/W
     DO 9 I =1,3
  9  AT(I) = T *D4(I,1)
 10  CALL COM
     DO 91  I = 1,3
     RX(I) = 0.
     DO 91  J = 1,3
 91  RX(I) = RX(I) + AP(J,I)*R(J)
     CALL OBLATE (RX,GXE,1,1)
     DO 92  I = 1,3
     AG(I) = 0.
     DO 92  J = 1,3
 92  AG(I) = AG(I) + AP(I,J)*GXE(J)
     DO 94 I=1,3
 94  A (I) = AT(I) + AG(I)
     VEI(1) = V(1) - OME2(2)*R(3)  +OME 2(3)*R(2)
     VEI(2) = V(2) - OME2(3)*R(1) +OME2(1)*R(3)
     VEI(3) = V(3) - OME2(1)*R(2)+OME2(2)*R(1)
     DO 11 I =1,3
     VEM(I)= 0.
     DO 11 J =1,3
 11  VEM(I) = VEM(I) + D4(J,I)*VEI(J)
     DO 113 I =1,3
```

```
113  AVEM(I) = ABSF(VEM(I))
     ATT = ATNIF(AVEM(2),AVEM(1))
     AAD = RTD* ATT
     IF(VEM(2)) 114,115,115
114  ANP = -AAD
     GO TO 1151
115  ANP = AAD
1151 ATTY = ATNIF(AVEM(3),AVEM(1))
     AADY = RTD *ATTY
     IF (VEM(3)) 116,117,117
116  AYP = +AADY
     GO TO 1171
117  AYP =-AADY
1171 VM = SQRTF(V(1)*V(1)+V(2)*V(2)+V(3)*V(3))
     VR = SQRTF(VEM(1)**2 + VEM(2)**2 + VEM(3)**2)
     FCY = FCY + 1.
    IF (FCY - CPT) 12,112,111
111  TA= TA+1.
    IF (TA - E) 133,112,112
112 IP = IP + 1
    AADP(IP) = AAD*9.
     XBP(IP) = ABSF(EYY*RTD*90.)
     QZQ(IP) = ABSF(THETD*RTD*90.)
 12  WRITE OUTPUT TAPE 9,13,TIME,W,QX,THEB,PSIB,PHIB,EPSIL1,EPSIL2,
   C    DELTA2,THETD,YAWD,    ROLLD,T1,VR,ANP,AYP ,A,DE,SP,Q,F,FA,
   C                          RA,VA,RP,VP,SR,CA,CN,CY,XCP
 13  FORMAT (1H /(1H ,6E19.8))
     IF (EXPP -2.) 132,131,131
131  WRITE OUTPUT TAPE 9,37,AT,AG,TV ,B,BT,RF,RL,D4,DAT,VEI,VEM,TA ,
   C                        F,D6,D5,D2,THETD,YAWD,ROLLD,ALPHA,BETTA,
   C                        THETA,PE,EP,DELV,V1,R1,V,AP,RX,GXE
132  TA = 0.
    IF (MEND - 1 ) 133,23,133
133 IF (LC1) 134,135,135
134 JUP = 2
     TB = 0.
     CPT = QPT - 1.
     FCY = 0.
     W = H(2)
    DELCT3 = DELTA - DELCT1
    DELT = DELCT3/2.
    LC1 = 2.
     EO = 0
     EOC = 1.
```

```
      EAK = 0
     GO TO 221
 135  IF (LC2) 136,16,16
 136 JUP = 3
      FCY = 0.
      W = H(3)
     DELCT4 = DELTA - DELCT2
     DELT = DELCT4/2.
     LC2 = 2.
      GO TO 221
  16  GO TO (17,181,20,212),JUP
  17  DW = C(1)*DELTA
      CC2 = TIME + DELTA
      IF (CC2 - STAGE1) 22,18,18
  18 LC1 = -2.
      CPT = 1.E6
      GO TO 22
 181  DW = C(2)*DELTA
      CC2 = W - DW
      IF(CC2 -STAGE2) 19,19,22
  19 LC2 = -2.
      DELCT2 = DELTA*(W -STAGE2)/DW
     DELT = DELCT2/2.
      GO TO 221
  20  DELO = DELTA - DELC
      IF (DELO) 22,21,21
  21  XINC = DELC/C2C
      JUP = 4
      GO TO 12
 212  IF (NC2C- NC)214,214,213
 213  TIME = TIME + XINC
     DELT = XINC/2.
      CPT = 1.E6
      NC = NC + 1
      GO TO 222
 214  IF (NSD -NR )  23, 23,215
 215  READ 3, YI,SL(1)
      NR = NR+1
      EOC = 1.
      GO TO 81
  22  LDELTA = LDELTA + KDELTA
      TTT = FLOATF(LDELTA)
      DELT = DELTA/2.
 221  TIME=TIMI + DELTA*TTT + DELCT1 +DELCT2 + DELCT3 +DELCT4
```

```
222   EZ =EZ +1.
      IF(EZ-8000.) 24,24,23
   23 CALL STOP
      RTIM  = CLOCKF(QQQ)
      CALL PLOT (90.,0.,3.,600 ,3,    XBP,AADP,QZQ)
      CALL EXIT
   24 JL= JL+ 1
      GO TO (31,26,26), JL
   26 GO TO (27,27,29), JL
   27 DO 28 I= 1,3
      R(I) = R(I)+ DELT*V(I) + DELT*B(1,I)/4.
   28 V(I) = V(I) + A(I)*DELT
      GO TO 303
   29 DO 30 I= 1,3
      R(I) = R(I)+ DELT*V(I) + DELT*B(2,I)/4.
   30 V(I) = V(I) + A(I)*DELT
  303 RZZ = R(3)
      VEI(1) = V(1) - OME2(2)*RZZ  +OME2(3)*R(2)
      VEI(2) = V(2) - OME2(3)*R(1) +OME2(1)*RZZ
      VEI(3) = V(3) - OME2(1)*R(2)+OME2(2)*R(1)
      DO301 I =1,3
      VEM(I)= 0.
      DO301 J =1,3
  301 VEM(I) = VEM(I) + D4(J,I)*VEI(J)
      VR = SQRTF(VEM(1)**2 + VEM(2)**2 + VEM(3)**2)
      DO 3195 I=1,3
 3195 AVEM(I) = ABSF(VEM(I))
      ATT = ATN1F(AVEM(2),AVEM(1))
      ATTY = ATN1F(AVEM(3),AVEM(1))
      QT = R(1)
      ZQ = R(2)
      ET = R(3)
D3191 RM = SQRTF(QT**2 +ZQ**2 +ET**2)
      DW = C(JUP) * DELT
 3194 W = W - DW
      Z = RM - SLA
      QX = Z
      CALL ATMOD ( Z,DE,PE,SP)
      VV1 = COSF(EPSIL1)
      WW1 = COSF(EPSIL2)
      DELTA1 = DELTA2
      T1(1) = YI(JUP)*VV1*COSF(DELTA1)
      T1(3) =-YI(JUP)*SINF(EPSIL1)
      T1(2) =-YI(JUP)    *SINF(DELTA1)
```

```
      T2(1) =       TB*WW1*COSF(DELTA2)
      T2(3) = -     TB*SINF(EPSIL2)
      T2(2) =   -   TB    *SINF(DELTA2)
      SR = VR/SP
      CALL TLU (SR,ATTY,ATT,CY,CN,CA,XCP)
      Q =(.5*DE*VR**2)
      FA(1) = -Q*AR(JUP)* CA
      FA(2) = -Q*AR(JUP)* CN
      FA(3) = +Q*AR(JUP)* CY
      IF (VEM(2)) 3192,3196,3196
3192  FA(2) = -FA(2)
3196  IF (VEM(3)) 3199,3198,3198
3198  FA(3) = -FA(3)
3199  DO 3197 I = 1,3
3197 TAS(I)=(FA(I) + T1(I) + T2(I))*GC/W
      CALL RUNKUT
      DO0302 I= 1,3
      AT(I) = 0.
      DO 302 J= 1,3
 302  AT(I) = AT(I) + D4(I,J)* TAS(J)
      DO0191  I = 1,3
      RX(I) = 0.
      DO0191  J = 1,3
 191  RX(I) = RX(I) + AP(J,I)*R(J)
      CALL OBLATE (RX,GXE,1,1)
      DO0192  I = 1,3
      AG(I) = 0.
      DO0192  J = 1,3
 192  AG(I) = AG(I) + AP(I,J)*GXE(J)
      DO0194 I=1,3
 194  A (I) = AT(I) + AG(I)
  31  DO 32 I= 1,3
     BT(JL,I) = AT(I)*DELT*2.
  32 B(JL,I) = A(I)*DELT*2.
      IF(JL-3) 24,321,345
 321  JL= 0
      DO 33 I= 1,3
      DELV(I) = (B(1,I) + 4.*B(2,I) + B(3,I))/6.
      R(I) = R1(I) +(V1(I) + (B(1,I) +B(2,I) + DELV(I))/6.)*DELT*2.
      V(I) =V1(I) + DELV(I)
      DELTV(I) =   (BT(1,I) + 4.*BT(2,I) + BT(3,I))/6.
      TV(I)= TV(I)+DELTV(I)
  33  F(I) = F(I) + DELTV(I)
      RA(1) = R(1)
```

```
      RA(2) = -R(3)
      RA(3) =  R(2)
      VA(1) = V(1)
      VA(2) = -V(3)
      VA(3) =  V(2)
      DO 331 I =1,3
      V1(I) = V(I)
      PSIB = ARSINF(-D4(2,1))
      IF (D4(2,2)) 336,335,336
335   PHIB = 1.5707963
      GO TO 337
336   PHIB = ATN1F (D4(2,3),-D4(2,2))
337   IF (D4(1,1))333,332,333
332   THEB = 1.5707963
      GO TO 334
333   THEB =ATN1F(+D4(3,1),D4(1,1))
334   CALL GUID1
      IF (DTSCO) 339,339,338
338   DELCT2 = DTSCO
      STAGE2 = 500000.
      E = 0.
      EXPP = 2.
339   GO TO 10
 37   FORMAT (6E18.8)
345   PAUSE
      END
```

Table D-VI

B. Analog Com.

```
C       BOB COFER
C       ANALOG SIMULATION
        SUBROUTINE COM
        DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2)
        COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,
      C         EPSIL1,EPSIL2,DELTA2,XCP,FA,SF1,SF,SFT
        IF (LC1) 2,1,1
     1  LC1 = -2
        CALL START
     2  CTM = TIME* SFT
        CEX =EXX * SF1(1)
        CEY =EYY * SF1(1)
        CEZ =EZZ * SF1(1)
        CXCP = XCP * SF1(2)
        CFA2 =-FA(3)*SF1(3)
        CFA3 = FA(2)*SF1(3)
        CALL FIXER (CTM,0,CTM)
        CALL FIXER(CEX,0,CEX)
        CALL FIXER(CEY,0,CEY)
        CALL FIXER(CEZ,0,CEZ)
        CALL FIXER(CXCP,0,CXCP)
        CALL FIXER(CFA2,0,CFA2)
        CALL FIXER(CFA3,0,CFA3)
        CALL DATA (7, 0,CTM,CEX,CEZ,CEY,CXCP,CFA2,CFA3,0)
        CALL WAIT
        CALL DATA (0,7,-7,CEPSIL,CDELT1,CDELT2,CEXR,CEYR,CEZR,CEDR )
        CALL FLTR(CEPSIL,0,CEPSIL)
        CALL FLTR(CDELT1,0,CDELT1)
        CALL FLTR(CDELT2,0,CDELT2)
        CALL FLTR(CEXR,0,CEXR)
        CALL FLTR(CEYR,0,CEYR)
        CALL FLTR(CEZR,0,CEZR)
        CALL FLTR ( CEDR,0, CEDR )
        EPSIL1 =CEPSIL * SF(1)
        EPSIL2 = CDELT1 * SF(1)
        DELTA2 =CDELT2 * SF(1)
        ROLLD = CEXR* SF(2)
        YAWD  = CEZR* SF(1)
        THETD =-CEYR* SF(1)
        CHECK = -CEDR * SF(1)
        WRITE OUTPUT TAPE 9,37, CHECK
    37   FORMAT ( 6E18.8 )
        RETURN
        END
```

Table D-VI

C. Digital Com.

```
C       DIGITAL SIMULATION
        SUBROUTINE COM
        DIMENSION DV1(24),DV2(20),DV3(131),FA(3),SF1(3),SF(2)
        COMMON DV1,TIME,DV2,THETD,YAWD,ROLLD,DV3,EXX,EYY,EZZ,
     C         EPSIL1,EPSIL2,DELTA2,XCP,FA,SF1,SF,SFT,DELTA
        FAST = FA(2)
        FA(2) = -FA(3)
        FA(3) = FAST
        CALL MC
        YSTORE = ROLLD
        ZSTORE = YAWD
        ROLLD = THETD
        YAWD = YSTORE
        THETD = -ZSTORE
        RETURN
        END
```

D. Table Look-Up

```
* DICK WADDING  6059  TABLE LOOK-UP AND INTERPOLATION                    48
C      6059  TABLE LOOK-UP AND INTERPOLATION  DICK WADDING
       SUBROUTINE TLU (X,AY,AN,Z1,Z2,Z3,Z4)
       DIMENSION TX(50),TA1(10),TA2(10),TZ1(50,10),TZ2(50,10),TZ3(50),TZ4
      1(50,10)
       IF (SENSE LIGHT 1) 1,4
 1     READ INPUT TAPE 5,2,NX,NA1,NA2
 2     FORMAT (2I3)
       READ INPUT TAPE 5,3,(TX(I),I=1,NX)
       READ INPUT TAPE 5,3,(TA1(J),J=2,NA1)
       READ INPUT TAPE 5,3,(TA2(J),J=2,NA2)
       READ INPUT TAPE 5,3,((TZ1(I,J),J=2,NA1),I=1,NX)
       READ INPUT TAPE 5,3,((TZ2(I,J),J=2,NA2),I=1,NX)
       READ INPUT TAPE 5,3,(TZ3(I),I=1,NX)
       READ INPUT TAPE 5,3,((TZ4(I,J),J=1,NA2),I=1,NX)
 3     FORMAT (10E7.3)
       TA1(1)=0.
       TA2(1)=0.
       DO 30 I=1,NX
       TZ1(I,1)=0.
 30    TZ2(I,1)=0.
 4     DO 5 I=1,NX
       IF (X-TX(I)) 6,6,5
 5     CONTINUE
        RETURN
 6     DO 7 J=1,NA1
       IF (AY-TA1(J)) 70,70,7
 7     CONTINUE
        RETURN
 70    DO 71 J2=1,NA2
       IF (AN-TA2(J2)) 8,8,71
 71    CONTINUE
        RETURN
 8     F=(X-TX(I-1))/(TX(I)-TX(I-1))
       G=(AY-TA1(J-1))/(TA1(J)-TA1(J-1))
       G2=(AN-TA2(J2-1))/(TA2(J2)-TA2(J2-1))
       ZB1=TZ1(I-1,J-1)+F*(TZ1(I,J-1)-TZ1(I-1,J-1))
       ZT1=TZ1(I-1,J)+F*(TZ1(I,J)-TZ1(I-1,J))
       Z1=ZB1+G*(ZT1-ZB1)
       J=J2
       ZB2=TZ2(I-1,J-1)+F*(TZ2(I,J-1)-TZ2(I-1,J-1))
       ZT2=TZ2(I-1,J)+F*(TZ2(I,J)-TZ2(I-1,J))
       Z2=ZB2+G2*(ZT2-ZB2)
       Z3=TZ3(I-1)+F*(TZ3(I)-TZ3(I-1))
       ZB4=TZ4(I-1,J-1)+F*(TZ4(I,J-1)-TZ4(I-1,J-1))
       ZT4=TZ4(I-1,J)+F*(TZ4(I,J)-TZ4(I-1,J))
       Z4=ZB4+G2*(ZT4-ZB4)
       RETURN
       END
```

Table D-VI

E. ATMOD

```
C       BOB COFER
C       JAN.    ,1963
        SUBROUTINE ATMOD(A,KUN,D,P,SS)
        DIMENSION HB(11),TB(11),PB(11),C(11)
        IF (LC1) 2,1,2
    1   LC1 = 2
        R = 6356.766
        Q = .0341647942
        AA= .14503776E-3
        BB= .194033175E-2
        CC= .014837582
        EE= 3.4838395E-3
        C3 = 2.3025851
        HB(1) = 0.
        HB(2) = 11.
        HB(3) = 25.
        HB(4) = 47.
        HB(5) = 53.
        HB(6) = 79.
        HB(7) = 90.
        HB(8) =105.
        HB(9) =160.
        HB(10)=170.
        HB(11)=200.
        TB(1) = 288.16
        TB(2) = 216.66
        TB(3) = 216.66
        TB(4) = 282.66
        TB(5) = 282.66
        TB(6) = 165.66
        TB(7) = 165.66
        TB(8) = 225.66
        TB(9) = 1325.66
        TB(10)= 1425.66
        TB(11)= 1575.66
        PB(1) = .50057165E1
        PB(2) = .43547194E1
        PB(3) = .33959537E1
```

```
      PB(4) = .2080772BE1
      PB(5) = .17658166E1
      PB(6) = .40772669E-2
      PB(7) =-.98115386
      PB(8) =-.21276979E1
      PB(9) =-.34412957E1
      PB(10)=-.35492009E1
      PB(11)=-.38460691E1
      C(1) =-.0065
      C(2) =0.
      C(3) = .003
      C(4) =0.
      C(5) =-.0045
      C(6) =0.
      C(7) = .004
      C(8) = .02
      C(9) = .01
      C(10)= .005
      C(11)= .0035
    2 GO TO (41,42),KUN
   41 AM = .30480061*A
      GO TO 43
   42 AM = A
   43 Z = .001*AM
      H = (R*Z)/(R+Z)
      IF (H-11.) 3,3,4
    3 TP= 288.16 - 6.5*H
      LEV = 1
      GO TO 26
    4 IF (H-25.) 5,5,6
    5 TP=216.66
      LEV = 2
      GO TO 26
    6 IF (H-47.) 7,7,8
    7 TP = 216.66 + 3.*(H-25.)
      LEV = 3
      GO TO 26
    8 IF (H -53.) 9, 9,11
    9 TP = 282.66
```

```
      LEV = 4
      GO TO 26
11    IF (H - 79.) 12,12,13
12    TP = 282.66 - 4.5* (H-53.)
      LEV = 5
      GO TO 26
13    IF (H - 90.) 14,14,15
14    TP = 165.66
      LEV = 6
      GO TO 26
15    IF (H - 105.) 16,16,17
16    TP = 165.66 + 4.*(H -90.)
      LEV = 7
      GO TO 26
17    IF (H - 160.) 18,18,19
18    TP = 225.66 + 20.*(H - 105.)
      LEV = 8
      GO TO 26
19    IF (H - 170.) 21,21,22
21    TP = 1325.66 + 10.*(H - 160.)
      LEV = 9
      GO TO 26
22    IF (H - 200.) 23,23,25
23    TP = 1425.66 + 5.*(H - 170.)
      LEV = 10
      GO TO 26
25    TP = 1575.66 + 3.5*(H - 200.)
      LEV = 11
26    V = 1000.*(H-HB(LEV))/TB(LEV)
      IF(C(LEV))27,28,27
27    V3 = Q/C(LEV)
      UU = LOGF(1.+C(LEV)*V)/C3
      ALOGP = PB(LEV) - V3*UU
      GO TO 29
28    VV = V*CC
      ALOGP = PB(LEV) - VV
29    PM = 10.**ALOGP
      P    = PM/9.80665
      G   =(EE*PM)/TP
      SS = 20.046433 * SQRTF(TP)
      GO TO (31,32),KUN
31    P = AA*PM
      D = BB*D
      SS = 4.2900833*SS
32    RETURN
      END
```

Table D-VI

F. OBLATE

```
      SUBROUTINE OBLATE (A,B,N,M)
      DIMENSION A(3),B(3)
      X=A(1)
      Y=A(2)
      Z=A(3)
      IF (N-1) 4,3,1
    1 IF (N-2) 4,2,4
    2 X=3.2808333*X
      Y=3.2808333*Y
      Z=3.2808333*Z
    3 IF (M) 4,10,11
   10 AJ2=0.
      GO TO 12
   11 AJ2=.108228E-02
   12 IF (M-1) 13,13,14
   13 AJ3=0.
      GO TO 15
   14 AJ3=-.23E-05
   15 IF (M-2) 16,16,17
   16 AJ4=0.
      GO TO 18
   17 AJ4=-.212E-05
   18 IF (M-3) 19,19,20
   19 AJ5=0.
      GO TO 21
   20 AJ5=-.2E-06
   21 IF (M-4) 22,22,23
   22 AJ6=0.
      GO TO 24
   23 AJ6=.1E-05
   24 IF (M-5) 9,4,4
    9 CK=.1407639E+17
      A=20925640.
      ZJ2=3./2.*AJ2
      ZJ3=5./2.*AJ3
      ZJ4=5./8.*AJ4
      ZJ5=3./8.*AJ5
      ZJ6=1./16.*AJ6
      ZJ10=3./2.*AJ3
      ZJ11=15./8.*AJ5
      RHO2=X**2+Y**2+Z**2
      RHO=SQRTF(RHO2)
```

```
      RHO3=RHO2*RHO
      ZR1=Z/RHO
      ZR2=ZR1**2
      ZR4=ZR2**2
      ZR6=ZR4*ZR2
      AR1=A/RHO
      AR2=AR1**2
      AR3=AR1*AR2
      AR4=AR2**2
      AR5=AR4*AR1
      AR6=AR3**2
      H1=    ZJ2*AR2
      H2=5.*ZR2
      H3=ZJ3*ZR1*AR3
      H4=7.*ZR2
      H5=ZJ4*AR4
      H6=63.*ZR4
      H7=ZJ5*ZR1*AR5
      H8=231.*ZR4
      H9=ZJ6*AR6
      H10=3003.*ZR6
      RHOX=H1*(1.-H2)+H3*(3.-H4)-H5*(3.-42.*ZR2+H6)-H7*(35.-210.*ZR2+H8)
     *+H9*(35.-945.*ZR2+3465.*ZR4-H10)+1.
      RHOZ=H1*(3.-H2)+H3*(6.-H4)-H5*(15.-70.*ZR2+H6)-H7*(105.-315.*ZR2+H
     *8)+H9*(245.-2205.*ZR2+4851.*ZR4-H10)+1.
      RHOZ2=CK/RHO2*(ZJ10*AR3-ZJ11*AR5)
      H11=-CK/RHO3
      FX=H11*X*RHOX
      FY=H11*Y*RHOX
      FZ=H11*Z*RHOZ+RHOZ2
      B(1)=FX
      B(2)=FY
      B(3)=FZ
      IF (N-1) 4,7,5
    5 IF (N-2) 4,6,4
    6 FX=FX/3.2808333
      FY=FY/3.2808333
      FZ=FZ/3.2808333
      X=X/3.2808333
      Y=Y/3.2808333
      Z=Z/3.2808333
      GO TO 7
    4 WRITE OUTPUT TAPE 9,8
    8 FORMAT (1H1,40H ERROR IN ARGUMENTS OF SUBROUTINE OBLATE)
    7 RETURN
      END
```

Table D-VI

G. RUNKUT

```
C       BOB COFER
        SUBROUTINE RUNKUT
        DIMENSION D4(3,3),DV1(45),DV2(32)
        COMMON DV1,THETD,YAWD,ROLLD,DV2,DELT,D4
        DO 4 I = 1,3
        C1 = D4(I,2)*THETD - D4(I,3) * YAWD
        D1 = D4(I,3)*ROLLD - D4(I,1) * THETD
        E1 = D4(I,1)*YAWD  - D4(I,2) * ROLLD
        A = D4(I,1)+C1*.5
        B = D4(I,2)+D1*.5
        C = D4(I,3)+E1*.5
        C2 = B*THETD - C*YAWD
        D2 = C*ROLLD - A*THETD
        E2 = A*YAWD  - B*ROLLD
        D = D4(I,1)+C2*.5
        E = D4(I,2)+D2*.5
        F = D4(I,3)+E2*.5
        C6 = E*THETD - F*YAWD
        D6 = F*ROLLD - D*THETD
        E6 = D*YAWD  - E*ROLLD
        R = D4(I,1) + C6
        S = D4(I,2) + D6
        T = D4(I,3) + E6
        C7 = S*THETD - T*YAWD
        D7 = T*ROLLD - R*THETD
        E7 = R*YAWD  - S*ROLLD
        D4(I,1) = D4(I,1) + (C1+C2*2. +C6*2. +C7)*DELT/6.
        D4(I,2) = D4(I,2) + (D1+D2*2. +D6*2. +D7)*DELT/6.
    4   D4(I,3) = D4(I,3) + (E1+E2*2. +E6*2. + E7)*DELT/6.
        RETURN
        END
```

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 (γ) and the angle between the orbit plane and the launch site (δ). The initial conditions on the spacecraft velocity along platform axis (A117) and the initial platform roll gimbal angle (ϕ_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."

62-5640020 (1)

SLOW LOOP (2 SAMPLES/ SEC)

FROM EXEC

A101 LC 4B

A102
$W_N = W_{NO}$ $\delta\Psi_N = \delta\Theta_N = \delta\Phi_N = 0$ ASCENT INITIALIZATION
$T_G = 175$ $\Delta\Psi_{LV} = \Delta\Phi_{LV} = \Delta\Theta_{LV} = 0$
$V_{Rf} = 0$ $\theta_0' = \psi_0' = 0$ LC 4A, 4C, 4D, 4E, 4F+
$\delta I = 0$ LC 4B-
$V_g = V_{go}$ LC 18, 20+
$V_A = V_{AO}$ $\dot{Y}_{Li} = 0$ LC 21, 24, +
$X = X_o$
$\bar{a}_f = \bar{a}_{fo}$ $Y = Y_o$ LC 29+
$\delta T_U = 0$ $Z = Z_o$ $T_U = 100$
$Q_N = Q_o$ $\dot{\Phi}_N = \dot{\Theta}_N = \dot{\Psi}_N = \eta_X = t_1 = \Delta t_s = \Delta t_s = 0$
$a_f = \bar{a}_{fo}$
$t_R = t_{STZ}$ $\Theta_N = 90°$

TO X 102

A103 LC 29 − TO A239

A104 LC 18

A105 LC 20

A106 READ T_E

A107 $T_E > 0$ YES / NO

A119 $T_{EO} = T_E$, $\iota = KAI$

A109 $t = 0$

A108 COMPUTE COS i, SIN i

A110 COMPUTER RUNNING ON DO-05 −

A120 READ T_E

A121 DAS SUBROUTINE

A122 $T_E > T_{EO}$ YES / NO

A123 $i = i - 1$ ≠0 / 0

A115

A130 $\Delta t_2 = \Delta t$, $t^* = T_E + t_B$, LC 20 −

X105

A111 ACCELEROMETER SUBROUTINE (F_x, F_y, F_z)

A113 LC 18

A114 PLATFORM RELEASED DI 36 (+)YES / NO (−)

A115 AGE SIMULATION DI 24 TO Y A130 (−) / N(+)

A116 $\eta_Y = \text{SIN } \delta$, $\Phi_N = C_5 - \gamma + 90°$

A117 $\cos\gamma_o = \cos\gamma$, $V_{xi} = C_o \cos\gamma$, $V_{zi} = -C_o \sin\gamma$, $\dot{X}_o = V_{xi}$, $\dot{Z}_o = V_{zi}$ TO A148

A118 $\Phi_N = \Phi_N + K_1'(\cos^2\gamma_o)^{-1}\cos^2\gamma(\Delta t)$
$\Theta_N = \Theta_N + K_2' \cos\gamma\,(\Delta t)$
$\Psi_N = \Psi_N + K_3' \text{ SIN } \gamma(\Delta t)$

A112 $\Phi\cdot\Omega = (\Phi\cdot\Omega) + \Delta t\,(\Omega_E - W')$
$\cos(\Phi\cdot\Omega)$, $SIN(\Phi\cdot\Omega)$
$\sin\gamma = +\sin i \cos(\Phi\cdot\Omega)$
$\gamma = \text{SIN } \gamma \left(1 + \frac{\text{SIN}^2\gamma}{6}\right)$
$\sin\delta = -a_1 \cos i + a_2 \sin i \; SIN\,(\Phi\cdot\Omega)$
$\cos\gamma = 1 - \frac{\text{SIN}^2\gamma}{2}$

A124 $t = t^*$

A125 $t_{RF} = C_6$
$\Delta\Phi_A = K_5' \text{ SINS} + \text{SIN } \delta(K_7' + K_8' \text{SIN } \gamma) - K_9'$
$\Delta T_R = .8(C_5 - \gamma - \Delta\Phi_A)$
$\Phi_P = \Phi_N - 1.25\,\Delta T_R$
$t_R = C_6 - \Delta T_R$

A126 LC 18 −

A132 $\Delta t = \Delta t + \Delta t_s$, $\Delta t_s = 0$

A133 $R^2 = X^2 + Y^2 + Z^2$, $R = \sqrt{R^2}$, $g = K/R^2$, $K_g = g/R$, $g_X = K_g X$, $g_Y = K_g Y$, $g_Z = K_g Z$

A134 $V_{x(i-1)} = V_{xi}$, $V_{y(i-1)} = V_{yi}$, $V_{z(i-1)} = V_{zi}$

A135 $V_{xi} = V_{x(i-1)} + F_x - g_x \Delta t$
$V_{yi} = V_{y(i-1)} + F_y - g_y \Delta t$
$V_{zi} = V_{z(i-1)} + F_z - g_z \Delta t$

A136 $X = X + (V_{xi} + V_{x(i-1)})\frac{\Delta t}{2}$
$Y = Y + (V_{yi} + V_{y(i-1)})\frac{\Delta t}{2}$
$Z = Z + (V_{zi} + V_{z(i-1)})\frac{\Delta t}{2}$

A137 $t_1 = t_1 + \Delta t$

A138 LC 21 SET-BY DCS WHEN UP DATE COMMAND RECEIVED

A139 LC 21 +

A140 $V_Z' = \eta_X V_{XG} + V_{ZG}$

A142 $T_U - t$ − / +

A127 $\delta t = \frac{T_U - t_2}{\Delta t}$

A144 $t_2 = t$

A128 $V_{xup} = V_x(i-1) + \delta t\,(V_{xi} - V_{x(i-1)})$
$V_{zup} = V_z(i-1) + \delta t\,(V_{zi} - V_{z(i-1)})$
$V_{yup} = V_y(i-1) + \delta t\,(V_{yi} - V_{y(i-1)})$

A129 $T_U = T_U + 50$

A141 $251 - T_U$ +

A143 $301 - T_U$ + / −

A145 $T_U = T_U - 40$

A146 $\Delta\eta = \frac{V_{ZUP} - V_Z'}{V_{XUP} - \dot{X}_o}$

A147 $V_{yi} = V_{yi} + (V_{YG} - V_{YUP})$
$V_{zi} = V_{zi} + (V_z' - V_{ZUP})$
$V_{xL} = V_{xL} + (V_{XG} - V_{XUP})$

FROM A226 & 280

A153 COMPUTE SIN & COS Φ_b, Ψ_b, Θ_b

A154
$N_{20} = \text{SIN } \Psi_b$
$N_{21} = \cos\Psi_b \cos\Phi_b$
$N_{22} = \text{SIN } \Phi_b$
$N_{23} = -\cos\Psi_b \sin\Phi_b$
$N_{24} = \cos\Phi_b$
$N_{25} = \cos\Psi_b \sin\Theta_b$
$N_{26} = \cos\Psi_b \cos\Theta_b$

A148 $\Phi_{FL} = \Phi_N$, $\Theta_{FL} = \Theta_N$, $\Psi_{FL} = \Psi_N$

A149 $t_{STZ} - t$ − TO A 202

A155 $t_{STZ} - t$ + / − A227

A156 $t_R - t$ −

A157 $t_{RF} - t$ +

$\dot{\Phi}_N = -1.25°/s$
$\Phi_N = \Phi_N + \dot{\Phi}_N$

A150 $\eta_X = \eta_X + \Delta\eta$

A151 $V_{xi} = V_{xi} + \Delta\dot{x}$
$V_{zi} = V_{zi} + \Delta\dot{z}$
$X = X + \Delta\dot{x}(t_1)$
$Z = Z + \Delta\dot{Z}(t_1)$

A152 $\Delta\dot{x} = -\Delta\eta\,\dot{Z}_o$
$\Delta\dot{Z} = +\Delta\eta\,\dot{x}_o$

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

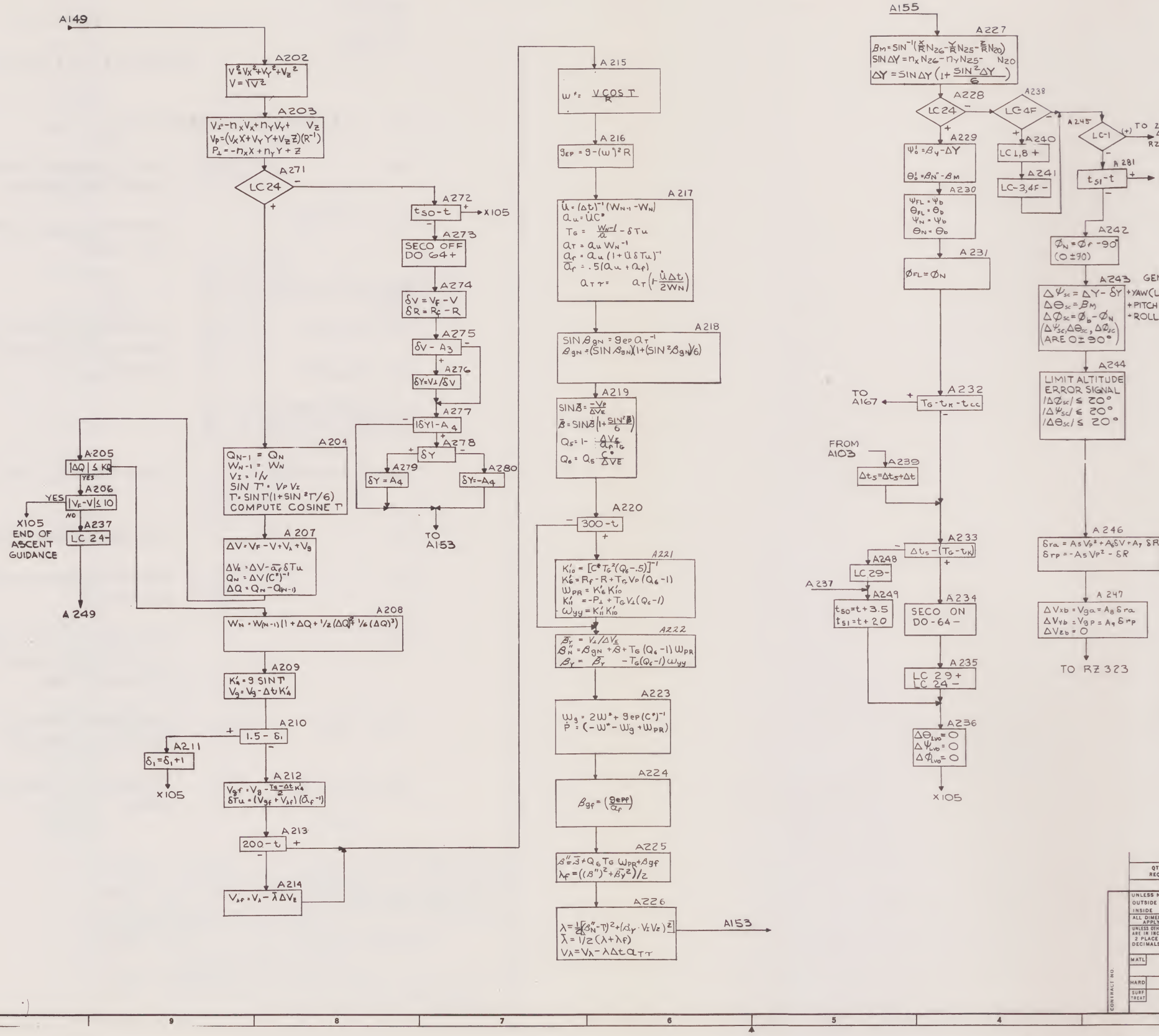

A149
A202
$V_x^2+V_y^2+V_z^2$
$V=\sqrt{V^2}$
A203
$V_z=-n_xV_x+n_yV_y+V_z$
$V_p=(V_xX+V_YY+V_zZ)(R^{-1})$
$P_\perp=-n_xX+n_yY+Z$
A271
LC 24
A272
$t_{so}-t$
X105
A273
SECO OFF
DO 64+
A274
$\delta V=V_F-V$
$\delta R=R_c-R$
A275
$\delta V-A_3$
A276
$\delta Y=V_z/\delta V$
A277
$|\delta Y|-A_4$
A278
δY
A279
$\delta Y=A_4$
A280
$\delta Y=-A_4$
TO A153
A204
$Q_{N-1}=Q_N$
$W_{N-1}=W_N$
$V_I=1/V$
SIN T' = V_P/V_Z
T' = SIN T'(1+SIN²T'/6)
COMPUTE COSINE T'
A205
$|\Delta Q|\leq K_Q$
YES
A206
$|V_E-V|\leq 10$
YES
NO
X105
END OF
ASCENT
GUIDANCE
A237
LC 24-
A249
A207
$\Delta V=V_F-V+V_A+V_g$
$\Delta V_E=\Delta V-\bar{a}_f\delta T_u$
$Q_N=\Delta V(C^*)^{-1}$
$\Delta Q=Q_N-Q_{N-1}$
A208
$W_N=W_{N-1}(1+\Delta Q+\frac{1}{2}(\Delta Q)^2+\frac{1}{6}(\Delta Q)^3)$
A209
$K_4'=g\sin T$
$V_g=V_g-\Delta t K_4'$
A210
$1.5-\delta_1$
A211
$\delta_1=\delta_1+1$
X105
A212
$V_{gf}=V_g-\frac{T_G-\Delta t}{2}K_4'$
$\delta T_u=(V_{gf}+V_{Af})(\bar{\alpha}_f^{-1})$
A213
$200-t$
A214
$V_{Af}=V_A-\bar{\lambda}\Delta V_E$
A215
$w'=\frac{V\cos T}{R}$
A216
$g_{ep}=g-(w')^2R$
A217
$\dot{U}=(\Delta t)^{-1}(W_{N-1}-W_N)$
$a_u=\dot{U}C^*$
$T_G=\frac{W_{N-1}}{a_u}-\delta T_u$
$a_T=a_uW_N^{-1}$
$a_f=a_u(1+\frac{1}{2}\dot{U}\delta T_u)^{-1}$
$\bar{a}_f=.5(a_u+a_f)$
$a_{TT}=a_T\left(1-\frac{\dot{U}\Delta t}{2W_N}\right)$
A218
$\sin\beta_{gN}=g_{ep}a_T^{-1}$
$\beta_{gN}=(\sin\beta_{gN})(1+(\sin^2\beta_{gN})/6)$
A219
$\sin\bar{\beta}=\frac{-V_E}{\Delta V_E}$
$\bar{\beta}=\sin\bar{\beta}\left(1+\frac{\sin^2\bar{\beta}}{6}\right)$
$Q_5=1-\frac{\Delta V_E}{a_fT_G}$
$Q_6=Q_5\frac{C^*}{\Delta V_E}$
A220
$300-t$
A221
$K_{10}'=[C^*T_G^2(Q_6-.5)]^{-1}$
$K_6'=R_f-R+T_GV_P(Q_6-1)$
$W_{PR}=K_6'K_{10}'$
$K_{11}'=-P_\perp+T_GV_\perp(Q_6-1)$
$\omega_{yy}=K_{11}'K_{10}$
A222
$\beta_Y=V_\perp/\Delta V_E$
$\beta_N''=\beta_{gN}+\beta+T_G(Q_6-1)W_{PR}$
$\beta_Y=\beta_Y-T_G(Q_6-1)\omega_{yy}$
A223
$W_g=2W^*+g_{ep}(C^*)^{-1}$
$\dot{P}=(-W^*-W_g+W_{PR})$
A224
$\beta_{gf}=\left(\frac{g_{ep}P}{a_f}\right)$
A225
$\beta''=\bar{\beta}+Q_6T_G\omega_{PR}+\beta_{gf}$
$\lambda_f=((\beta'')^2+\bar{\beta}_Y^2)/2$
A226
$\lambda=\frac{1}{\Delta t}[(\beta_N''-T)^2+(\beta_Y\cdot V_\perp V_Z)^2]$
$\bar{\lambda}=1/2(\lambda+\lambda_f)$
$V_\lambda=V_\lambda-\lambda\Delta t a_{TT}$
A153
A155
A227
$\beta_M=\sin^{-1}(\frac{X}{R}N_{26}-\frac{Y}{R}N_{25}-\frac{Z}{R}N_{20})$
$\sin\Delta\gamma=n_xN_{26}-n_yN_{25}-N_{20}$
$\Delta\gamma=\sin\Delta\gamma\left(1+\frac{\sin^2\Delta\gamma}{6}\right)$
A228
LC 24
A238
LC 4F
A245
LC-1
TO Z RZ
A229
$\psi_0'=\beta_y-\Delta\gamma$
$\theta_0'=\beta_N''-\beta_M$
A240
LC 1,8 +
A241
LC-3,4F-
A281
$t_{s1}-t$
A230
$\psi_{FL}=\psi_0$
$\theta_{FL}=\theta_0$
$\psi_N=\psi_0$
$\theta_N=\theta_0$
A242
$\phi_N=\phi_F-90$
(0 ± 90)
A231
$\phi_{FL}=\phi_N$
A243
GE
$\Delta\psi_{sc}=\Delta\gamma-\delta\gamma$
$\Delta\theta_{sc}=\beta_M$
$\Delta\phi_{sc}=\phi_0-\phi_N$
$(\Delta\psi_{sc},\Delta\theta_{sc},\Delta\phi_{sc}$
ARE $0\pm 90°$)
+YAW(L
+PITCH
+ROLL
A244
LIMIT ALTITUDE
ERROR SIGNAL
$|\Delta\phi_{sc}|\leq 20°$
$|\Delta\psi_{sc}|\leq 20°$
$|\Delta\theta_{sc}|\leq 20°$
TO A167
A232
$T_G-t_R-t_{cc}$
FROM A103
A239
$\Delta t_s=\Delta t_s+\Delta t$
A233
$\Delta t_s-(T_G-t_R)$
A248
LC 29-
A237
A249
$t_{so}=t+3.5$
$t_{s1}=t+2.0$
A234
SECO ON
DO-64-
A235
LC 29+
LC 24-
A236
$\Delta\theta_{LVD}=0$
$\Delta\psi_{LVD}=0$
$\Delta\phi_{LVD}=0$
X105
A246
$\delta r_a=A_5V_p^2+A_6\delta V+A_7\delta R$
$\delta r_p=-A_5V_p^2-\delta R$
A247
$\Delta V_{xb}=V_{ga}=A_8\delta r_a$
$\Delta V_{yb}=V_{gp}=A_9\delta r_p$
$\Delta V_{zb}=0$
TO RZ 323

Figure E-1B.

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_λ	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 λ, where λ is latitude of launch point.
a_2	Cosine λ, where λ 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)

$\overline{\beta}$	Thrust attitude required to compensate for vehicle radial velocity.
β''	Final value of commanded thrust attitude.
β_M	Computed value of vehicle pitch attitude. Actual vehicle pitch attitude with respect to local horizontal.
β_Y	Commanded vehicle yaw angle (with respect to orbit plane). Includes explicit yaw steering.
$\overline{\beta}_Y$	Vehicle yaw angle (with respect to orbit plane) required to kill velocity perpendicular to the orbit plane.
β_{gN}	Thrust attitude required to compensate for effective gravity.
β_{gf}	Final value of thrust attitude required for gravity compensation.
β''_N	Commanded vehicle thrust attitude with respect to local horizontal.
Γ	Vehicle flight path angle with respect to local horizontal.
Δt	Length of slow loop computation cycle.
Δt_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)

ΔT_R	Time vehicle will roll at a constant rate to reach the proper azimuth orientation.
$\Delta t'$	Fast-loop cycle time (50 msec).
ΔV	Total velocity to be gained prior to nominal shutdown including approximated velocity loss due to gravity and steering.
ΔV_E	Velocity to be gained corrected for actual shutdown time ($V_e = \int_t^{\text{cutoff}} a_T$)
ΔY	Computed value of vehicle yaw attitude. Angle between vehicle X-axis and orbit plane.
θ, ϕ, ψ	Refer to pitch, roll, and yaw, respectively.
$\Delta\theta_{LV}, \Delta\phi_{LV}, \Delta\psi_{LV}$	Computed vehicle attitude errors.
$\Delta\theta_{LVO}, \Delta\phi_{LVO}, \Delta\psi_{LVO}$	Limited vehicle attitude errors delivered to autopilot.
$\Delta\theta_{SC}, \Delta\phi_{SC}, \Delta\psi_{SC}$	Computed spacecraft attitude errors during insertion velocity adjust.
$\Delta\theta_{SCO}, \Delta\phi_{SCO}, \Delta\psi_{SCO}$	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.
θ'_o	Vehicle pitch attitude error quantity.
θ_b, ϕ_b, ψ_b	Measured gimbal angles.
θ_N, ϕ_N, ψ_N	Commanded platform gimbal angles during Stage 1. During Stage 2, θ_N and ψ_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.
λ	Coefficient used in computation of steering loss.
$\bar{\lambda}$	Average value of steering loss coefficient between now and shutdown.
λ_f	Final value of steering loss coefficient at time of shutdown.
ϕ	Longitude of vehicle with respect to Greenwich.
ψ_o'	Vehicle yaw attitude error quantity.
Ω	Longitude of ascending node of Agena orbit.
Ω_E	Rate of earth rotation.

D. GREEK (SMALL)

γ	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. γ is positive when X is displaced north.
γ_o	Value of γ at the time of platform release.
δ	Represents the angle between the launch site and the orbit plane; positive when vehicle is below orbit plane.
δ_Y	Spacecraft yaw angle required to kill velocity perpendicular to the orbit plane.
δ_1	Quantity used to allow convergence of certain Stage 1 computations upon initiation of Stage 2 guidance.
δ_R	Position increment above or below nominal insertion altitude.

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

δt	Computed quantity which is used in the linear interpolation of velocity data. Velocity data is corrected to update time.
δT_u	Computed quantity representing the adjustment to nominal Stage 1 engine shutdown time.
δV	Velocity increment above or below nominal insertion velocity.
δr_a	Total computed position increment above or below apogee.
δr_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.
η_X, η_Y, η_Z	Matrix coefficients used to obtain platform components perpendicular to the orbit plane.
ω'	Nodal precession rate of the orbit plane.
ω^*	Pitch rate term used to keep thrust attitude constant with respect to local vertical.
ωPr	Pitch rate term used to satisfy vehicle altitude constraint.
ω_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 (ϕ_N) will be set equal to the value computed for the final roll gimbal angle (ϕ_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 (β_M) and actual yaw attitude with respect to the orbit plane (ΔY) will be computed in A227. The quantities will be used in A229 to determine the pitch (θ'_0) and yaw (ψ'_0) attitude errors of the vehicle. θ'_0 and ψ'_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 ΔQ (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, δV, and δR) 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 ΔV 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 ΔV 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.

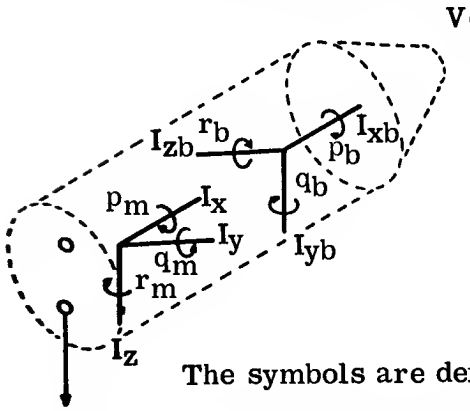
G. SWITCHOVER FADE-IN

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:

$$p_b = p_m \qquad \omega_{xb} = \omega_{xm}$$

$$q_b = r_m \qquad \omega_{yb} = \omega_{zm}$$

$$r_b = -q_m \qquad \omega_{zb} = -\omega_{ym}$$

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 τ, 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

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

Δp_m = Launch vehicle roll change in time τ,

Δq_m = Launch vehicle pitch change in time τ,

and

Δr_m = Launch vehicle yaw change in time τ.

Also

$\Delta\theta$ = Change in θ in time τ,

$\Delta\psi$ = Change in ψ in time τ,

and

$\Delta\phi$ = Change in ϕ in time τ.

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

θ_b, ψ_b, ϕ_b = present gimbal angles

θ_N, ψ_N, ϕ_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, Δp_m, Δq_m, and Δr_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

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

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

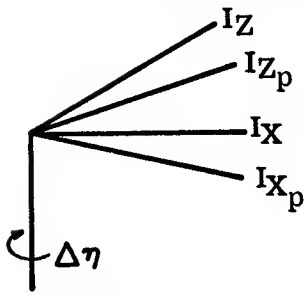
$$\dot{\psi} = -\dot{P}\cos\phi,$$

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 explicity.

Now

$$V_X = V_{oX} + \int_o^t a_X \, dt,$$

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

and

$$V_Z = V_{oZ} + \int_o^t a_Z \, dt.$$

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

V_{oX} and V_{oZ} = calculated velocity due to earth's rotation in the desired frame.

With a misoriented platform,

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

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

and

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

Here, V_{oX} and V_{oZ} are values inserted during initialization.

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

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

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

and

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

Now

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

or

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

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.

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

$$V_{Yp}' = V_Y$$

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

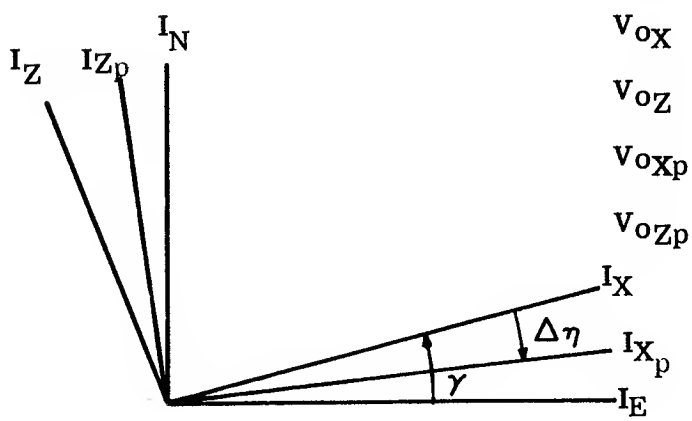
Future updates now require that the primed values denoting ground measured velocity be transformed to calculated platform axes. Since η_X is originally assumed equal to zero, the following relationships are valid for the general case of angle update:

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

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

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

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

$$V_{oXp} = C_o \cos \gamma + C_o \sin \gamma \Delta\eta = V_{oX} - V_{oZ} \Delta\eta ,$$

and

$$V_{oZp} = C_o \sin \gamma + C_o \cos \gamma \Delta\eta = V_{oZ} + V_{oX} \Delta\eta .$$

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

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

and

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

Positions X and Z must also be updated. So,

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

and

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

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:

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

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

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

and

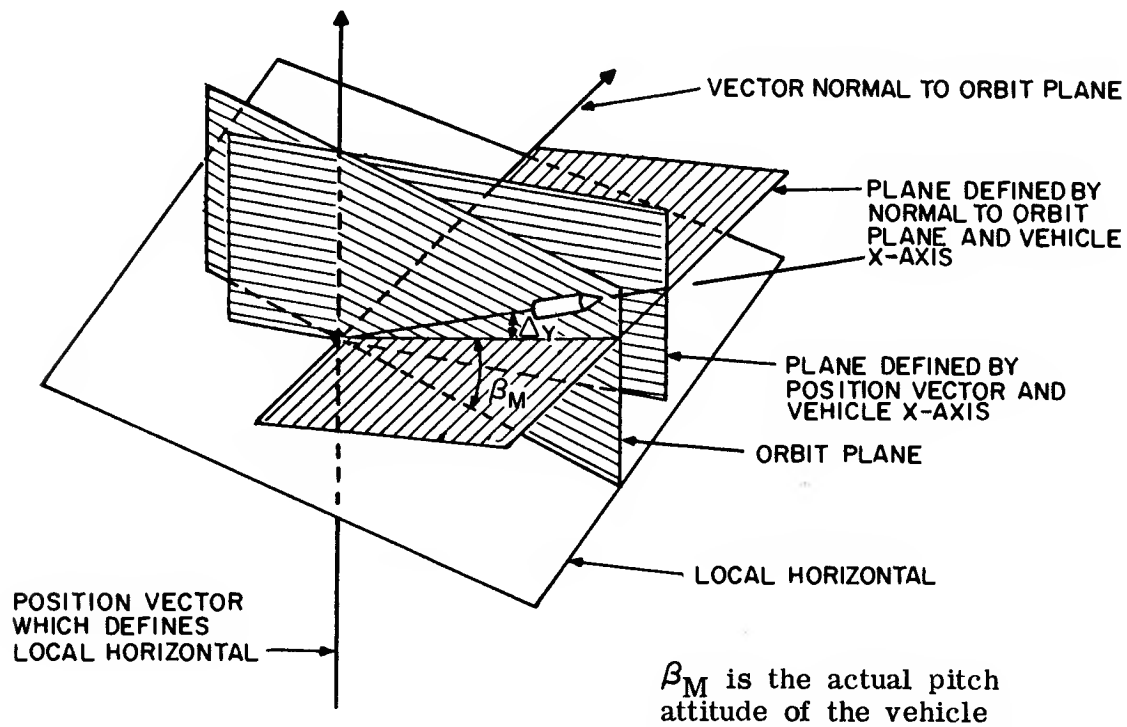
$$\sin\beta_M = \frac{X}{R} \cos\psi_b \cos\theta_b - \frac{Y}{R} \cos\psi_b \sin\theta_b - \frac{Z}{R} \sin\psi_b$$

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

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

By using a similar equation,

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



β_M is the actual pitch attitude of the vehicle with respect to the local horizontal.

ΔY 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)
ω_N	52K (8)
QN	52C (1)
V_P	52C (2)
V_λ	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_4^2	52C (11)
K_1^{12}	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
θ_B	THEB	E
ψ_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
η_x	ETX	1

Symbol	Env. Fortran Symbol	* From
η_y	ETY	1
η_z	ETZ	1
g	GT	1
N25	A25	1
N26	A25	1
Δt_S	DTS	2
LC29	LC29	2
Δt_{SECO}	—	2
$\dot{P}$	P1	2
ϕ_f	PHIF	1
LCA232	LCA232	2
R	RM	1
δ_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 VP0(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),FP0(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.
```

```
      VP0(1)= C0*CGM
      VP0(3)=-C0*SGM
      VP(1) = VP0(1)
      VP(3) = VP0(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*VP0(3)
    DVP(3)= -DPSI*VP0(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 DO57 = -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     DO64=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   PP0=BY-DT
      TP0=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 (MS20) 201,202,202
201   RETURN
202   WRITE OUTPUT TAPE 9,203,LCA232,LC29,LC24,LC1,LC3,LC8,LC4F,VN,VY,VP
     1,R1,DO64,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
```

Appendix F

DATA CONSTANTS FOR GEMINI ASCENT GUIDANCE SIMULATION

Appendix F

DATA CONSTANTS FOR GEMINI ASCENT GUIDANCE SIMULATION

Table F-I is a list of constants used for the GEMINI Ascent Guidance Simulation; Table F-II lists the initiatization constants for Stage 2 Guidance.

Table F-I

DATA CONSTANTS

Parameter	Value	Footnote
Launch latitude	28.46 deg. N	—
Launch longitude	83.55 deg. W	1
Gravitational constant	32.174 ft/sec^2	2
Launch point radius magnitude	20.909749×10^6 ft	3,7
Earth's equatorial radius	$20,925,640 \times 10^6$ ft	4
Oblateness, 2nd harmonic	0.108228×10^{-2}	4
Orbital precession rate	0.145×10^{-5} radians/sec.	—
Earth's rotation rate	0.7292115×10^{-4} radians/ sec.	—
Vehicle initial weight, Stage 1	329,000 lbs	—
Vehicle initial weight, Stage 2	71,500 lbs	—
Fuel flow rate, Stage 1	1662.1622 lbs/sec.	—
Fuel flow rate, Stage 2	325.41899 lbs/sec.	—
Burning time, Stage 1	148 sec.	5
Engine thrust, Stage 1	215,000 lbs	—
Engine thrust, Stage 2	100,000 lbs	—
Vehicle frontal area	78.5 ft^2	—
Initial orientation of vehicle - East to -y	94 deg.	—
Orbital inclination	28.76 deg.	—
Central angle between launch longitude and longitude of ascending node at $t = t_{ref}$	94 deg.	—
Time from t ref to platform release	2 sec.	—
Angle between platform Z-axis and normal to orbit plane	0.2407797 deg.	6

Table F-I. Data Constants (cont)

Parameter	Value	Footnote
Angle between platform X-axis and east	-1.93737 deg.	6
Time from platform release to launch	6 sec.	—
Time of end of roll program	20.48 sec. (20.5)	7
Time of initiation of pitch program	23.04 sec. (23.5)	7
Time of pitch rate change	89.6 sec. (90.0)	7
Time of autopilot gain change	105.0 sec.	7
Pitch rate No. 1	-0.7727 deg./sec.	7
Pitch rate No. 2	-0.32 deg./sec.	7
Gravitational constant	$0.14083914 \times 10^{17}$	7,3
Time of change from GUID1 to GUID2	156.04 sec. (156.5)	7
GEMINI Computer Comp. cycle time	0.5 sec.	8
GEMINI Computer Fast-loop Comp. cycle time	0.05 sec.	7,9

Notes:

1. True value 80.55 deg. W
2. Value used to convert fuel weight to mass — not used in gravitational acceleration determination
3. Derived from Fisher Ellipsoid evaluated at latitude of launch point
4. Defined in subroutine oblate
5. Specified in lieu of Stage 1 final weight for determination of Stage 1 cutoff
6. Computed result from initial conditions
7. Value in GEMINI Computer — time values go to next 0.5 sec. value
8. Estimated — chosen for ease of programming
9. Estimated — equal to 0.1 of comp. cycle time

Table F-2

STAGE 2 INITIALIZATION CONSTANTS*

Parameter	Value
V_λ	150 fps
V_g	750 fps
V_F	25,765.9 fps
STu	0
KQ	0
C*	10,000
T_G	180 sec.
$\bar{a}_f$	250 ft/sec.^{2}
Q_N	1.66
R_f	21,411,962 ft
$V_{\lambda f}$	0
t_K	0
t_{cc}	0
W_N	5.2593
g_{epf}	-0.56 fps^2

*Symbols defined in Appendix E

Appendix G

VEHICLE DATA FOR GEMINI ASCENT GUIDANCE SIMULATION

Appendix G

VEHICLE DATA FOR GEMINI ASCENT GUIDANCE SIMULATION

The data in this appendix represents the composite vehicle model used for the Ascent Guidance Simulation. This data is not identical to the latest compatibility information because of the preliminary nature of the data available during the initial stages of the simulation and the difficulty of incorporating the latest data into the program.

Table G-I summarizes the vehicle dimensions used in the program; Figures G-1 through G-6 present aerodynamic and weight-related parameters.

Table G-1

VEHICLE DIMENSIONS

Station of engines - Stage 1	1274.121	inches
Station of engines - Stage 2	500.00	inches
Distance from vehicle longitudinal axis to engines - Stage 1	29.07	inches
Distance from vehicle longitudinal axis to roll engine - Stage 2	44.00	inches

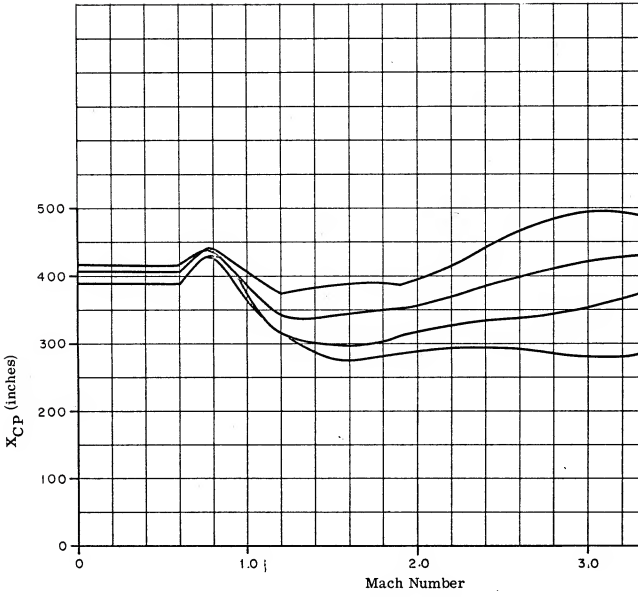


Figure G-1. Center of Pressure Location vs. Mach Number

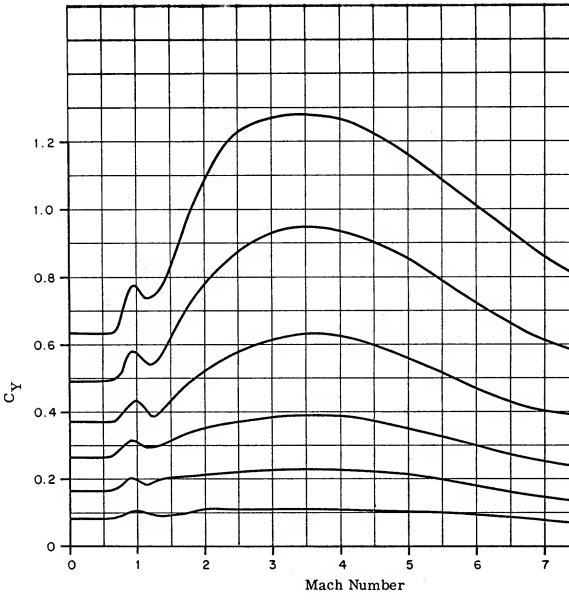


Figure G-2. Side Force Coefficient vs. Mach Numbe

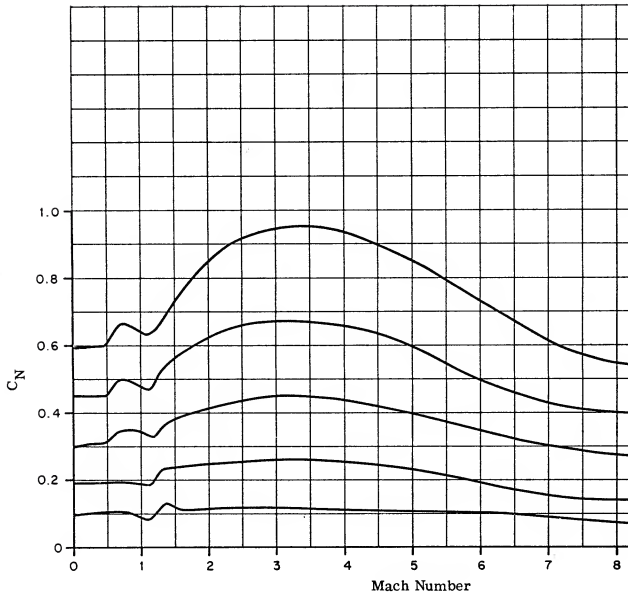


Figure G-3. Normal Force Coefficient vs. Mach Number

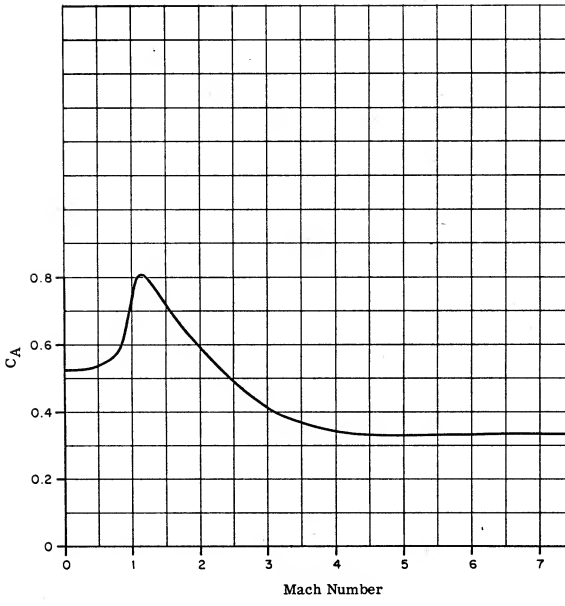


Figure G-4. Axial Force Coefficient vs. Mach Numbe

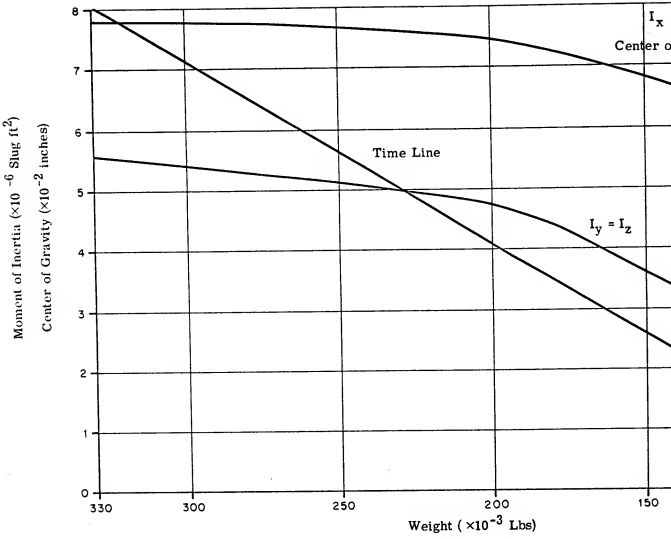


Figure G-5. Moment of Inertia and Center of Gravity vs. Vehicle Weight -

Moment of Inertia ($\times 10^{-3}$ Slug ft)
Center of Gravity (inches)
140
130
120
110
100
90
80
70
60
350
340
330
320
310
300
290
280
270
80
70
60
50
40
30
20
Weight ($\times 10^{-3}$ Lbs)
Center of Gra
$I_Y = I_Z$
Time Line

Appendix H

REFERENCES

Appendix H

REFERENCES

1. MacPherson, D., Sature, C.W., "An Explicit Method of Guiding a Vehicle from an Arbitrary Initial Position and Velocity to a Prescribed Orbit." 13 February 1961, Aerospace, TDR-594-(1565-01) TN-1.

2. MacPherson, D., "IGS Launch Guidance Equations for GEMINI," 31 October 1962, Aerospace 1955.1-168.

3. Sims, J.R., "Definition of GEMINI Stage 2 Ascent Equations," 12 July 1962, IBM No. 62-564-0125.

4. Sims, J.R., "Definition of GEMINI Stage 1 Guidance," 7 August 1962, IBM No. 62-564-0107B.

5. Sims, J.R., "Insertion into Orbit Using Spacecraft Energy," IBM Internal Memo dated 22 August 1962.

6. "GEMINI Stage 2 Guidance Linear Perturbation Study," 13 October 1962, IBM No. 62-564-0132.

7. Bracato, W., "RGS Launch Guidance Equations for GEMINI," 7 November 1962, Aerospace 1955.1-172.

8. "GEMINI Computer Math Flow Description," 12 November 1962, IBM No. 62-564-0140.

IBM DISTRIBUTION LIST

J. H. Alexander	564, 102-2
C. D. Babb	837, 201-3
D. F. Bachman	564, 102-2
D. R. Baldauf	564, 102-2
D. P. Bedford	Huntsville, Ala.
R. I. Berge (IBM Rep. at MAC)	
K. A. Blaine	564, 102-2
H. F. Branning	564, 102-2
J. H. Burrell	564, 102-2
D. E. Cieslak	564, 102-2
R. E. Cofer	508, 101-2
R. E. Eckstrom	519, 101-2
S. I. Gass	(Bethesda)
P. L. Hertan	564, 102-2
T. W. Hill	564, 102-2
P. J. Holcombe	564, 102-2
J. C. Hundley	837, 201-3
R. B. Jasinski	564, 102-2
L. S. Jimerson	564, 102-2
J. W. Joachim	542, 102-2
R. F. Lovelett	385, 101-3
C. L. McClure	564, 102-2
E. H. Mertz	564, 102-2
P. P. Mooney	542, 102-2
J. R. Pachuts	518, 101-2
J. Park	518, 101-2
J. H. Pursell	564, 102-2
T. F. Roe	564, 102-2
J. R. Sims	Cornell U.
F. L. Tuttle (MAC Rep)	201-3
R. J. Urquhart	564, 102-2
J. F. VanHorn	564, 102-2
R. V. Wadding	508, 101-2
H. K. Wadman	547, 102-1
J. J. Walsh	503, 102-2
R. A. Watson	550, 102-1
E. J. Zola	564, 102-2