NASA CONTRACTOR REPORT

NASA CR-141402

SENS-5D TRAJECTORY AND WIND-SENSITIVITY CALCULATIONS FOR UNGUIDED ROCKETS

(NASA-CR-141402) SENS-5D TRAJECTORY AND WIND-SENSITIVITY CALCULATIONS FOR UNGUIDED ROCKETS (Computer Sciences Corp., Wallops Island, Va.) 155 p HC $6.75 CSCL 01A

N76-18066

Unclas

G3/02 18411

Raghubansh P. Singh

Louis C.P. Huang

Ronald A. Cook

Prepared under Contract No. NAS6-2369 by

Computer Sciences Corporation

Field Services Division

Wallops Island, Virginia 23337

MAR 1976
RECEIVED
NASA STI FACILITY
INPUT BRANCH

National Aeronautics and
Space Administration

Wallops Flight Center
Wallops Island, Virginia 23337
AC 804 824-3411

December 1975

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No
NASA CR-141402		

4. Title and Subtitle	5. Report Date
SENS-5D Trajectory and Wind-Sensitivity Calculations for Unguided Rockets	December 1975
	6. Performing Organization Code
7. Author(s) Raghubansh P. Singh Louis C. P. Huang Ronald A. Cook	8. Performing Organization Report No.
	10. Work Unit No.
9. Performing Organization Name and Address Computer Sciences Corporation Field Services Division 50 E. Kearsarge Drive Wallops Island, Virginia 23337	11. Contract or Grant No. NAS6-2369
	13. Type of Report and Period Covered Contractor Report
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Wallops Flight Center Wallops Island, Virginia 23337	14. Sponsoring Agency Code

15. Supplementary Notes

The F-P-S System of Units is used to be consistent with other routines at the WFC.
In accordance with para. 5.d. of NASA Policy Directive, NPD 2220.4, "Use of the International System of Units (SI) in NASA Publications," a waiver has been granted for this report.

16. Abstract

This report describes a computational procedure which numerically integrates the equations of motion of an unguided rocket. Three translational and two angular (roll discarded) degrees of freedom are integrated through the final burnout; and then, through impact, only three translational motions are considered.

Input to the routine are initial time, altitude and velocity, vehicle characteristics, and other defined options. Input format has a wide range of flexibility for special calculations.

Output are geared mainly to the wind-weighting procedure, and includes summary of trajectory at burnout, apogee and impact, summary of spent-stage trajectories, detailed position and vehicle data, unit-wind effects for head, tail and cross winds, coriolis deflections, range derivative, and the sensitivity curves (the so called F(Z) and DF(Z) curves).

The numerical integration procedure is a fourth-order, modified Adams-Bashforth Predictor-Corrector method. This method is supplemented by a fourth-order Runge-Kutta method to start the integration at $t = 0$ and whenever error criteria demand a change in step size.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Trajectory, Aerodynamics, Numerical Integration, Initial Value Problems, Wind Weighting, Sounding Rockets	Unclassified STAR Category 02

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	152	

*For sale by the National Technical Information Service, Springfield, Virginia 22151

ACKNOWLEDGEMENT

The program requirements were formulated by W. Edward Melson, Jr. and James Andrews of the Wallops Flight Center. They also helped with the final tests of the program. We gratefully acknowledge their assistance.

TABLE OF CONTENTS

SECTION		Page
1	INTRODUCTION	1
2	ASSUMPTIONS	2
3	COORDINATES AND TRANSFORMATIONS	3
	3.1 COORDINATES SYSTEMS	3
	3.2 TRANSFORMATION	4
	3.3 BODY AXES DIRECTION COSINE VARIATIONS	7
4	ROCKET MOTION RELATIVE TO THE AIR	8
	4.1 WIND PROFILE	8
	4.2 COMBINATION OF THE WIND TO THE ROCKET MOTION	8
5	AERODYNAMICS	10
6	EQUATIONS OF MOTIONS	12
	6.1 5-D RIGID BODY DYNAMICS	12
7	NUMERICAL INTEGRATION	14
	7.1 PREDICTOR-CORRECTOR METHOD	14
	7.2 RUNGE-KUTTA METHOD	15
8	WIND-WEIGHTING FUNCTIONS	16
	8.1 BALLISTIC WIND FUNCTION	16
	8.2 UNIT-WIND EFFECTS	16
9	INPUT DESCRIPTION	17
10	OUTPUT DESCRIPTION	25
11	SUBROUTINE DESCRIPTION	26
12	PROGRAM VARIABLE DEFINITIONS	28
	REFERENCES	36
	APPENDIX A: INPUT FORMS	
	APPENDIX B: SAMPLE INPUT	
	APPENDIX C: SAMPLE OUTPUT	
	APPENDIX D: CONTROL CARDS	
	APPENDIX E: SOURCE PROGRAM	

1. INTRODUCTION

Wind weighting, the adjustment of rocket launcher or impact parameters to accommodate prevailing winds, has for years been more of an art than a science. The main reasons are the non-steady characteristics of the wind profile and the complicated nature of the rocket response during the wind perturbation period.

Conventionally, the problem of wind weighting on a rocket has been phrased in two ways: (1) for a given launch elevation and azimuth, evaluate the impact range and bearing; (2) for a given impact range and bearing, predict the required launch elevation and azimuth.[1] The former is straightforward. Given the physical and aerodynamic characteristics of the rocket and a wind profile, the equations of motion may be numerically integrated to achieve impact. The latter involves an iterative procedure, and therefore, several trajectories may have to be integrated. Iterations may or may not be convergent, and in many cases the procedure can become unacceptably time consuming.

In order to avoid integrating the equations of motions several times, the usual practice is to simulate the total wind effect on the complete trajectory rather than evaluating the continuous wind response along the path. This is achieved by defining a "ballistic wind velocity" and a "unit-wind effect", such that their product yields the effective displacement of the impact point due to wind.

A computational procedure, SENS-5D, has been developed and programmed in Fortran-Y to calculate the wind-weighting functions needed to evaluate ballistic wind and unit-wind effects. Coriolis deflections and the first derivative of impact range with launch elevation are also calculated. If desired, a detailed time history of the coordinate and aerodynamic variables computed during integration of the equations of motion and a summary of salient variables at the final burnout, apogee and impact points (payload and spent stages) are available. These data may be used, with a known field, to solve the wind-weighting problem in either of its two forms.

2. ASSUMPTIONS

For simple and accurate simulation, this program utilizes the 5 degree-of-freedom rigid-body dynamics model from first stage ignition to the last stage burnout. After the last burnout, the program shifts to the 3 degree-of-freedom point mass dynamics.

To save time the program assumes linear aerodynamics, i.e., the aerodynamics coefficients are functions of Mach number only.

For stability concern, both the aerodynamic damping and the jet damping terms are used for angular motions.

Only axially symmetric rocket is considered. No attempt has been made to include the effect of thrust or fin misalignment and center-of-gravity deviation from the symmetric axis.

Earth model is a rotating sphere. A 1962 standard atmosphere has been chosen for atmospheric model.

3. COORDINATES AND TRANSFORMATIONS[2]

3.1 COORDINATES SYSTEMS

The coordinate systems used in this trajectory simulation are defined below.

Inertial Coordinate System (X, Y, Z):

Origin -- at earth center.

X - on the earth equatorial plane, pointing to the zero longitude at launch.

Y - on the earth equatorial plane, pointing to the 90° longitude at launch.

Z - perpendicular to the equatorial plane, pointing to the North Pole.

Earth Coordinate System (X_E, Y_E, Z_E):

Origin -- at earth center.

X_E - on the earth equatorial plane, always pointing to the Greenwich longitude.

Y_E - on the earth equatorial plane, always pointing to the 90° longitude.

Z_E - perpendicular to the equatorial plane, pointing to the North Pole.

Instantaneous Topocentric Coordinate System (x_T, y_T, z_T):

Origin -- at the projection point of the moving rocket on earth surface.

x_T - on the local horizon plane tangent to the instantaneous projection point of the rocket, directed along the local geocentric north.

y_T - on the local horizon plane tangent to the instantaneous projection point of the rocket, directed along the local geocentric east.

z_T - perpendicular to the instantaneous local tangent plane, directed along the geocentric radius vector and pointing toward the earth center.

Body Axis Coordinate System (x_B, y_B, z_B):

Origin -- at the center of gravity of the rocket.

x_B - along the rocket principle (longitudinal) axis, positive forward.

y_B - normal to the x_B - z_B symmetric plane, completing the right-hand system.

z_B - in the principle plane of symmetry of the rocket, perpendicular to the x_B axis and positive downward.

3.2 TRANSFORMATION

- Transformation from Body Coordinate System to Instantaneous Topocentric Coordinate System

$$\begin{bmatrix} x_T \\ y_T \\ z_T \end{bmatrix} = \begin{bmatrix} R \end{bmatrix}_{B \to T} \begin{bmatrix} x_B \\ y_B \\ z_B \end{bmatrix} + \begin{bmatrix} T \end{bmatrix}_{B \to T} \tag{3-1}$$

where

$$\begin{bmatrix} T \end{bmatrix}_{B \to T} = \begin{bmatrix} o \\ o \\ -h \end{bmatrix} = \text{translation matrix} \tag{3-2}$$

$$\begin{bmatrix} R \end{bmatrix}_{B \to T} = \begin{bmatrix} \mathrm{Cos}\theta\mathrm{Cos}\psi & \mathrm{Sin}\phi\mathrm{Sin}\theta\mathrm{Cos}\psi - \mathrm{Cos}\phi\mathrm{Sin}\psi & \mathrm{Cos}\phi\mathrm{Sin}\theta\mathrm{Cos}\psi + \mathrm{Sin}\phi\mathrm{Sin}\psi \\ \mathrm{Cos}\theta\mathrm{Sin}\psi & \mathrm{Sin}\phi\mathrm{Sin}\theta\mathrm{Sin}\psi + \mathrm{Cos}\phi\mathrm{Cos}\psi & \mathrm{Cos}\phi\mathrm{Sin}\theta\mathrm{Sin}\psi - \mathrm{Sin}\phi\mathrm{Cos}\psi \\ -\mathrm{Sin}\theta & \mathrm{Sin}\phi\mathrm{Cos}\theta & \mathrm{Cos}\phi\mathrm{Cos}\theta \end{bmatrix}$$

$$= \text{rotation matrix} \tag{3-3}$$

θ - Euler angle for pitch

ψ - Euler angle for yaw

ϕ - Euler angle for roll

However, for the present 5-degree-of-freedom case, the roll angle is always equal to zero. Thus, the rotation matrix can be simplified to

$$\left[R\right]_{B\to T} = \begin{bmatrix} \mathrm{Cos}\theta\mathrm{Cos}\psi & -\mathrm{Sin}\psi & \mathrm{Sin}\theta\mathrm{Cos}\psi \\ \mathrm{Cos}\theta\mathrm{Sin}\psi & \mathrm{Cos}\psi & \mathrm{Sin}\theta\mathrm{Sin}\psi \\ -\mathrm{Sin}\theta & 0 & \mathrm{Cos}\theta \end{bmatrix} \tag{3-4}$$

- Transformation from Instantaneous Topocentric Coordinate System to Inertial Coordinate System

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \left[R\right]_{T\to I} \begin{bmatrix} x_T \\ y_T \\ z_T \end{bmatrix} + \left[T\right]_{T\to I} \tag{3-5}$$

where

$$\left[R\right]_{T\to I} = \begin{bmatrix} -\mathrm{Sin}\mu\mathrm{Cos}\bar{\lambda} & -\mathrm{Sin}\bar{\lambda} & -\mathrm{Cos}\mu\mathrm{Cos}\bar{\lambda} \\ -\mathrm{Sin}\mu\mathrm{Sin}\bar{\lambda} & \mathrm{Cos}\bar{\lambda} & -\mathrm{Cos}\mu\mathrm{Sin}\bar{\lambda} \\ \mathrm{Cos}\mu & 0 & -\mathrm{Sin}\mu \end{bmatrix}$$

$$= \text{ rotation matrix} \tag{3-6}$$

$$\left[T\right]_{T\to I} = \begin{bmatrix} R_E\ \mathrm{Cos}\mu\mathrm{Cos}\bar{\lambda} \\ R_E\ \mathrm{Cos}\mu\mathrm{Sin}\bar{\lambda} \\ R_E\ \mathrm{Sin}\mu \end{bmatrix} = \text{ translation matrix} \tag{3-7}$$

μ - latitude of the Instantaneous Topocentric Coordinates origin.

λ - longitude of the Instantaneous Topocentric Coordinates origin with respect to the rotating Greenwich longitude.

$\bar{\lambda}$ - longitude of the Instantaneous Topocentric Coordinates origin with respect to the inertial zero longitude at launch.

$$\bar{\lambda} = \lambda + \omega_E t$$

ω_E - angular velocity of the earth rotation.

- Transformation from Body Coordinate System to Inertial Coordinate System

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} R \end{bmatrix}_{B \to I} \begin{bmatrix} x_B \\ y_B \\ z_B \end{bmatrix} + \begin{bmatrix} T \end{bmatrix}_{B \to I} \tag{3-8}$$

where

$$\begin{bmatrix} R \end{bmatrix}_{B \to I} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} = \text{rotation matrix} \tag{3-9}$$

$$R_{11} = -\sin\mu\cos\bar{\lambda}\cos\theta\cos\psi - \sin\bar{\lambda}\cos\theta\sin\psi + \cos\mu\cos\bar{\lambda}\sin\theta$$

$$R_{12} = \sin\mu\cos\bar{\lambda}\sin\psi - \sin\bar{\lambda}\cos\psi$$

$$R_{13} = -\sin\mu\cos\bar{\lambda}\sin\theta\cos\psi - \sin\bar{\lambda}\sin\theta\sin\psi - \cos\mu\cos\bar{\lambda}\cos\theta$$

$$R_{21} = -\sin\mu\sin\bar{\lambda}\cos\theta\cos\psi + \cos\bar{\lambda}\cos\theta\sin\psi + \cos\mu\sin\bar{\lambda}\sin\theta$$

$$R_{22} = \sin\mu\sin\bar{\lambda}\sin\psi + \cos\bar{\lambda}\cos\psi$$

$$R_{23} = -\sin\mu\sin\bar{\lambda}\sin\theta\cos\psi + \cos\bar{\lambda}\sin\theta\sin\psi - \cos\mu\sin\bar{\lambda}\cos\theta$$

$$R_{31} = \cos\mu\cos\theta\cos\psi + \sin\mu\sin\theta$$

$$R_{32} = -\cos\mu\sin\psi$$

$$R_{33} = \cos\mu\sin\theta\cos\psi - \sin\mu\cos\theta$$

NOTE: Here again the Euler angle of roll (ϕ) has been eliminated from the transformation components.

$$\begin{bmatrix} T \end{bmatrix}_{B \to I} = \begin{bmatrix} R\cos\mu\cos\bar{\lambda} \\ R\cos\mu\sin\bar{\lambda} \\ R\sin\mu \end{bmatrix} = \text{translation matrix} \tag{3-10}$$

$$R = (R_E + h) = (X^2 + Y^2 + Z^2)^{1/2} \tag{3-11}$$

3.3 BODY AXES DIRECTION COSINE VARIATIONS

The angles of the direction cosine are sometimes undefined, e.g., the heading azimuth becomes indeterminate if the flight elevation reaches 90°. Thus, the elements of the rotation matrix (e.g. 3-9) are calculated from the orientation angles only during problem starts. Afterward, the elements of the rotation matrix are determined by integrating the time rate of change of the direction cosine elements.

The time rate of change of the rotation matrix elements can be calculated from pitch ($\tilde{P}$), yaw ($\tilde{Y}$), and spin ($\tilde{S}$) as follows:

$$\begin{bmatrix} \dot{R}_{11} & \dot{R}_{12} & \dot{R}_{13} \\ \dot{R}_{21} & \dot{R}_{22} & \dot{R}_{23} \\ \dot{R}_{31} & \dot{R}_{32} & \dot{R}_{33} \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \begin{bmatrix} 0 & -\tilde{Y} & \tilde{P} \\ \tilde{Y} & 0 & -\tilde{S} \\ -\tilde{P} & \tilde{S} & 0 \end{bmatrix} \tag{3-12}$$

Since three of the nine rotation matrix elements depend on the remaining six, it is sufficient to integrate only six rotation matrix elements and calculate the remaining three from the relation $\vec{k}_B = \vec{i}_B \times \vec{j}_B$, i.e.

$$\begin{bmatrix} R_{13} \\ R_{23} \\ R_{33} \end{bmatrix} = \begin{bmatrix} R_{21} R_{32} - R_{31} R_{22} \\ R_{31} R_{12} - R_{11} R_{32} \\ R_{11} R_{22} - R_{21} R_{12} \end{bmatrix} \tag{3-13}$$

4. ROCKET MOTION RELATIVE TO THE AIR

4.1 WIND PROFILE

The wind profile data is usually gathered and made available to the range by on-site or nearby weather station. The composite wind data are smoothed and edited into the format of local north wind component (W_N) and local east wind component (W_E) vs. altitude (h). Due to its random nature, the vertical drift or gust of the wind is totally neglected in the present simulation.

Since the wind data are in local topocentric coordinates, transformation must be performed in order to re-orient its components into the body axes coordinates.

Assume $[u_W, v_W, w_W]$ are the wind velocity components in the body axes coordinates, then

$$\begin{bmatrix} u_W \\ v_W \\ w_W \end{bmatrix} = \begin{bmatrix} R \end{bmatrix}_{T \to B} \begin{bmatrix} W_N \\ W_E + \omega_E R \cos\mu \\ 0 \end{bmatrix} \tag{4-1}$$

where

$\begin{bmatrix} R \end{bmatrix}_{T \to B}$ = rotation matrix from topocentric coordinates to body axes coordinates

4.2 COMBINATION OF THE WIND TO THE ROCKET MOTION

The rocket velocity relative to the true air is equal to the rocket absolute velocity relative to earth plus the wind velocity, i.e.

$$\begin{bmatrix} u_{RA} \\ v_{RA} \\ w_{RA} \end{bmatrix} = \begin{bmatrix} u + u_W \\ v + v_W \\ w + w_W \end{bmatrix} \tag{4-2}$$

where (u, v, w) are the velocity component in body axes coordinates. Finally, the total velocity relative to the true air becomes

$$V_{RA} = [u_{RA}^2 + v_{RA}^2 + w_{RA}^2]^{1/2} \tag{4-3}$$

and

$$\text{Dynamic Pressure} = q = \frac{\rho V_{RA}^2}{2} \tag{4-4}$$

$$\text{Mach No.} = M = \frac{V_{RA}}{a} \tag{4-5}$$

5. AERODYNAMICS

In SENS-5D forces are assumed to vary linearly with the aerodynamic angles arising from vehicle motions relative to the air mass. These angles are defined as:

$$\text{Angle of attack} \equiv \alpha = \mathrm{Tan}^{-1}\left[w_{RA}/u_{RA}\right]$$

$$\text{Angle of sideslip} \equiv \beta = \mathrm{Tan}^{-1}\left[v_{RA}/u_{RA}\right] \qquad (5\text{-}1)$$

The resulting forces are:

$$\text{Drag (force along X-axis)} \equiv D = C_D qs$$

$$\text{Normal force due to } \alpha \text{ (along Z-axis)} \equiv N_\alpha = C_{N\alpha} qs$$

$$\text{Normal force due to } \beta \text{ (along Y-axis)} \equiv N_\beta = C_{N\beta} qs \qquad (5\text{-}2)$$

$$\text{Jet damping force (along Z-axis)} \equiv F_{jz} \equiv \dot{m} L_{NC} \tilde{P}$$

$$\text{Jet damping force (along Y-axis)} \equiv F_{jY} = \dot{m} l_{NC} \tilde{Y}$$

Because an axially symmetrical vehicle is assumed, we have $C_{N\alpha} = C_{N\beta} = C_N$ and $C_{N\alpha,\beta} = 0$ when $\alpha,\beta = 0$.

The moments arising from these forces are:

$$\text{Pitching moment (about Y axis)} \equiv M_Y = N_\alpha L_{sm}$$

$$\text{Yawing moment (about Z axis)} \equiv M_Z = N_\beta L_{sm}$$

$$\text{Pitch damping moment (about Y axis)} \equiv M_{mY} = C_{mp}(\tilde{P}d/2V_{RA})qsd \qquad (5\text{-}3)$$

$$\text{Yaw damping moment (about Z axis)} \equiv M_{mZ} = C_{mY}(\tilde{Y}d/2V_{RA})qsd$$

$$\text{Jet damping moment (about Y axis)} \equiv M_{jY} = F_{jz} L_{NC}$$

$$\text{Jet damping moment (about Z axis)} \equiv M_{jZ} = F_{jY} L_{NC}$$

Again due to symmetry, $C_{mp} = C_{my}$. In the above equations:

q = dynamic pressure

s = reference area

d = reference diameter

$\dot{m}$ = time rate of change of mass

L_{sm} = static margin (5-4)

L_{NC} = distance from nozzle exit plane to c.g.

C_D = drag or axial force coefficient

C_N = normal force coefficient

C_{mp} = pitch damping force coefficient

$\tilde{P}$ = pitch rate (angular velocity)

$\tilde{Y}$ = yaw rate (angular velocity)

6. EQUATIONS OF MOTIONS

For the purpose of achieving both time saving and sufficient accuracy, the present trajectory simulation employs 5-D rigid-body dynamics from the first ignition to the last burnout. After the last burnout, 3-D point-mass dynamics is adopted.

6.1 5-D RIGID BODY DYNAMICS

6.1.1 Equations of Linear Motions

The equations of linear motion are written in the inertial system. Let

$$r = \sqrt{X^2 + Y^2 + Z^2}$$

be the distance of the vehicle from the center of the earth. The linear motions are then given by

$$\frac{d^2\vec{r}}{dt^2} = -\frac{GM}{\vec{r}^{\,3}}\,\vec{r} + \left[R\right]_{B\to I}\frac{\vec{E}}{m}\,, \tag{6-1}$$

where

GM = product of universal gravitational constant and mass of the earth,

$\left[R\right]_{B\to I}$ = rotation matrix from body to inertial system (see 3-9),

m = total mass of the vehicle,

$\vec{E}$ = external force on the vehicle in body system,

$$= \begin{bmatrix} Th - D \\ -N_\beta + 2F_{jy} \\ -N_\alpha - 2F_{jz} \end{bmatrix} \tag{6-2}$$

with Th and D as thrust and drag forces respectively, and other terms are defined in Equation 5-2.

6.1.2 Equations of Angular Motions

The equations of angular motions are written in body system. It is assumed that rocket is axially symmetric and that roll is neglected.

Let $\vec{\Omega}(\tilde{P},\tilde{Y})$ be the angular velocity vector about Y and Z axes (i.e., pitch and yaw) respectively. Then the angular motions are given by

$$\frac{d^2\vec{\Omega}}{dt^2} = \frac{\vec{M}}{I} , \tag{6-3}$$

where

I = moment of inertia about Y or Z axis,

$\vec{M}$ = external moment (torque) on the vehicle,

$$= \begin{bmatrix} 0 \\ -M_y - M_{my} - M_y \\ M_z - M_{mz} - M_{jz} \end{bmatrix} ,$$

with the terms above defined in Equation 5-3.

6.2 3-D POINT MASS DYNAMICS

The equations of linear motions are written in inertial system. No angular motions are considered. The equations are given as in (6-1), but external forces are now computed in topocentric system, i.e.

$$\frac{d^2\vec{r}}{dt^2} = -\frac{GM}{\vec{r}^{\,3}}\vec{r} + \left[R\right]_{T\to I}\frac{\vec{E}}{m} , \tag{6-4}$$

where

$\left[R\right]_{T\to I}$ = rotation matrix from topocentric to inertial system, and

$$\vec{E} = \begin{bmatrix} (Th-D)\ \text{Cos e Cos a} \\ (Th-D)\ \text{Cos e Sin a} \\ -(Th-D)\ \text{Sin e} \end{bmatrix} , \tag{6-5}$$

with e and a as flight path angles (elevation and azimuth respectively.)

7. NUMERICAL INTEGRATION

7.1 PREDICTOR-CORRECTOR METHOD[3,4]

The integration technique is Adams-Bashforth's fourth-order, modified predictor-corrector method for the solution of general initial-value problems.

Given k first-order differential equations

$$\frac{d^2y_i}{dx^2} = f_i(x,y_i); \quad i = 1,----k \tag{7-1}$$

and values of y, and therefore f(x,y), at four equally- spaced intervals (h) of x, the result for y at the fifth point of x is obtained by the following scheme:

$$\text{Predictor: } y_{n+1}^{(0)} = y_n + \frac{h}{24}\,[55f_n - 59f_{n-1} + 37f_{n-2} - 9f_{n-3}], \tag{7-2}$$

$$\text{Modifier: } \bar{y}_{n+1}^{(0)} = y_{n+1}^{(0)} - \frac{251}{270}\,(y_n^{(0)} - y_n), \tag{7-3}$$

$$\text{Corrector: } y_{n+1}^{(c)} = y_n + \frac{h}{24}\,[9f(x_{n+1},\bar{y}_{n+1}^{(0)}) + 19f_n - 5f_{n-1} + f_{n-2}], \tag{7-4}$$

$$\text{Final Value: } y_{n+1} = y_{n+1}^{(c)} + \frac{19}{270}\,(y_{n+1}^{(0)} - y_{n+1}^{(c)}) \tag{7-5}$$

Above the subscript i is dropped because (7-2) to (7-5) hold for any i.

Expressions (7-2) and (7-4) have local truncation errors. Errors are proportional to h^5 times the fifth derivative of y. Assuming that this fifth derivative does not vary appreciably in the interval, they are given by the second terms in (7-3) and (7-5) for the predictor and for the corrector respectively. The second term in (7-5) is, therefore, a measure of the truncation error in y_{n+1}, and therefore control of accuracy and adjustment of step size h can be done by generating the following test value:

$$\delta = \frac{1}{K}\sum_{i=1}^{k} w_i \left| y_{n+1}^{(o)} - y_{n+1}^{(c)} \right| , \tag{7-6}$$

where w_i are error weights specified in the input.

If δ is greater than the upper error tolerance, step size h is halved. If δ is less than the lower error tolerance, step size is doubled.

7.2 RUNGE-KUTTA METHOD

Starting the predictor-corrector method requires the functional and derivative values at four equidistant points. y_0 and $f(x_0,y_0)$ are given initial conditions. For computation of y_1, $f(x_1,y_1)$, y_2, $f(x_2,y_2)$, y_3 and $f(x_3,y_3)$ and for adjustment of the step size h to accuracy requirements, a special fourth-order Runge-Kutta procedure suggested by Ralston* is used.

For initial control of accuracy and adjustment of step size h in starting the integration, a result for $y_2^{(1)} = y(x_0+h)$ is computed using step size h, and then another result $y_2^{(2)} = y(x_0+\frac{h}{2}+\frac{h}{2})$ is computed using step size h/2 twice. If the relative error, defined by

$$\delta = \frac{1}{15} \left| (y_2^{(2)} - y_2^{(1)}) \Big/ y_2^{(2)} \right| \tag{7-7}$$

is less than 10^{-5}, then solution $y_2^{(2)}$ is assumed correct. If $\delta > 10^{-5}$, step size is halved, and the procedure starts again at point x_0. In case step size is less than a given minimum, the last solution $y_2^{(2)}$ would be assumed correct.

*See Reference 3, Page 200, Expression (5.6-49)

8. WIND-WEIGHTING FUNCTIONS[1]

8.1 BALLISTIC WIND FUNCTION

Ballistic wind velocity $\vec{W}$ is a hypothetical wind which is constant in direction and magnitude from the ground level to a defined upper limit of the effective atmosphere Z_{max}. In SENS-5D, Z_{max} = 100000 ft. The integrated effect of $\vec{W}$ on the rocket impact is equal to the integrated effected of the actual wind.

Ballistic wind is defined by

$$\vec{W} = \sum_{i=1}^{N} d_i \vec{W}_i \; ; \tag{8-1}$$

$$\sum_{i=1}^{N} d_i = 1, \; Z_{max} = Z_N, \tag{8-2}$$

where $\vec{W}_i$ is the wind velocity at altitude levels i = 1,2,3,---N, and d_i are the weights defined by

$$d_i = \frac{I_i - I_{i-1}}{I_N - I_0} \; , \tag{8-3}$$

such that I_i is the impact vector with winds between ground and the ith altitude levels only. d_i's are sometimes called wind-weighting function, so called delta-f(z) curves.

Another wind-weighting function, which is of interest is f(z) curve defined by

$$f_i = \frac{I_i - I_0}{I_N - I_0} \quad . \tag{8-4}$$

8.2 UNIT-WIND EFFECTS

The unit-wind effect $\delta(e)$ for a given launch elevation e, is the magnitude of the impact displacement vector due to a unit ballistic wind to the height Z_{max}. Assuming that the rocket response to wind is linear, the impact displacement vector due to the winds is $\delta(e)W$. The definition of $\delta(e)$ is extended to provide for head, tail and cross winds.

9. INPUT DESCRIPTION

As with any aerodynamics program, the most difficult problem for the user is understanding and preparing the voluminous input required. We have attempted to simplify the preparation of the input data.

The input data are split into two sets. Set 1 is a series of four program control lists, which use the FORTRAN 'NAMELIST' facility. Set 2 encompasses the aerodynamic tables. The data is read with FORMAT(V), a list directed read operation which may be peculiar to Honeywell Series 6000 FORTRAN.

SET 1

All the entries in Set 1 are entered in 'NAMELIST' format, on file code 05.

Set 1 consists of four lists, all of which must be present. The lists are: DLIST, BLIST, FLIST and ULIST. DLIST may be empty if one desires to keep all of its default values. The other lists may be empty only if they are to be bypassed. All the lists have default values defining some or all the input variables. They may be overridden by re-defining them.

Note that the units used in this program are feet, pounds, seconds, and degrees.

Definition of the Variables in DLIST

The format of DLIST is:

$DLIST in Columns 2 through 7, and then

GDLATL = Geodetic latitude of launcher

LONGL = Geodetic longitude of launcher

TZERO = Initial time

AZERO = Initial altitude

VZERO = Initial velocity

ITAPE1 = Number of the file code used for the input of Set 1. This is a dummy entry. The program forces Set 1 to be read from file code 05.

ITAPE2 = Number of the file code used for the input of data Set 2.

ITAPE3 = Number of the file code for storing thrust-mach tables after cross-interpolation to get one time table and one mach table.

ITAPE4 = Number of the file code for storing data to be used for spent stage calculations.

DTLA = Step size to be used from launch through apogee

DTAI = Step size to be used from apogee through impact

DMIN = Minimum step size allowed

DMAX = Maximum step size allowed

EPTINY = Minimum error allowed during integrations

EPBIG = Maximum error allowed during integration

ERW = Vector of error weights used in the predictor-corrector

VOLIM = Minimum initial velocity

VLIM = If current velocity < VLIM, apogee is assumed.

HLIM = If current altitude (after apogee) < HLIM, impact is assumed.

ORDER = Array of length 14 storing the orders of polynomials used in interpolations to create one time table and one mach table.

IPUNCH = Punch-out option.

= 0 No punch-out.
≠ 0 Will be punched-out.

$ after the last item in the list.

Definition of the Variables in BLIST

BLIST controls to option to calculate burnout flight elevation, apogee altitude, spent-stage impact, and payload impact range as a function of launch elevation angle and payload weight.

If this option is not required, enter BLIST as an empty list with $BLIST in Columns 2 through 7 and $ in Column 10.

The format of a non-empty BLIST is:

$BLIST in Columns 2 through 7, and then

NPL = Number of payloads

PLM = Array to store NPL payloads

AZGDL = Launch azimuth (geodetic)

NANG = Number of launch elevation angles

ANG = Array of NANG elevation angles (geodetic)

WIND,
WNAZ,
ALOW
AHIGH = These four variables are used to define a given constant wind from one altitude to another, and zero elsewhere.

WIND = Wind speed (always positive)

WNAZ = Wind azimuth, measured clockwise from North, (North-to-South or East-to-West is positive wind)

ALOW = Minimum altitude of the wind

AHIGH= Maximum altitude of the wind

NLEV,
ALTW,
SPEED,
DIR = These four variables are used for defining a variable wind, i.e. a wind which varies with respect to altitude

NLEV = Number of altitude levels

ALTW = Array of NLEV altitudes

SPEED = Array of wind speeds (magnitude of wind velocity)

DIR = Array of wind direction (measured clockwise from local North)

IROT = = 0: non-rotating earth
≠ 0: rotating earth

IPRINT = Print-out option

= 0: summary only
= 1: detail trajectory with summary
= 2: debug option

JSPENT = Option for calculating spent stage(s) trajectories
= 0: no calculation
= Ø: will be calculated

$ after the last item in the list.

Definition of the Variables in FLIST

FLIST controls the option to calculate the wind-weighting factor as a function of altitude.

If this option is not desired, FLIST must still be present as an empty list with $FLIST in Columns 2 through 7, and $ in Column 10.

The format of a non-empty FLIST is:

$FLIST in Columns 2 through 7, and then

WPL = Payload weight

AZGDL = Launch azimuth (geodetic)

ELGDL = Launch elevation (geodetic)

WIND = Wind speed (always positive)

WNAZ = Wind azimuth (measured clockwise from North, while North-to-South or East-to-West is positive)

NLEV = Number of wind altitudes

ALEV = Array of NLEV altitudes

IROT = = 0: non-rotating earth
≠ 0: rotating earth

IPRINT = = 0: summary only
= 1: detail trajectory with summary
= 2: debug option

$ after the last item in the list.

Definition of the Variables in ULIST

ULIST controls the option to calculate the coriolis deflections due North and East, range derivatives and unit-wind effects for head, tail and cross winds as a function of launch elevation angle.

As in lists BLIST and FLIST, ULIST may be empty if this option is not desired. For an empty list, put $ULIST in columns 2 through 7 and $ in Column 10.

The format of a non-empty ULIST is:

$ULIST in Columns 2 through 7, and then

WPL	=	Payload weight
AZGDL	=	Launch azimuth (geodetic)
NANG	=	Number of launch elevations
ANG	=	Array of NANG launch elevations (geodetic)
WIND	=	Wind speed (always positive)
ALOW	=	Minimum altitude
AHIGH	=	Maximum altitude
IROT	=	= 0: non-rotating earth ≠ 0: rotating earth
IPRINT	=	= 0: summary only = 1: detail trajectory with summary = 2: debug option

$ after the last item in the list.

Default Values

Unless otherwise defined, the following values for the variables in NAMELIST data will be assumed by the program.

NAMELIST	VARIABLE		
$DLIST	GDLATL	=	37.848
	LONGL	=	-75.4736
	TZERO	=	0
	AZERO	=	0
	VZERO	=	0
	ITAPE1	=	5 - This value is forced by the program.
	ITAPE2	=	1
	ITAPE3	=	2
	ITAPE4	=	3
	DTLA	=	0.01
	DTAI	=	1.0
	DMIN	=	0.0002
	DMAX	=	20.0
	EPTINY	=	0.0001
	EPBIG	=	0.001
	ERW	=	All of them = 1.0
	VOLIM	=	0.1
	VLIM	=	40.0

NAMELIST	VARIABLE		
$DLIST (continued)	HLIM	=	20.0
	ORDER	=	All of them = 1
	IPUNCH	=	0
$BLIST	WIND	=	0
	WNAZ	=	0
	ALOW	=	0
	AHIGH	=	100,000
	NLEV	=	0
	IROT	=	1
	IPRINT	=	0
	JSPENT	=	0
$FLIST	WIND	=	20.0
	WNAZ	=	0
	NLEV	=	12
	ALEV	=	0, 50, 100, 200, 500, 1000, 2000, 3000, 5000, 10000, 50000, 100000
	IROT	=	0
	IPRINT	=	0
$ULIST	WIND	=	20.0
	ALOW	=	0
	AHIGH	=	100,000
	IROT	=	0
	IPRINT	=	0

SET 2

Vehicle Data

These data are read from file code ITAPE2 (=1 unless redefined in DLIST of SET 1) using "FORMAT(V)". This is a list directed read which may be peculiar to Honeywell Series 6000 Fortran. Data are entered on cards using columns one through 72 (1-72), and are separated by commas, spaces, or by beginning a new card (see pitfall 2 below). Decimal points should be used in real numbers, deleted in integers. All units are feet, pounds, degrees or seconds.

SET 2 begins with a title card and is divided into a block of general data and a series of phases. A new phase is required whenever a change in aerodynamics, vehicle geometry or a step change in weight occurs. Each phase is preceded by its own title card. Within the general block and each phase, data are presented in a specific order as groups or tables. Each group or table is also preceded by a title card. The content of the title cards is left to the user, but their presence is mandatory.

In the general data block, the first group contains the total rocket weight (less payload); followed by the time at which point mass (rather than 5-D) integration is to begin (normally the start of the final phase). This group is followed by three tables in exactly this order,

1. phase start times (referenced to zero)

2. time for spent stage separations (these must also be phase start times)

3. discarded weight of spent stages (one for each time in Table 2)

The first value in each table must be the number of data points (an integer) in the table. Each table must have at least one entry, zero if no other value is appropriate. If the limits of an independent table are exceeded, the last corresponding dependent values are used in subsequent calculations.

In each phase the first group of data consists of four vehicle dimensions. They are nozzle exit area, length of the vehicle, reference area, and reference length. It is followed by fourteen (14) tables in exactly this order.

1. Time (since phase start) for thrust and weight;

2. Thrust (one value for each time in Table 1. Vacuum reference is assumed unless the word "SEA" appears in the title card for this table);

3. Expendable weight remaining in this stage (one value for each time in Table 1);

4. Mach number for drag coefficients;

5. Drag coefficients (one for each entry in Table 4);

6. Time (since phase start) for gravimetrics;

7. Distance from nose to center of gravity (one for each time in Table 6)

8. Pitch moment of inertia (one for each time in Table 6);

9. Mach number for pitch damping coefficient;

10. Pitch damping coefficient (one value, entered as a positive number, for each entry in Table 9);

11. Mach number for slope of normal force coefficient;

12. Slope of normal force coefficient per unit angle of attack (one for each entry in Table 11);

13. Mach number for center of pressure location;

14. Distance from nose to center of pressure (one for each entry in Table 13).

The rules for table entries in the general block also apply to the phase tables. If the limits of an independent table are exceeded, the last corresponding dependent values are used in subsequent calculations. Data are not saved from phase to phase.

Input of SET 2 is greatly facilitated by the use of preprinted forms containing the title cards and spaces to record data. A set of forms prepared for use by the Ground and Flight Safety Section, SQAEB, NASA Wallops Flight Center is shown in Appendix A. The user may find this input clearer after scanning these forms and the sample input for a Nike Cajun shown in Appendix B.

Pitfalls to Avoid

1. It is a rule of "FORMAT(V)" that only one delimiter (comma, blank, etc) appear between values in tables. Particularly, lines should not end with a comma, and a given value may not be continued on another card.

2. A zero level of thrust is always taken with respect to a vacuum. Therefore, during coasting phases, the thrust table (input as a single value, zero) must not be titled with the word "SEA", or errant forces may be calculated for nozzle exit corrections.

3. SET 2, each phase, and each data group or table must be immediately preceded by a title card and must appear in the order specified.

4. In SET 1, all four namelists must be present, and each must be closed with a dollar sign ($). Examples of empty lists appear in Appendix A and of used lists in Appendix B.

10. OUTPUT DESCRIPTION

Appendix C shows the output which is produced when the input set of Appendix B is used.

Page 1 is an echo of the Set No. 1 input lists.

Pages 2 through 5 are an echo of the Set No. 2 tables.

Page 6 gives a summary for the reference trajectories.

Page 7 gives a summary of the trajectory of the spent stage.

Page 8 lists the F(Z) and DF(Z) data for the given wind condition.

Pages 9 and 10 list the unit-wind effects, range derivatives, and coriolis deflections both in English units and in MKS units (which are often required for off-range operations).

Page 11 is an echo of the punch output of the SENS-5D data which may be used in wind weighting. Reader should note IPUNCH = 1 in $DLIST of input data (see Appendix B).

Pages 12 through 19 list a detailed trajectory which may be useful in comparing the SENS-5D trajectory with another trajectory provided by a range user.

11. SUBROUTINE DESCRIPTION

APCAL: Calculates current time, altitude, surface azimuth, and surface range at the apogee point.

APSTEP: Calculates the time that will be taken from current position to the apogee point. This is used in adjusting the step size until the apogee is approached.

ATMSPH: Computes temperature, pressure, density, and the speed of sound using an eighth degree polynominal approximation for the 1962 Standard Atmospheric Tables.

ATTERP: Calculates the indices within which the current altitude falls in the altitude tables (called by ATMSPH).

AZRAN: Calculates current latitude, longitude, surface azimuth, and surface range.

BAI: Integrates the trajectory, for a given set of NANG elevation angles and a set of NPL payloads. Prints out salient features at burnout, apogee, and impact points.

BOCAL: Calculates current time, altitude, surface range and azimuth, and relative velocity at the final burnout.

EFFECT: Calculates, for a given payload and a given set of NANG elevation angles, unit-wind effects for head, tail, and cross winds, coriolis deflection, and tower tilt.

FZCURV: Calculates for a given payload and elevation angle the F(Z) curve. Also calculates the ballistic wind factors defined by DF(Z) = F(Z) - F(Z-1).

IPCAL: Calculates current time, altitude, surface azimuth and range at the impact.

IPSTEP: Calculates the time that will be taken from the current position to the impact point. This is used in adjusting the step size until the impact is approached.

LINT: Program for linear interpolation using Aitken's Iteration method employing a polynomial of degree up to 10. This can be used for extrapolation, but with caution.

LIST: Prints input data.

PCM: Knowing values at a set of four points, it integrates at the fifth point (step size DT). Procedure is modified Adams-Bashforth Predictor-Corrector method. If error > EPBIG, step size is decreased by a factor of 2; if error < EPTINY, step size is increased by a factor of 2. It determines if the phase will change after DT, and updates the data accordingly. (References 3 and 4)

RDATA: Reads from disc number ITAPE2 the thrust and drag data, interpolates for one time table and one mach number table, and stores the data on disc number ITAPE3.

REFAZE: Updates the thrust-drag data, the integration variables, and mass at the phase change. Also stores integration variables for spent stages.

RK4: Runge-Kutta method of order 4. At launch and whenever the step size changes, this subroutine is called. There is an option to integrate one step or three steps. (Reference 3)

ROCKET: Interpolates aerodynamic variables employing tables created in RDATA at current time and Mach No.

RTERP: Calculates indices to be used in ROCKET.

RTIMAT: Calculates at the current time the rotation matrix from topocentric to inertial axes. Also calculates altitude.

SETIV: Initializes the various physical parameters necessary to start the integration at launch.

SOLV3D: Solves for functions F in equation Y' = f in 3D after final burnout.

SOLV5D: Solves for functions F in equation Y' = f in 5D from launch to final burnout.

SPENT: Calculates trajectory for each spent stage.

SPLASH: This is similar to BAI, but is called by EFFECT and FZCURV.

TPRINT: Prints in detail the salient features along the trajectory.

WCONST: Constant wind table. Given the input parameters (WIND, WNAZ, ALOW, AHIGH), it creates table: ALTW(1) = ALOW, ALTW(2) = ALOW+1, ALTW(3) = AHIGH; VELN(1) = VELE(1) = 0, VELN(2) = WIND x COS(WNAZ+PI), VELE(2) = WIND X SIN(WNAZ+PI), VELE(3) = VELE(2), and VELN(3) = VELN(2).

WTERP: Interpolates for wind velocities at current altitude.

12. PROGRAM VARIABLE DEFINITIONS

A — Rocket thrust and mach number tables, A (I,J). I is the element number. J has the following meanings:

- J = 1: Time array for thrust and propellant weight.
- = 2: Thrust table.
- = 3: Propellant weight table.
- = 4: Mach number array for drag coefficient.
- = 5: Axial drag coefficient.
- = 6: Time array for C.G. position (from nose) and moment of inertia.
- = 7: Distance from nose to C.G.
- = 8: Pitch moment of inertia.
- = 9: Mach number table for pitch damping coefficient.
- = 10: Pitch damping coefficient
- = 11: Mach number table for slope of normal force coefficient.
- = 12: Slope of normal force coefficient.
- = 13: Mach number table for nose to c.p. distance.
- = 14: Nose to center-of-pressure distance.

ACC: Relative acceleration in inertial coordinate system.

AD: Axial drag.

AHIGH: Maximum altitude up to which wind is used and beyond which is zero.

ALEV: Altitudes of wind strata for F(Z) curve.

ALOW: Minimum altitude below which wind is zero.

ALT: Current altitude.

ALTW: Altitude table for winds.

ANG: Array for launch elevation angles.

APA: Altitude at the apogee point.

APAZ: Surface azimuth below the apogee point.

APR: Surface range below the apogee point.

APREV: Previous altitude for interpolation among the wind altitude table.

APT: Current time at the apogee.

ARRAY: Dummy array to store data or variables.

AZ: Current azimuth (geocentric).

AZGD: Current azimuth (geodetic).

AZGDL: Launch azimuth (geodetic).

B: Array of length 4. In B(J), J stands for:

J = 1: nozzle exit area
= 2: distance from nose to nozzle exit
= 3: aerodynamic reference area
= 4: reference diameter

BOA: Altitude at the burnout point.

BOAZ: Surface azimuth at burnout.

BOEL: Elevation angle at burnout.

BOR: Surface range at burnout.

BOT: Current time at burnout.

BOVL: Relative velocity (topocentric) at burnout.

CA: Drag coefficient.

CF: Array of conversion factors. Multiplying by CF(J) means:

J = 1: radian to degree
= 2: degree to radian
= 3: nautical miles to feet
= 4: feet to nautical miles
= 5: inverse of acceleration due to gravity (GO)
= 6: feet to meters
= 7: meter to feet
= 8: nautical miles to kilometers
= 9: kilometers to nautical miles

CMP: Pitch damping coefficient

CONST: $=34.163194 \times 10^{-3}$ °K/meter. Constant in atmosphere pressure formula.

DENS: Current atmospheric density.

DENSO: Air density at sea level = 2.3769×10^{-3} slug/ft^3.

DIR: Direction table for wind.

DMAX: Maximum step size during integration procedure.

DMIN: Minimum step size during integration procedure.

DPS: Array for storing integration variables S(14).

DT: Current step size for integration.

DTAI: Initial step size for integration from apogee to impact.

DTLA: Initial step size for integration from launch to apogee.

EL: Current elevation angle (geocentric).

ELGD: Current elevation angle (geodetic).

EPBIG: Tolerance used in step size control (if current error > EPBIG, step size is decreased).

EPTINY: Tolerance used in step size control (if current error < EPTINY, step size is increased).

ERW: Vector of error weights used in predictor-corrector method. These weights, in conjunction with EPBIG and EPTINY, are used to control step size during integrations.

ESQ: Square of the earth's eccentricity $= 6.6934217 \times 10^{-3}$.

ESQ1: 1 - ESQ

ESQI: 1/ESQ1

F: Array to store the functions f in the differential equation y' = f. In F(I,K), I denotes:

I = 1,2,3: inertial velocity components along x,y,z.

= 4,5,6: inertial acceleration components along x,y,z.

= 7,8: rate of change of angular momentum about y and z.

= 9,10,11,12,13,14: time derivative of the body-to-inertial rotation matrix.

GAMA: Current elevation angle (geocentric).

GAMAL: Launch elevation angle (geocentric).

GDLAT: Current latitude (geodetic).

GDLATL: Latitude at launch (geodetic).

GM: Product of the universal gravitational constant and the mass of the earth.

GR: Acceleration due to gravity at the current altitude.

GO: Acceleration due to gravity at the sea level.

HALTB: Geopotential altitude table for the temperature curve from 1962 U.S. Standard Atmosphere.

HLIM: Altitude from the surface at which the trajectory (impact) has to be terminated.

HPI: Half of π.

HPREV: Previous interpolation value used in atmosphere table.

IPAZ: Surface azimuth below the impact point.

IPR: Surface range below the impact point.

IPRINT: Option for printout of the trajectory variables:

= 0: summary only
= 1: detail trajectory with summary
= 2: debug option

IPT: Current time at the impact.

IPUNCH: Option for punchout of range, burnout, flight elevation, unit-wind effects, and delta f(z) curves. These are used as input to 5D wind weighting.

= 0: no punchout
≠ 0: will be punched out

IROT: Option to consider rotating or non-rotating earth:

= 0: non-rotating earth
≠ 0: rotating earth

ITAPE1: Disc number (=5 always) for inputting on cards the NAMELIST data.

ITAPE2: Disc number for inputting thrust and mach tables.

ITAPE3: Disc number on which processed vehicle tables are stored. The tables are cross-interpolated prior to the trajectory simulation phase for faster program execution. (The two tables with time as an independent variable are given a common time axis; the four tables whose independent variable is mach number are treated similarly.)

ITAPE4: Disc number to store data for spent stage calculations.

IYY: Pitch (= yaw) moment of inertia.

JS: Subscript for separation times.

JSPENT: Option for calculating trajectory for spent stage(s):

= 0: no calculation
≠ 0: will be calculated

KPREV: Previous subscript used in atmosphere interpolation subroutine.

KSP: Number of entries in the spent stage table for drag coefficient.

KSTEP: Number of integration steps required before doubling the step size.

LAT: Current latitude(geocentric).

LATL: Latitude at launch (geocentric).

LCG: Distance from nose to the c.g.

LONG: Current longitude.

LONGL: Longitude at launch.

LPREV: Array for storing previous subscript used in interpolation of thrust and drag tables.

MACH: Current mach number.

MASS: Current mass of the vehicle.

MDOT: Current rate of change of the propellant mass.

MLAST: Number of levels in wind tables.

MPL: Mass of the payload.

MPREV: Previous subscript used in the wind tables.

MPROP: Current mass of the propellant.

MSEP: Table of separation masses.

NAME: Name of the vehicle.

NANG: Number of entries in array ANG (launch elevation).

NCON: Option for initialization for various purposes:

- < 0: Initialize all variables if launch coordinates (latitude, longitude) are changed.
- = 0: Initialize all variables if launch settings (elevation, azimuth) are changed.
- = 0: Wind not present.
- > 0: Winds present.

NGOOD: Number of steps integrated at the current time without changing step size.

NLEV: Number of entries in the wind altitude table.

NM: Array for storing number of entries in array A.

NP: Current phase number.

NPL: Number of payloads.

NP1: NP + 1

NPMAX: Maximum number of phases that the program can take.

NPST: Total number of phases.

NSEP: Number of weights to be separated.

NSMAX: Maximum number of separations that the program can take.

NSP: Total number of separations.

NVAR: Maximum number of entries for array A.

OMEGA: Earth's rotation rate (rad/sec).

ONE: 1.0

ORDER: Array to store the degrees of polynomial for each A in cross-interpolating the time and Mach number tables.

PHI: Polar longitude (0° <PHI <360°).

PHIL: Polar longitude at launch (").

PHT: Array to store times at which a new phase begins.

PI: Constant π.

PLM: Array to store payload masses.

PSI: Current azimuth (0° <PSI <360°).

PSIL: Azimuth at launch (").

PRES: Current atmospheric pressure.

PRESB: Ratio of air pressure at a given altitude to sea level pressure (Ref. to 1962 U. S. Standard Tables.)

PRESO: Atmospheric pressure at sea level.

QPI: $\pi/4.0$

QS: Product of dynamic pressure and aerodynamic reference area.

R: Current distance from the center of the earth(inertial).

RE: Earth's radius at launch site.

RKSTEP: = 1: Integrations performed one step by Runge-Kutta method. Used near the phase termination boundary.

3: Integrations performed three steps by Runge-Kutta method. Three steps of Runge-Kutta are used in obtaining starting values for predictor-corrector.

RMASS: Total mass of the vehicle at the ignition minus payload.

RQ: Equatorial radius of the earth.

RTI: Array to store rotation matrix (from topocentric to inertial).

RXY: Projection of R on the inertial X-Y plane.

S: Array to store the integration variables: S(J)

J = 1,2,3: inertial position coordinates (x,y,z)

= 4,5,6: inertial velocity component (along x,y,z)

= 7: pitch angular momenta

= 8: yaw angular momenta

= 9,10,11,12,13,14: the elements of body-to-inertial rotation matrix at the current time.

SLOPE: Change in atmospheric temperature with altitude.

SP: Array to store the integration variable(s) for spent stages.

SPEED: Speed table for wind. (Wind direction is denoted by DIR, and altitude by ALTW).

SPM: Mass of the spent stage.

SOUND:	Current velocity of sound.
SOUND0:	Velocity of sound at sea level.
SRAZ:	Current surface azimuth (bearing).
SUBM:	Rocket mass minus propellant mass of a thrusting stage.
SY:	Angular rate about body y axis.
SZ:	Angular rate about body z axis.
T:	Current time.
TAV:	Array to store initial time, altitude and velocity.
TBO:	Time at the final burnout. Integration change from 5D to 3D here.
TEMP:	Current temperature in atmosphere.
TEMPB:	Temperature points from the 1962 U.S. Standard Atmosphere Tables.
TEMPO:	Temperature (atmospheric) at sea level.
TEPS:	Minimum time interval allowed for accuracy (not used in program).
THETA:	Polar latitude (0° < THETA <180°)
THETAL:	Polar latitude at launch ('').
THR:	Thrust at vacuum.
TNP1:	Time when the next phase begins.
TPI:	2π
TSEP:	Mass separation times.
TSLOPE:	Slope of temperature with respect to altitude.
TSP:	Times at spent stage separations.
T3D:	= TBO.
VA:	Components of air-relative velocity in body axes.
VEL:	Current relative velocity.
VELE:	Table of east components of wind velocity.
VELN:	Table of north components of wind velocity.
VLIM:	If the current VEL < VLIM then apogee is assumed.
VPREV:	Array for storing previous interpolation value in A.
VT:	Components of velocity in topocentric system.
V0LIM:	Minimum velocity at launch.
WE:	Current east component of the wind velocity.
WIND:	Magnitude of wind velocity.

WN: Current north component of the wind velocity.

WNAZ: Direction of wind velocity measured clockwise from North.

WPL: Weight of the payload.

X0: Inertial X component at launch.

Y0: Inertial Y component at launch.

ZERO: Practical value of zero = 1.0×10^{-15}.

Z0: Inertial Z component at launch.

REFERENCES

1. Hennigh, K.E.; "Field Wind Weighting and Impact Prediction for Unguided Rockets", NASA Technical Note (NASA TN D-2142), National Aeronautics and Space Administration, Washington, D.C., March 1964.

2. Everett, H.V. and McGregor, C.O.; "The Near-Earth Mission Analysis Routine (NEMAR), LA0076", Missiles and Space Division, LTV Aerospace Corporation, Dallas, Texas 75222.

3. Ralston, Anthony; "A First Course in Numerical Analysis", McGraw-Hill Book Co., New York, 1965.

4. Hildebrand, F.B.; "Introduction to Numerical Analysis", McGraw-Hill Book Co., New York, 1956.

APPENDIX A

INPUT FORMS

PRECEDING PAGE BLANK NOT FILMED

ORIGINAL PAGE IS
OF POOR QUALITY

FORTRAN CODING FORM

CARD CODES HOLLERITH

		=	3 – 8	¬	4 – 8		
		'	0 – 3 – 8	(	0 – 4 – 8		
–	11	$	11 – 3 – 8	*	11 – 4 – 8		
+	12		12 – 3 – 8	)	12 – 4 – 8		

PROBLEM: SENS-5D

CODEP: DATE: PAGE 1 of 7

C COMMENTS
V - VFAP
STATEMENT NUMBER | CONTINUATION | FORTRAN STATEMENT | IDENTIFICATION

```
$       IDENT   153202,SENS-5D          COMMENTS
$       EXECUTE DUMP
$       LIMITS          ,35K,,5K
$       TAPE    R*,X1D,,
$       FILE    02,X2R,,5L
$       FILE    03,X3R,,5L

$       DATA    05
$       INCODE  IBMF
  $DLIST  $

  $BLIST  $

  $FLIST  $

  $ULIST  $
```

FORTRAN CODING FORM

CARD CODES HOLLERITH

PROBLEM: SENS-5D

CODER | DATE | PAGE 2 of 7

```
$ DATA 01
$ INCODE IBMF
COMMENTS
ROCKET WEIGHT BURN OUT TIME
N, START TIMES FOR PHASES
N, START TIMES FOR SPENT STAGES
N, TABLE OF WEIGHTS FOR SPENT STAGES
```

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

		=	3 – 8	~		4 – 8	
			0 – 3 – 8	!	0 – 4 – 8		
–	11	$	11 – 3 – 8	*	11 – 4 – 8		
+	12		12 – 3 – 8	(	12 – 4 – 8		

PROBLEM: SENS-5D

CODEP | DATE | PAGE: 3 of 7

C = COMMENTS
V = VFAP
STATEMENT NUMBER | CONTINUATION | FORTRAN STATEMENT | IDENTIFICATION

```
PHASE NO. 1 BEGINS. COMMENTS
NOZZLE EXIT AREA, LENGTH FROM NOSE TO NOZZLE, REF. AREA, REF. DIAMETER
N, TIME TABLE FOR THRUST AND PROPELLANT WEIGHT
N, THRUSTS AT VACUUM
N, PROPELLANT WEIGHTS (NOT MASS)
```

FORTRAN CODING FORM

CARD CODES HOLLERITH

PROBLEM: SENS-5D

CODER | DATE | PAGE 4 of 7

C COMMENTS
V VFAP
STATEMENT NUMBER

FORTRAN STATEMENT

IDENTIFICATION

```
N, TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.

N, AXIAL DRAG COEFFS.

N, TIME TABLE FOR C.G. (FROM NOSE) DISTANCE AND PITCH MOMENT OF INERTIA

N, C.G. DISTANCE FROM NOSE

N, PITCH MOMENTS OF INERTIA

N, TABLE OF MACH NOS. FOR PITCH DAMPING COEFFS.
```

NASA WI-963 (1-68)

ORIGINAL PAGE IS
OF POOR QUALITY

FORTRAN CODING FORM

CARD CODES HOLLERITH

=	3 – 8	–	4 – 8
.	0 – 3 – 8	(	0 – 4 – 8
– 11	$ 11 – 3 – 8	*	11 – 4 – 8
+ 12	12 – 3 – 8	)	12 – 4 – 8

PROBLEM: SENS-5D

CODER | DATE | PAGE 5 of 7

C COMMENTS / V VFAP / STATEMENT NUMBER	FORTRAN STATEMENT	IDENTIFICATION
N,	PITCH DAMPING COEFFS.	
N,	TABLE OF MACH NOS. FOR SLOPE-OF-NORMAL FORCE COEFFS.	
N,	SLOPE OF NORMAL FORCE COEFFS.	
N,	TABLE OF MACH NOS. FOR C.P. (FROM NOSE) DISTANCES	
N,	C.P. DISTANCES FROM NOSE	

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

PROBLEM: SENS-5D

CODER | DATE | PAGE 6 of 7

C COMMENTS
V VERIFY
STATEMENT NUMBER | CONTINUATION | FORTRAN STATEMENT | IDENTIFICATION

```
SPENT STAGE NO. BEGINS. COMMENTS
TIME OF SEPARATION, SEPARATED WEIGHT, REF. AREA

N, TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.

N, AXIAL DRAG COEFFS.
```

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

PROBLEM: SENS-5D

CODER | DATE | PAGE 7 of 7

C COMMENTS V VEAP STATEMENT NUMBER	FORTRAN STATEMENT	IDENTIFICATION
$	ENDJOB	
* * * EOF		

NASA WI 963 (1-68)

APPENDIX B

SAMPLE INPUT

ORIGINAL PAGE IS
OF POOR QUALITY

FORTRAN CODING FORM

CARD CODES HOLLERITH

PROBLEM: SENS-5D

CODEP: R. SINGH DATE: JAN. 14, 1975 PAGE 1 of 7

C - COMMENTS
V - VFAP
STATEMENT NUMBER

FORTRAN STATEMENT

IDENTIFICATION

```
$      IDENT    153202, SENS-5D      NIKE CAJUN
$      EXECUTE  DUMP
$      LIMITS   100,35K,,5K
$      TAPE     R*,X1D,,0334
$      FILE     02,X2R,5L
$      FILE     03,X3R,5L

$      DATA     05
$      INCODE   IBMF
$DLIST  AZERO=20, IPUNCH=1  $

$BLIST  NPL=1, PLM=50, AZGDL=120, NANG=11, ANG=70,72,74,76,78,80,82,84,
        86,88,90, JSPENT=1  $

$FLIST  WPL=50, AZGDL=90, ELGDL=80, WIND=20, WNAZ=300,
        NLEV=12, ALEV=0,50,100,150,200,250,300,1000,3000,10000,30000,
        60000 $

$VLIST  WPL=50, AZGDL=90, NANG=11, ANG=70,72,74,76,78,80,82,84,86,88,90,
        WIND=20, AHIGH=60000 $
```

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

PROBLEM: S-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 2 of 7

FORTRAN STATEMENT

IDENTIFICATION

```
$       DATA    01
$       INCODE  IBMF
NIKE    CAT:N
ROCKET  WEIGHT, BURN-OUT TIME
  1519.0, 22.0
N, START TIMES FOR PHASES
  4,  -0.017, 3.523, 17.0, 22.0

N, START TIMES FOR SPENT STAGES
  1,  3.523

N, TABLE OF WEIGHTS FOR SPENT STAGES
  1,  564.0
```

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

	=	3 – 8	–	4 – 8	
	.	0 – 3 – 8	(	0 – 4 – 8	
–	11	$	11 – 3 – 8	*	11 – 4 – 8
+	12	.	12 – 3 – 8	)	12 – 4 – 8

PROBLEM: ENS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 3 of 7

COMMENTS / STATEMENT NUMBER	CONTINUATION	FORTRAN STATEMENT	IDENTIFICATION
PHASE		NO. 1 BEGINS. COMMENTS THIS IS THRUSTING PHASE	
NOZZLE		EXIT AREA, LENGTH FROM NOSE TO NOZZLE, REF. AREA, REF. DIAMETER	
1.5,		27.5, 1.474, 1.375	
N, TIME		TABLE FOR THRUST AND PROPELLANT WEIGHT	
22,		0.0, 0.01, 0.04, 0.05, 0.09, 0.15, 0.84, 1.14, 1.74, 2.04, 2.34	
		2.49, 2.64, 2.79, 2.99, 3.09, 3.21, 3.28, 3.34, 3.40, 3.46, 3.54	
N, THRUSTS		AT ~~VACUUM~~ SEA LEVEL	
22,		1923.0, 26439.0, 39417.0, 41628.0, 42397.0, 42589.0, 43263.0	
		43551.0, 44705.0, 45667.0, 46820.0, 46723.0, 45570.0, 42974.0	
		37975.0, 33648.0, 24034.0, 16343.0, 10864.0, 6538.0, 3268.0, 0.0	
N, PROPELLANT		WEIGHTS (NOT MASS)	
22,		738.0, 737.26, 732.10, 729.98, 721.20, 707.87, 553.07, 485.01	
		346.63, 275.78, 203.27, 166.6, 130.42, 95.72, 53.41, 34.69, 16.60	
		9.22, 4.95, 2.22, 0.68, 0.0	

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

		=	3 – 8	–	4 – 8
		.	0 – 3 – 8	(	0 – 4 – 8
–	11	$	11 – 3 – 8	*	11 – 4 – 8
+	12	.	12 – 3 – 8	)	12 – 4 – 8

PROBLEM: NS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 4 of 7

C - COMMENTS
V - VIAP
STATEMENT NUMBER | CONTINUATION | FORTRAN STATEMENT | IDENTIFICATION

```
N, TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.
  11, 0.0, 0.75, 1.0, 1.20, 1.60, 2.0, 2.4, 2.8, 3.3, 4.0, 5.0

N, AXIAL DRAG COEFFS.
  11, 0.675, 0.595, .925, .87, .78, .71, .66, .615, .565, .52, .455

N, TIME TABLE FOR C.G. (FROM NOSE) DISTANCE AND PITCH MOMENT OF INERTIA
  2, 0., 3.54

N, C.G. DISTANCE FROM NOSE
  2, 17.583, 16.0

N, PITCH MOMENTS OF INERTIA
  2, 1530., 1255.

N, TABLE OF MACH NOS. FOR PITCH DAMPING COEFFS.
  12, 0., .9, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 20.0
```

NASA WI-963 (1-68)

ORIGINAL PAGE IS
OF POOR QUALITY

FORTRAN CODING FORM

CARD CODES HOLLERITH

	=	3 – 8	–	4 – 8	
	.	0 – 3 – 8	[	0 – 4 – 8	
–	11	$	11 – 3 – 8	*	11 – 4 – 8
+	12	.	12 – 3 – 8	]	12 – 4 – 8

PROBLEM: IS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 5 of 7

C = COMMENTS, V = STAP, STATEMENT NUMBER | CONTINUATION | FORTRAN STATEMENT | IDENTIFICATION

```
N, PITCH DAMPING COEFFS.
12, 15.88, 18.85, 19.72, 21.99, 14.66, 11.35, 8.9, 8.03, 7.33, 6.98
    6.98, 6.98

N, TABLE OF MACH NOS. FOR SLOPE-OF-NORMAL FORCE COEFFS.
12, 0.0, 0.9, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 20.0

N, SLOPE OF NORMAL FORCE COEFFS.
12, 0.27, .27, .325, .24, .20, .175, .155, .14, .13, .125, .12, .12

N, TABLE OF MACH NOS. FOR C.P. (FROM NOSE) DISTANCES
12, 0., 0.9, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 20.0

N, C.P. DISTANCES FROM NOSE
12, 20, 20, 19.583, 19.333, 19.167, 18.75, 17.917, 17.5, 17.083, 16.25, 15.833
    15.833
```

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

=	3 – 8	–	4 – 8
,	0 – 3 – 8	(	0 – 4 – 8
– 11	$ 11 – 3 – 8	*	11 – 4 – 8
+ 12	12 – 3 – 8	)	12 – 4 – 8

PROBLEM: ENS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 3 of 7

COMMENTS / STATEMENT NUMBER	CONTINUATION	FORTRAN STATEMENT	IDENTIFICATION
PHASE NO. 2 BEGINS. COMMENTS THIS IS COASTING PHASE.			
NOZZLE EXIT AREA, LENGTH FROM NOSE TO NOZZLE, REF. AREA, REF. DIAMETER			
0.0, 0.0, 0.230, 0.54			
N, TIME TABLE FOR THRUST AND PROPELLANT WEIGHT			
1, 0.			
N, THRUSTS AT VACUUM			
1, 0.			
N, PROPELLANT WEIGHTS (NOT MASS)			
1, 0.			

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

		=	3 – 8	–		4 – 8	
		'	0 – 3 – 8	(	0 – 4 – 8		
–	11	$	11 – 3 – 8	*	11 – 4 – 8		
+	12		12 – 3 – 8	)	12 – 4 – 8		

PROBLEM: NS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 4 of 7

C = COMMENTS V = TAP	STATEMENT NUMBER	CONTINUATION	FORTRAN STATEMENT	IDENTIFICATION
N			TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.	
	7	1	10, 20, 30, 40, 50, 60, 80	
N			AXIAL DRAG COEFFS.	
	7	0	.93, .72, .59, .51, .45, .42, .36	
N			TIME TABLE FOR C.G. (FROM NOSE) DISTANCE AND PITCH MOMENT OF INERTIA	
	1	0		
N			C.G. DISTANCE FROM NOSE	
	1	7	7.83	
N			PITCH MOMENTS OF INERTIA	
	1	2	0.5	
N			TABLE OF MACH NOS. FOR PITCH DAMPING COEFFS.	
	9	1	.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0	

NASA WI-003 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

PROBLEM: IS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 5 of 7

C = COMMENTS, V = VFAP, STATEMENT NUMBER, CONTINUATION

FORTRAN STATEMENT

IDENTIFICATION

```
N, PITCH DAMPING COEFFS.
9,  174.4, 157., 122.1, 104.6, 88.94, 78.48, 69.76, 69.76, 69.76

N, TABLE OF MACH NOS. FOR SLOPE-OF-NORMAL FORCE COEFFS.
9,  1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0

N, SLOPE OF NORMAL FORCE COEFFS.
9,  0.58, .47, .33, .26, .22, .19, .17, .16, .16

N, TABLE OF MACH NOS. FOR C.P. (FROM NOSE) DISTANCES
9,  1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0

N, C.P. DISTANCES FROM NOSE
9,  11.042, 10.833, 10.333, 9.917, 9.5, 9.167, 8.875, 8.625, 8.625
```

NASA WI-963 (1-68)

ORIGINAL PAGE IS
OF POOR QUALITY

FORTRAN CODING FORM

CARD CODES HOLLERITH

PROBLEM: ENS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 3 of 7

C = COMMENTS
V = LEAP
STATEMENT NUMBER

FORTRAN STATEMENT

IDENTIFICATION

```
PHASE NO.3 BEGINS. COMMENTS THIS IS THRUSTING PHASE
NOZZLE EXIT AREA, LENGTH FROM NOSE TO NOZZLE, REF. AREA, REF. DIAMETER
  0.223,15.083,0.230,0.542
N, TIME TABLE FOR THRUST AND PROPELLANT WEIGHT
  21,  0.,0.04,.08,.15,.5,.7,1.,1.2,1.4,1.6,2.,2.4,2.8,2.98
       3.04,3.16,3.26,3.4,3.5,5.,6.0

N, THRUSTS AT VACUUM
  21,  471.,2521.,7871.,7671.,7546.,7671.,8171.,8521.,8746.
       8871.,8971.,9196.,9841.,9521.,7371.,3171.,1371.,696.,521.
       471.,0.0

N, PROPELLANT WEIGHTS (NOT MASS)
  21,  119.,118.8,117.91,115.5,103.73,97.0,86.46,79.03,71.34
       63.48,47.54,31.30,14.24,6.43,4.17,1.46,.61,.24,.17,0.,0.
```

NASA WI-903 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

		=	3 – 8			–	4 – 8
		:	0 – 3 – 8			(	0 – 4 – 8
–	11	$	11 – 3 – 8			*	11 – 4 – 8
+	12		12 – 3 – 8			)	12 – 4 – 8

PROBLEM: NS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 4 of 7

C COMMENTS V = AP	STATEMENT NUMBER	CONTINUATION	FORTRAN STATEMENT	IDENTIFICATION
N			TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.	
	8	1	1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 8.0, 20.0	
N			AXIAL DRAG COEFFS.	
	8	0	.78, .62, .53, .46, .42, .38, .34, .34	
N			TIME TABLE FOR C.G. (FROM NOSE) DISTANCE AND PITCH MOMENT OF INERTIA	
	2	0	0.0, 5.0	
N			C.G. DISTANCE FROM NOSE	
	2	7	.783, 7.158	
N			PITCH MOMENTS OF INERTIA	
	2	1	20.5, 96.3	
N			TABLE OF MACH NOS. FOR PITCHING COEFFS.	
	9	1	.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0	

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

PROBLEM: VS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 5 of 7

C - COMMENTS
V - VFAP
STATEMENT NUMBER

FORTRAN STATEMENT

IDENTIFICATION

```
N, PITCH DAMPING COEFFS.
9,  174.4, 157., 122.1, 104.6, 88.94, 78.48, 69.76, 69.76, 69.76

N, TABLE OF MACH NOS. FOR SLOPE-OF-NORMAL FORCE COEFFS.
9,  1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0

N, SLOPE OF NORMAL FORCE COEFFS.
9,  0.58, .47, .33, .26, .22, .19, .17, .16, .16

N, TABLE OF MACH NOS. FOR C.P. (FROM NOSE) DISTANCES
9,  1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0

N, C.P. DISTANCES FROM NOSE
9,  11.042, 10.833, 10.333, 9.917, 9.5, 9.167, 8.875, 8.625, 8.625
```

NASA WI-983 (1-66)

FORTRAN CODING FORM

CARD CODES HOLLERITH

=	3 – 8	–	4 – 8
.	0 – 3 – 8	(	0 – 4 – 8
– 11	$ 11 – 3 – 8	*	11 – 4 – 8
+ 12	12 – 3 – 8	)	12 – 4 – 8

PROBLEM: ENS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 3 of 7

C = COMMENTS V = MAP STATEMENT NUMBER	CONTINUATION	FORTRAN STATEMENT	IDENTIFICATION
PHASE		NO. 4 BEGINS. COMMENTS. THIS IS COASTING AND LAST PHASE	
NOZZLE		EXIT AREA, LENGTH FROM NOSE TO NOZZLE, REF. AREA, REF. DIAMETER	
0.0,		0.0, 0.230, 0.0	
N, TIME		TABLE FOR THRUST AND PROPELLANT WEIGHT	
1, 0.			
N, THRUSTS		AT VACUUM	
1, 0.			
N, PROPELLANT		WEIGHTS (NOT MASS)	
1, 0.			

NASA WI-903 (1-68)

ORIGINAL PAGE IS
OF POOR QUALITY

FORTRAN CODING FORM

CARD CODES HOLLERITH

=	3 – 8	–	4 – 8
'	0 – 3 – 8	(	0 – 4 – 8
– 11	$	11 – 3 – 8	*	11 – 4 – 8
+ 12		12 – 3 – 8	)	12 – 4 – 8

PROBLEM: NS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 4 of 7

FORTRAN STATEMENT

```
N, TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.
8,  1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 8.0, 20.0

N, AXIAL DRAG COEFFS.
8,  0.93,  .72,  .59,  .51,  .45,  .42,  .36,  .36

N, TIME TABLE FOR C.G. (FROM NOSE) DISTANCE AND PITCH MOMENT OF INERTIA
1, 0.

N, C.G. DISTANCE FROM NOSE
1, 0.

N, PITCH MOMENTS OF INERTIA
1, 0.

N, TABLE OF MACH NOS. FOR PITCH DAMPING COEFFS.
1, 0.
```

IDENTIFICATION

NASA W1-963 (3-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

	=	3 – 8	–	4 – 8	
		0 – 3 – 8	(	0 – 4 – 8	
– 11	$	11 – 3 – 8	*	11 – 4 – 8	
+ 12		12 – 3 – 8	)	12 – 4 – 8	

PROBLEM: JS-5D

CODER: R. SINGH

DATE: JAN. 14, 1975

C = COMMENTS
V = VERIFY
STATEMENT NUMBER | CONTINUATION | FORTRAN STATEMENT | IDENTIFICATION

```
N, PITCH DAMPING COEFFS.
  1, 0.

N, TABLE OF MACH NOS. FOR SLOPE-OF-NORMAL FORCE COEFFS.
  1, 0.

N, SLOPE OF NORMAL FORCE COEFFS.
  1, 0.

N, TABLE OF MACH NOS. FOR C.P. (FROM NOSE) DISTANCES
  1, 0.

N, C.P. DISTANCES FROM NOSE
  1, 0.
```

ORIGINAL PAGE IS
OF POOR QUALITY

FORTRAN CODING FORM

CARD CODES HOLLERITH

	=	3 – 8	–	4 – 8	
	.	0 – 3 – 8	(	0 – 4 – 8	
– 11	$	11 – 3 – 8	*	11 – 4 – 8	
+ 12	.	12 – 3 – 8	)	12 – 4 – 8	

PROBLEM: NS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 6 of 7

C COMMENTS V-VERIFY STATEMENT NUMBER | CONTINUATION | FORTRAN STATEMENT | IDENTIFICATION

```
SPENT STAGE NO. 1 BEGINS. COMMENTS
TIME OF SEPARATION, SEPARATED WEIGHT, REF. AREA
  3,523, 564.0, 0.0
N, TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.
  2,  0., 20.0

N, AXIAL DRAG COEFFS.
  2,  0., 0.
```

NASA WI-963 (1-68)

FORTRAN CODING FORM

CARD CODES HOLLERITH

	=	3 – 8	–	4 – 8	
	'	0 – 3 – 8	(	0 – 4 – 8	
– 11	$	11 – 3 – 8	*	11 – 4 – 8	
+ 12		12 – 3 – 8	)	12 – 4 – 8	

PROBLEM: NS-5D

CODER: R. SINGH DATE: JAN. 14, 1975 PAGE: 7 of 7

COMMENTS / VERSION / STATEMENT NUMBER (1–5)	CONTINUATION (6)	FORTRAN STATEMENT (7–72)	IDENTIFICATION (73–80)
$		ENDJOB	
* * * EOF			

NASA WI-953 (1-68)

APPENDIX C

SAMPLE OUTPUT

SENS-50 CALCULATIONS BEGIN

INPUT DATA SET NO. 1

```
$DLIST  AZERO=20, IPUNCH=1  $
$BLIST  NPL=1,PLM=50, AZGDL=120, NANG=11,ANG=70,72,74,76,78,80,82,84,
        86,88,90, JSPENT=1  $
$FLIST  WPL=50, AZGDL=90, ELGDL=80, WIND=20,WNAZ=300,
        NLEV=12, ALEV=0,50,100,150,200,250,300,1000,3000,40000,30000,
        60000 $
$ULIST  WPL=50,AZGDL=90, NANG=11,ANG=70,72,74,76,78,80,82,84,86,88,90,
        WIND=20,AHIGH=60000 $
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
INPUT DATA SET NO. 2

NIKE CAJUN
ROCKET WEIGHT, BURNOUT TIME
 1519.0, 22.0
N, START TIMES FOR PHASES
 4,  -0.017, 3.523, 17.0, 22.0
N, START TIMES FOR SPENT STAGES
 1,  3.523
N, TABLE OF WEIGHTS FOR SPENT STAGES
 1,  564.0
PHASE NO.1  BEGINS, COMMENTS  THIS IS THRUSTING PHASE
NOZZLE EXIT AREA, LENGTH FROM NOSE TO NOZZLE, REF. AREA, REF. DIAMETER
 1.5, 27.5, 1.474, 1.375
N, TIME TABLE FOR THRUST AND PROPELLANT WEIGHT
 22, 0.0, 0.01, 0.04, 0.05, 0.09, 0.15, 0.84, 1.14, 1.34, 2.04, 2.34
     2.69, 2.64, 2.79, 2.99, 3.09, 3.21, 3.29, 3.34, 3.40, 3.46, 3.54
N, THRUSTS AT  SEA LEVEL
 22, 1923.0, 26439.0, 39417.0, 41628.0, 42397.0, 42589.0, 43263.0
     43581.0, 44705.0, 45667.0, 46820.0, 46723.0, 45570.0, 42974.0
     37975.0, 33648.0, 24034.0, 16343.0, 10864.0, 6588.0, 3268.0, 0.0
N, PROPELLANT WEIGHTS (NOT MASS)
 22, 738.0, 737.26, 732.10, 729.98, 721.20, 707.87, 553.07, 485.01
     346.63, 275.78, 203.27, 166.6, 130.42, 95.72, 53.41, 34.69, 16.60
     9.22, 4.95, 2.22, 0.68, 0.0
N, TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.
 11, 0.0, 0.75, 1.0, 1.20, 1.60, 2.0, 2.4, 2.8, 3.0, 4.0, 5.0
N, AXIAL DRAG COEFFS.
 11, 0.675, 0.595, .925, .87, .78, .71, .66, .615, .565, .52, .455
N, TIME TABLE FOR C.G. (FROM NOSE) DISTANCE AND PITCH MOMENT OF INERTIA
 2,  0., 3.54
N, C.G. DISTANCE FROM NOSE
 2,  17.383, 16.0
N, PITCH MOMENTS OF INERTIA
 2,  1530., 1255.
N, TABLE OF MACH NOS. FOR PITCH DAMPING COEFFS.
 12, 0., .9, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 20.0
N, PITCH DAMPING COEFFS.
 12, 15.88, 18.85, 19.72, 21.99, 14.66, 11.35, 8.9, 8.03, 7.33, 6.98
     6.98, 6.98
N, TABLE OF MACH NOS. FOR SLOPE-OF-NORMAL FORCE COEFFS.
 12, 0.8, 0.9, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 20.0
N, SLOPE OF NORMAL FORCE COEFFS.
 12, 0.27, .27, .329, .24, .20, .175, .155, .14, .13, .125, .12, .12
N, TABLE OF MACH NOS. FOR C.P. (FROM NOSE) DISTANCES
```

```
 12, 0., 0.9, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 20.0
N, C.P. DISTANCES FROM NOSE
 12, 20,20,19.583,19.333,19.167,18.75,17.917,17.5,17.083,16.25,15.833
     15.833
PHASE NO.2   BEGINS, COMMENTS  THIS IS COASTING PHASE,
NOZZLE EXIT AREA, LENGTH FROM NOSE TO NOZZLE, REF. AREA, REF. DIAMETER
 0,0,0,0,0.230,0.54
N, TIME TABLE FOR THRUST AND PROPELLANT WEIGHT
 1, 0,
N, THRUSTS AT VACUUM
 1, 0,
N, PROPELLANT WEIGHTS (NOT MASS)
 1, 0,
N, TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.
 7, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 8.0
N, AXIAL DRAG COEFFS.
 7, 0.93, .72, .59, .51, .45, .42, .36
N, TIME TABLE FOR C.G. (FROM NOSE) DISTANCE AND PITCH MOMENT OF INERTIA
 1, 0,
N, C.G. DISTANCE FROM NOSE
 1, 7.783
N, PITCH MOMENTS OF INERTIA
 1, 120.5
N, TABLE OF MACH NOS. FOR PITCH DAMPING COEFFS.
 9, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0
N, PITCH DAMPING COEFFS.
 9, 174.4, 157., 122.1, 104.6, 88.94, 78.48, 69.76, 69.76, 69.76
N, TABLE OF MACH NOS. FOR SLOPE-OF-NORMAL FORCE COEFFS.
 9, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0
N, SLOPE OF NORMAL FORCE COEFFS.
 9, 0.58, .47, .33, .26, .22, .19, .17, .16, .16
N, TABLE OF MACH NOS. FOR C.P. (FROM NOSE) DISTANCES
 9, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0
N, C.P. DISTANCES FROM NOSE
 9, 11.042,10.833,10.333,9.917,9.5,9.167,8.875,8.625,8.625
PHASE NO.3   BEGINS, COMMENTS    THIS IS  THRUSTING PHASE
NOZZLE EXIT AREA, LENGTH FROM NOSE TO NOZZLE, REF. AREA, REF. DIAMETER
 0.223,19.083,0.230,0.542
N, TIME TABLE FOR THRUST AND PROPELLANT WEIGHT
 21, 0., 0.04, .08, .15, .5, .7, 1., 1.2, 1.4, 1.6, 2., 2.4, 2.8, 2.98
    3.04, 3.16, 3.26, 3.4, 3.5, 5., 6.0
N, THRUSTS AT VACUUM
 21, 471., 2521., 7871., 7671., 7546., 7671., 8171., 8521., 8746.,
     8871., 8971., 9196., 9641., 9521., 7371., 3171., 1571., 696., 521.,
```

```
     471., 0.0
N, PROPELLANT WEIGHTS (NOT MASS)
 21, 119., 118.8, 117.91, 115.2, 103.73, 97.0, 86.46, 79.03, 71.34
     63.48, 47.54, 31.30, 14.84, 6.43, 4.17, 1.46, .61, .24, .17, 0., 0.
N, TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.
 8, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 8.0, 20.0
N, AXIAL DRAG COEFFS.
 8, 0.78, .62, .53, .46, .42, .38, .34, .34
N, TIME TABLE FOR C.G. (FROM NOSE) DISTANCE AND PITCH MOMENT OF INERTIA
 2, 0.0, 5.0
N, C.G. DISTANCE FROM NOSE
 2, 7.783, 7.158
N, PITCH MOMENTS OF INERTIA
 2, 120.5, 96.3
N, TABLE OF MACH NOS. FOR PITCH DAMPING COEFFS.
 9, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0
N, PITCH DAMPING COEFFS.
 9, 174.4, 157., 122.1, 104.6, 88.94, 78.48, 69.76, 69.76, 69.76
N, TABLE OF MACH NOS. FOR SLOPE-OF-NORMAL FORCE COEFFS.
 9, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0
N, SLOPE OF NORMAL FORCE COEFFS.
 9, 0.58, .47, .33, .26, .22, .19, .17, .16, .16
N, TABLE OF MACH NOS. FOR C.P. (FROM NOSE) DISTANCES
 9, 1.5, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 20.0
N, C.P. DISTANCES FROM NOSE
 9, 11.042,10.833,10.333,9.917,9.5,9.167,8.875,8.625,8.625
PHASE NO.4   BEGINS, COMMENTS,  THIS IS COASTING AND LAST PHASE
NOZZLE EXIT AREA, LENGTH FROM NOSE TO NOZZLE, REF. AREA, REF. DIAMETER
 0.0,0.0,0.230,0.0
N, TIME TABLE FOR THRUST AND PROPELLANT WEIGHT
 1, 0.
N, THRUSTS AT VACUUM
 1, 0.
N, PROPELLANT WEIGHTS (NOT MASS)
 1, 0.
N, TABLE OF MACH NOS. FOR AXIAL DRAG COEFFS.
 8, 1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 8.0, 20.0
N, AXIAL DRAG COEFFS.
 8, 0.93, .72, .59, .51, .45, .42, .36, .36
N, TIME TABLE FOR C.G. (FROM NOSE) DISTANCE AND PITCH MOMENT OF INERTIA
 1, 0.
N, C.G. DISTANCE FROM NOSE
 1, 0.
N, PITCH MOMENTS OF INERTIA
```

```
 1, 0,
N, TABLE OF MACH NOS, FOR PITCH DAMPING COEFFS,
 1, 0,
N, PITCH DAMPING COEFFS,
 1, 0,
N, TABLE OF MACH NOS, FOR SLOPE-OF-NORMAL FORCE COEFFS,
 1, 0,
N, SLOPE OF NORMAL FORCE COEFFS,
 1, 0,
N, TABLE OF MACH NOS, FOR C.P. (FROM NOSE) DISTANCES
 1, 0,
N, C.P. DISTANCES FROM NOSE
 1, 0,
SPENT STAGE NO.1   BEGINS, COMMENTS
TIME OF SEPARATION, SEPARATED WEIGHT, REF. AREA
 3.523, 9640.0, 0.0
N, TABLE OF MACH NOS, FOR AXIAL DRAG COEFFS,
 2, 0., 20.0
N, AXIAL DRAG COEFFS,
 2, 0., 0.
```

ORIGINAL PAGE IS OF POOR QUALITY

NASA WALLOPS FLIGHT CENTER
WALLOPS ISLAND, VIRGINIA

TRAJECTORY SUMMARY AT BURNOUT, APOGEE AND IMPACT

VEHICLE = NIKE CAJUN
PAY LOAD = 50.00 LBS
LAUNCH AZ = 120.00 DEG
WIND = ZERO
EARTH = ROTATING MODEL

	APOGEE			BURNOUT						IMPACT		
EL (DEG)	TIME (SEC)	ALT (FT)	RANGE (NM)	TIME (SEC)	ALT (FT)	RANGE (NM)	VEL (FT/SEC)	FLT/EL (DEG)	FLT/AZ (DEG)	TIME (SEC)	RANGE (NM)	AZ (DEG)
70.00	177.94	429521.	63.9	22.00	55920.	4.1	5748.33	63.30	119.94	350.82	128.2	121.18
72.00	182.59	452973.	59.8	22.00	56985.	3.7	5765.00	65.96	119.91	359.64	120.0	121.25
74.00	186.62	474109.	54.9	22.00	57930.	3.3	5776.98	68.63	119.87	367.39	110.2	121.33
76.00	190.19	493497.	49.4	22.00	58785.	2.9	5789.34	71.31	119.82	374.36	99.1	121.42
78.00	193.37	510755.	43.2	22.00	59532.	2.5	5799.81	74.00	119.76	380.39	86.8	121.54
80.00	196.05	525976.	36.6	22.00	60165.	2.1	5808.36	76.70	119.68	385.44	73.4	121.69
82.00	198.17	537708.	29.4	22.00	60684.	1.7	5815.22	79.40	119.55	389.52	59.2	121.90
84.00	199.76	546997.	22.0	22.00	61087.	1.3	5820.49	82.11	119.35	392.61	44.2	122.24
86.00	200.85	553482.	14.3	22.00	61372.	0.8	5824.05	84.82	118.94	394.70	28.7	122.95
88.00	201.46	557308.	6.4	22.00	61539.	0.4	5826.15	87.53	117.62	395.79	12.9	125.35
90.00	201.54	558125.	1.6	22.00	61589.	0.0	5826.76	89.73	323.35	395.90	3.2	283.22

NASA WALLOPS FLIGHT CENTER
WALLOPS ISLAND, VIRGINIA

TRAJECTORY SUMMARY FOR SPENT STAGES,

VEHICLE = NIKE CAJUN
PAY LOAD = 50.00 LBS (PARENT VEHICLE)
LAUNCH AZ = 120.00 DEG (+)
WIND = ZERO
EARTH = ROTATING MODEL

		SEPARATION								IMPACT		
LAUNCH EL (DEG)	NO.	TIME (SEC)	WEIGHT (SLUG)	ALT (FT)	RANGE (NM)	AZ (DEG)	VEL (FT/SEC)	FLT/EL (DEG)	FLT/AZ (DEG)	TIME (SEC)	RANGE (NM)	AZ (DEG)
70.00	1	3.52	564.00	5762.	0.37	119.63	3308.48	68.29	119.62	199.64	39.27	120.59
72.00	1	3.52	564.00	5842.	0.33	119.57	3308.30	70.46	119.58	202.32	35.93	120.61
74.00	1	3.52	564.00	5911.	0.30	119.51	3308.08	72.64	119.52	204.72	32.40	120.64
76.00	1	3.52	564.00	5974.	0.26	119.44	3307.96	74.82	119.44	206.84	28.69	120.66
78.00	1	3.52	564.00	6030.	0.23	119.31	3307.83	77.00	119.34	208.67	24.81	120.70
80.00	1	3.52	564.00	6076.	0.19	119.14	3307.72	79.18	119.19	210.21	20.80	120.74
82.00	1	3.52	564.00	6115.	0.15	118.90	3307.62	81.37	118.98	211.45	16.67	120.80
84.00	1	3.52	564.00	6145.	0.11	118.62	3307.56	83.55	118.62	212.40	12.45	120.90
86.00	1	3.52	564.00	6164.	0.07	117.76	3307.48	85.74	117.90	213.05	8.16	121.09
88.00	1	3.52	564.00	6177.	0.04	115.58	3307.45	87.92	115.67	213.40	3.83	121.73
90.00	1	3.52	564.00	6180.	0.00	355.31	3307.44	89.80	353.52	213.45	0.53	291.93

NASA WALLOPS FLIGHT CENTER
WALLOPS ISLAND, VIRGINIA

F(Z)-CURVE AND BALLISTIC WIND FACTORS

VEHICLE = NIKE CAJUN
PAY LOAD = 50.00 LBS
LAUNCH EL = 80.00 DEG
LAUNCH AZ = 90.00 DEG
WIND = 20.00 FT/SEC (6.10 M/SEC) 300.00 DEG AZ FROM NORTH
EARTH = NON-ROTATING MODEL

Z	WIND ALT (FT)	WIND ALT (M)	IMPACT RANGE (NM)	IMPACT RANGE (KM)	F(Z)	DF(Z) F(Z)-F(Z=1)	WIND ALT (FT)	WIND ALT (M)
1	0.	0.	75.69	140.18	0.	0.	0.	0.
2	50.	15.	73.90	136.87	0.13472	0.13472	50.	15.
3	100.	30.	72.35	134.00	0.25345	0.11874	100.	30.
4	150.	46.	71.21	131.87	0.33946	0.08600	150.	46.
5	200.	61.	70.31	130.22	0.40820	0.06874	200.	61.
6	250.	76.	69.72	129.13	0.45358	0.04539	250.	76.
7	300.	91.	69.17	128.11	0.49609	0.04251	300.	91.
8	1000.	305.	66.07	122.37	0.74018	0.24408	1000.	305.
9	3000.	914.	64.23	118.95	0.88880	0.14863	3000.	914.
10	10000.	3048.	63.90	118.34	0.91596	0.02716	10000.	3048.
11	30000.	9144.	64.47	119.40	0.87196	-0.04401	30000.	9144.
12	60000.	18288.	62.88	116.46	1.00000	0.12804	60000.	18288.

NASA WALLOPS FLIGHT CENTER
WALLOPS ISLAND, VIRGINIA

UNIT-WIND EFFECTS, CORIOLIS DEFLECTION AND RANGE DERIVATIVE

VEHICLE = NIKE CAJUN
PAY LOAD = 50.00 LBS
LAUNCH AZ = 90.00 DEG
WIND = 20.00 FT/SEC (6.10 M/SEC),
EARTH = NON-ROTATING MODEL

F-P-S SYSTEM

LAUNCH EL (DEG)	NO-WIND RANGE(NM)	IMPACT AZ(DEG)	CORIOLIS NORTH(NM)	DEFLECTION EAST(NM)	RANGE DERIV (NM/DEG)	UNIT WIND EFFECTS HEAD(NM/FPS)	TAIL(NM/FPS)	CROSS(NM/FPS)
70.00	128.69	89.49	-2.65	-0.58	-3.18931	0.30895	-0.40667	0.74411
72.00	120.89	89.43	-2.60	-1.06	-4.21265	0.40376	-0.50130	0.77422
74.00	111.56	89.35	-2.52	-1.35	-4.94776	0.49411	-0.58509	0.80140
76.00	100.85	89.25	-2.39	-1.78	-5.45032	0.57858	-0.65790	0.82562
78.00	88.77	89.12	-2.23	-2.03	-6.34817	0.65901	-0.72564	0.84603
80.00	75.69	88.94	-2.04	-2.33	-6.72390	0.72354	-0.78550	0.86415
82.00	61.70	88.68	-1.82	-2.59	-7.16007	0.78922	-0.83258	0.88042
84.00	46.95	88.23	-1.58	-2.80	-7.49689	0.83186	-0.86762	0.89212
86.00	31.66	87.34	-1.31	-2.98	-7.70074	0.86653	-0.89372	0.89890
88.00	15.98	84.68	-1.04	-3.08	-7.87882	0.89329	-0.90552	0.90347
90.00	1.48	360.00	-0.75	-3.11	-7.96376	0.90417	-0.90612	0.90489

M-K-S SYSTEM

LAUNCH EL (DEG)	NO-WIND RANGE(KM)	IMPACT AZ(DEG)	CORIOLIS NORTH(KM)	DEFLECTION EAST(KM)	RANGE DERIV (KM/DEG)	UNIT WIND EFFECTS HEAD(KM/MPS)	TAIL(KM/MPS)	CROSS(KM/MPS)
70.00	238.34	89.49	-4.91	-1.07	-5.90660	1.87721	-2.47100	4.52131
72.00	223.89	89.43	-4.82	-1.96	-7.80182	2.45326	-3.04597	4.70424
74.00	206.61	89.35	-4.66	-2.49	-9.16326	3.00228	-3.55509	4.86942
76.00	186.73	89.25	-4.43	-3.29	-10.09399	3.51549	-3.99749	5.01656
78.00	164.40	89.12	-4.13	-3.77	-11.75680	4.00422	-4.40908	5.14058
80.00	140.18	88.94	-3.78	-4.31	-12.45266	4.39634	-4.77282	5.25069
82.00	114.26	88.68	-3.37	-4.79	-13.26044	4.77107	-5.05888	5.34953
84.00	86.95	88.23	-2.92	-5.18	-13.88425	5.05448	-5.27177	5.42060
86.00	58.64	87.34	-2.43	-5.53	-14.26178	5.26513	-5.43035	5.46182
88.00	29.59	84.68	-1.92	-5.70	-14.59157	5.42772	-5.50205	5.48956

ORIGINAL PAGE IS
OF POOR QUALITY

90.00	2.75	360.00	=1.39	=5.76	=14.74888	5.49382	=5.50568	5.49823

THIS CONCLUDES THE CALCULATION.

ORIGINAL PAGE IS
OF POOR QUALITY

```
SENS-50 TABLES FOR NTKF CANON
N. LAUNCH ELEVATIONS (DEG) FOR IMPACT RANGE, R-0 FLT-EL AND APOGEE ALT
 11  70.00  72.00  74.00  76.00  78.00  80.00  82.00  84.00  86.00
  88.00  90.00
IMPACT RANGES IN NM
 128.19  120.00  110.18   99.10   86.79   73.42   59.17   44.21   28.73
  12.94    3.20
R-0 FLT/EL IN DEG
  63.30   65.96   68.63   71.31   74.00   76.70   79.40   82.11   84.82
  87.53   89.73
APOGEE ALTITUDE IN FT
  429521.   452973.   474109.   493497.   510755.   525576.   537708.
  546997.   553482.   557308.   558125.
N. ALTITUDE TABLE FOR F(Z) AND DELTA F(Z) CURVES
 12        0.        50.       100.       150.       200.       250.
     300.      1000.      3000.     10000.     30000.     60000.
F(Z) CURVE
  0.        0.1347    0.2535    0.3395    0.4082    0.4536    0.4961    0.7402
  0.8888    0.9160    0.8720    1.0000
DELTA F(Z) CURVE
  0.        0.1347    0.1187    0.0860    0.0687    0.0454    0.0425    0.2441
  0.1486    0.0272   -0.0440    0.1280
N. LAUNCH ELEVATIONS (DEG) FOR UNIT-WIND EFFECTS, RANGE DERIV AND CORIOLIS DEFL
 11  70.00  72.00  74.00  76.00  78.00  80.00  82.00  84.00  86.00
  88.00  90.00
HEAD UNIT-WIND EFFECT (NM/FT/SEC)
  0.309   0.404   0.494   0.579   0.659   0.724   0.785   0.832   0.867
  0.893   0.904
TAIL UNIT-WIND EFFECT (NM/FT/SEC)
 -0.407  -0.501  -0.585  -0.658  -0.726  -0.786  -0.833  -0.868  -0.894
 -0.906  -0.906
CROSS UNIT-WIND EFFECT (NM/FT/SEC)
  0.744   0.774   0.801   0.826   0.846   0.864   0.880   0.892   0.899
  0.903   0.905
RANGE DERIV (NM/DEG)
 -3.189  -4.213  -4.948  -5.450  -6.348  -6.724  -7.160  -7.497  -7.701
 -7.879  -7.964
CORIOLIS DEFLECTION TO NORTH (NM)
 -2.652  -2.604  -2.519  -2.393  -2.232  -2.041  -1.819  -1.576  -1.314
 -1.038  -0.752
CORIOLIS DEFLECTION TO EAST (NM)
 -0.577  -1.059  -1.347  -1.779  -2.033  -2.330  -2.588  -2.799  -2.985
 -3.076  -3.112
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
SENS-5D CALCULATIONS BEGIN ----------------------

INPUT DATA SET NO. 1

$DLIST  AZERO=20  $
$BLIST  NPL=1,PLM=50, AZGDL=120, NANG=1,ANG=80, IPRINT=1  $
$FLIST  $
$ULIST  $
```

NASA WALLOPS FLIGHT CENTER
WALLOPS ISLAND, VIRGINIA

TRAJECTORY SUMMARY AT BURNOUT, APOGEE AND IMPACT

VEHICLE = NIKE CAJUN
PAY LOAD = 50.00 LBS
LAUNCH AZ = 120.00 DEG
WIND = ZERO
EARTH = ROTATING MODEL

ORIGINAL PAGE IS
OF POOR QUALITY

DETAILED PRINT-OUT OF TRAJECTORY FOR LAUNCH ELEVATION = 80.00 DEG
UNITS ARE F-P-S-DEGREE, EXCEPT RANGE(NM) AND ACC(GO).

TIME	RANGE	BEARING	ALTITUDE	FLT/EL	FLT/AZ	VELOCITY	ACC	MACH	THRUST	DRAG	D/PRES	ETA	QALFA	WEIGHT	WINDN	WINDE
0.00	0.00	180.00	20.	79.66	119.11	0.	18.0	0.00	29740.	0.	0.	0.43	0.	1566.9	0.	0.
0.00	0.00	180.00	20.	79.64	119.11	1.	18.0	0.00	29807.	0.	0.	0.45	0.	1566.9	0.	0.
0.00	0.00	180.00	20.	79.63	119.10	1.	18.1	0.00	29875.	0.	0.	0.46	0.	1566.9	0.	0.
0.00	0.00	180.00	20.	79.62	119.10	1.	18.1	0.00	29943.	0.	0.	0.47	0.	1566.9	0.	0.
0.00	0.00	180.00	20.	79.61	119.10	1.	18.2	0.00	30010.	0.	0.	0.48	0.	1566.8	0.	0.
0.00	0.00	180.00	20.	79.60	119.10	1.	18.2	0.00	30078.	0.	0.	0.49	0.	1566.8	0.	0.
0.00	0.00	180.00	21.	79.60	119.10	1.	18.3	0.00	30145.	0.	0.	0.49	0.	1566.8	0.	0.
0.00	0.00	180.00	21.	79.59	119.10	1.	18.3	0.00	30213.	0.	0.	0.50	0.	1566.8	0.	0.
0.00	0.00	218.37	21.	79.59	119.10	1.	18.3	0.00	30281.	0.	0.	0.50	0.	1566.7	0.	0.
0.00	0.00	180.00	21.	79.58	119.10	2.	18.7	0.00	30889.	0.	0.	0.51	0.	1566.5	0.	0.
0.00	0.00	180.00	21.	79.58	119.10	2.	18.8	0.00	31024.	0.	0.	0.52	0.	1566.4	0.	0.
0.00	0.00	180.00	21.	79.58	119.10	2.	18.9	0.00	31159.	0.	0.	0.52	0.	1566.4	0.	0.
0.00	0.00	180.00	20.	79.58	119.10	3.	19.0	0.00	31295.	0.	0.	0.52	0.	1566.3	0.	0.
0.00	0.00	180.00	20.	79.57	119.10	3.	19.1	0.00	31430.	0.	0.	0.52	0.	1566.3	0.	0.
0.00	0.00	180.00	20.	79.57	119.10	3.	19.2	0.00	31565.	0.	0.	0.52	0.	1566.2	0.	0.
0.01	0.00	180.00	20.	79.57	119.10	3.	19.3	0.00	31700.	0.	0.	0.52	0.	1566.2	0.	0.
0.01	0.00	180.00	20.	79.58	119.10	3.	19.3	0.00	31835.	0.	0.	0.52	0.	1566.1	0.	0.
0.01	0.00	180.00	20.	79.58	119.10	4.	19.4	0.00	31970.	0.	0.	0.52	0.	1566.1	0.	0.
0.01	0.00	180.00	21.	79.58	119.10	5.	20.2	0.00	33187.	0.	0.	0.51	0.	1565.6	0.	0.
0.01	0.00	180.00	21.	79.58	119.10	6.	20.4	0.01	33458.	0.	0.	0.51	0.	1565.5	0.	0.
0.01	0.00	180.00	21.	79.58	119.10	6.	20.6	0.01	33728.	0.	0.	0.51	0.	1565.4	0.	0.
0.01	0.00	180.00	21.	79.59	119.10	7.	20.7	0.01	33998.	0.	0.	0.51	0.	1565.3	0.	0.
0.01	0.00	218.37	21.	79.59	119.10	7.	20.9	0.01	34269.	0.	0.	0.50	0.	1565.1	0.	0.
0.01	0.00	207.82	21.	79.59	119.10	7.	21.1	0.01	34539.	0.	0.	0.50	0.	1565.0	0.	0.
0.01	0.00	207.82	21.	79.59	119.10	8.	21.3	0.01	34809.	0.	0.	0.50	0.	1564.9	0.	0.
0.01	0.00	207.82	21.	79.59	119.10	8.	21.4	0.01	35080.	0.	0.	0.50	0.	1564.8	0.	0.
0.01	0.00	201.59	21.	79.60	119.10	9.	21.6	0.01	35350.	0.	0.	0.50	0.	1564.7	0.	0.
0.02	0.00	201.59	21.	79.61	119.10	13.	23.2	0.01	37784.	0.	0.	0.48	0.	1563.8	0.	0.
0.02	0.00	201.59	21.	79.62	119.10	14.	23.5	0.01	38324.	0.	0.	0.48	0.	1563.5	0.	0.
0.02	0.00	201.59	21.	79.62	119.10	15.	23.9	0.01	38865.	0.	0.	0.47	0.	1563.3	0.	0.
0.02	0.00	201.59	21.	79.62	119.10	16.	24.2	0.01	39406.	0.	0.	0.47	0.	1563.1	0.	0.
0.02	0.00	201.59	21.	79.63	119.10	17.	24.4	0.01	39689.	0.	0.	0.46	0.	1562.8	0.	0.
0.03	0.00	201.59	21.	79.63	119.10	18.	24.6	0.02	39965.	0.	0.	0.46	0.	1562.6	0.	0.
0.03	0.00	201.59	21.	79.63	119.10	19.	24.8	0.02	40242.	0.	0.	0.46	0.	1562.3	0.	0.
0.03	0.00	201.59	21.	79.64	119.10	20.	25.0	0.02	40518.	0.	0.	0.45	0.	1562.0	0.	0.
0.03	0.00	201.59	21.	79.64	119.10	21.	25.1	0.02	40794.	0.	1.	0.45	0.	1561.8	0.	0.
0.04	0.00	180.00	21.	79.66	119.10	32.	25.8	0.03	41822.	1.	1.	0.43	1.	1558.8	0.	0.

TIME	RANGE	BEARING	ALTITUDE	FLT/EL	FLT/AZ	VELOCITY	ACC	MACH	THRUST	DRAG	D/PRES	ETA	QALFA	WEIGHT	WINDN	WINDE
0.04	0.00	180.00	21.	79.67	119.10	33.	25.9	0.03	41846.	1.	1.	0.43	1.	1558.5	0.	0.
0.05	0.00	180.00	21.	79.67	119.10	34.	25.9	0.03	41870.	1.	1.	0.43	1.	1558.2	0.	0.
0.05	0.00	180.00	21.	79.67	119.10	35.	25.9	0.03	41894.	1.	1.	0.42	1.	1558.0	0.	0.
0.05	0.00	180.00	21.	79.67	119.10	36.	25.9	0.03	41918.	2.	2.	0.42	1.	1557.7	0.	0.
0.05	0.00	180.00	21.	79.67	119.10	37.	25.9	0.03	41942.	2.	2.	0.42	1.	1557.4	0.	0.
0.05	0.00	180.00	21.	79.67	119.10	38.	26.0	0.03	41966.	2.	2.	0.42	1.	1557.1	0.	0.
0.05	0.00	180.00	21.	79.67	119.10	39.	26.0	0.04	41990.	2.	2.	0.42	1.	1556.9	0.	0.
0.05	0.00	180.00	21.	79.67	119.10	40.	26.0	0.04	42014.	2.	2.	0.42	1.	1556.6	0.	0.
0.06	0.00	180.00	22.	79.68	119.10	50.	26.2	0.04	42231.	3.	3.	0.41	1.	1554.1	0.	0.
0.07	0.00	180.00	22.	79.68	119.10	52.	26.2	0.05	42279.	3.	3.	0.41	1.	1553.6	0.	0.
0.07	0.00	180.00	22.	79.68	119.10	54.	26.3	0.05	42327.	3.	3.	0.41	1.	1553.0	0.	0.
0.07	0.00	180.00	22.	79.69	119.10	56.	26.3	0.05	42375.	4.	4.	0.41	2.	1552.5	0.	0.
0.07	0.00	165.22	22.	79.69	119.10	58.	26.3	0.05	42403.	4.	4.	0.41	2.	1551.9	0.	0.
0.08	0.00	165.22	23.	79.69	119.10	60.	26.4	0.05	42411.	4.	4.	0.40	2.	1551.4	0.	0.
0.08	0.00	165.22	23.	79.69	119.10	62.	26.4	0.06	42419.	5.	5.	0.40	2.	1550.8	0.	0.
0.08	0.00	165.22	23.	79.69	119.10	65.	26.4	0.06	42428.	5.	5.	0.40	2.	1550.3	0.	0.
0.08	0.00	162.43	23.	79.69	119.10	67.	26.4	0.06	42436.	5.	5.	0.40	2.	1549.7	0.	0.
0.11	0.00	147.65	25.	79.70	119.10	86.	26.5	0.08	42508.	9.	9.	0.40	3.	1544.7	0.	0.
0.11	0.00	147.65	25.	79.70	119.10	90.	26.6	0.08	42524.	9.	10.	0.39	4.	1543.6	0.	0.
0.12	0.00	152.18	26.	79.70	119.10	94.	26.6	0.08	42540.	10.	11.	0.39	4.	1542.5	0.	0.
0.12	0.00	141.63	26.	79.70	119.10	99.	26.6	0.09	42556.	11.	12.	0.39	5.	1541.4	0.	0.
0.13	0.00	141.63	27.	79.70	119.10	103.	26.6	0.09	42572.	12.	13.	0.39	5.	1540.3	0.	0.
0.13	0.00	145.84	27.	79.70	119.10	107.	26.7	0.10	42588.	13.	14.	0.39	5.	1539.2	0.	0.
0.14	0.00	137.86	28.	79.70	119.10	112.	26.7	0.10	42596.	14.	15.	0.39	6.	1538.0	0.	0.
0.14	0.00	137.86	28.	79.70	119.10	116.	26.7	0.10	42601.	16.	16.	0.39	6.	1536.9	0.	0.
0.15	0.00	141.63	29.	79.70	119.10	120.	26.7	0.11	42606.	17.	17.	0.39	7.	1535.8	0.	0.
0.19	0.00	130.97	35.	79.71	119.10	159.	27.0	0.14	42650.	29.	30.	0.38	11.	1525.7	0.	0.
0.20	0.00	132.37	37.	79.71	119.10	168.	27.0	0.15	42660.	32.	33.	0.38	13.	1523.5	0.	0.
0.21	0.00	126.74	38.	79.71	119.10	176.	27.0	0.16	42670.	36.	37.	0.38	14.	1521.2	0.	0.
0.22	0.00	128.29	40.	79.71	119.10	185.	27.1	0.17	42680.	39.	41.	0.38	15.	1519.0	0.	0.
0.23	0.00	126.08	42.	79.71	119.10	194.	27.1	0.17	42690.	43.	45.	0.38	17.	1516.7	0.	0.
0.24	0.00	125.60	44.	79.72	119.10	203.	27.2	0.18	42700.	47.	49.	0.37	18.	1514.5	0.	0.
0.25	0.00	123.86	46.	79.72	119.10	211.	27.2	0.19	42710.	51.	53.	0.37	20.	1512.2	0.	0.
0.26	0.00	125.22	48.	79.72	119.10	220.	27.3	0.20	42720.	55.	57.	0.37	21.	1510.0	0.	0.
0.27	0.00	123.62	50.	79.72	119.10	229.	27.3	0.21	42730.	60.	62.	0.37	23.	1507.7	0.	0.
0.36	0.00	121.51	74.	79.72	119.10	309.	27.7	0.28	42821.	107.	113.	0.35	40.	1487.6	0.	0.
0.38	0.00	120.89	80.	79.73	119.10	326.	27.8	0.29	42841.	120.	126.	0.35	44.	1483.1	0.	0.
0.40	0.00	119.81	87.	79.73	119.10	344.	27.9	0.31	42861.	133.	141.	0.34	48.	1478.6	0.	0.

ORIGINAL PAGE
OF POOR QUALITY

TIME	RANGE	BEARING	ALTITUDE	FLT/EL	FLT/AZ	VELOCITY	ACC	MACH	THRUST	DRAG	D/PRES	ETA	QALFA	WEIGHT	WINDN	WINDE
0.42	0.00	120.02	94.	79.73	119.10	362.	28.0	0.32	42882.	147.	156.	0.34	52.	1474.1	0.	0.
0.44	0.00	120.24	101.	79.73	119.10	380.	28.1	0.34	42902.	161.	171.	0.33	57.	1469.6	0.	0.
0.46	0.00	119.95	109.	79.73	119.10	398.	28.2	0.36	42923.	177.	188.	0.32	61.	1465.1	0.	0.
0.48	0.00	120.14	117.	79.73	119.10	417.	28.3	0.37	42943.	193.	206.	0.32	65.	1460.6	0.	0.
0.50	0.00	119.90	125.	79.73	119.10	435.	28.4	0.39	42964.	209.	224.	0.31	69.	1456.1	0.	0.
0.52	0.00	119.73	134.	79.73	119.10	453.	28.5	0.41	42984.	226.	243.	0.30	72.	1451.7	0.	0.
0.70	0.01	119.49	229.	79.71	119.11	621.	29.3	0.56	43171.	412.	455.	0.18	83.	1411.3	0.	0.
0.74	0.01	119.43	254.	79.71	119.11	658.	29.5	0.59	43213.	461.	511.	0.15	76.	1402.3	0.	0.
0.78	0.01	119.40	281.	79.70	119.11	696.	29.7	0.62	43255.	513.	572.	0.11	65.	1393.3	0.	0.
0.82	0.01	119.39	309.	79.69	119.11	735.	29.9	0.66	43297.	567.	636.	0.08	51.	1384.4	0.	0.
0.86	0.01	119.54	339.	79.67	119.11	773.	30.1	0.69	43339.	623.	704.	0.05	32.	1375.3	0.	0.
0.94	0.01	119.24	403.	79.64	119.11	851.	30.4	0.76	43423.	769.	851.	0.02	14.	1357.1	0.	0.
0.96	0.01	119.37	420.	79.63	119.12	871.	30.5	0.78	43444.	835.	890.	0.03	26.	1352.6	0.	0.
0.98	0.01	119.39	437.	79.63	119.12	891.	30.6	0.80	43465.	905.	931.	0.04	37.	1348.1	0.	0.
1.00	0.01	119.41	455.	79.62	119.12	910.	30.7	0.82	43486.	978.	972.	0.05	49.	1343.5	0.	0.
2.17	0.07	119.18	2202.	79.33	119.16	2169.	37.3	1.96	46483.	5540.	5239.	0.00	12.	1070.8	0.	0.
3.05	0.14	119.17	4542.	79.23	119.18	3209.	28.4	2.92	35056.	9504.	10693.	0.00	12.	869.7	0.	0.
3.52	0.19	119.14	6076.	79.18	119.19	3308.	11.6	3.03	638.	9468.	10844.	0.00	8.	831.0	0.	0.
4.07	0.24	119.14	7832.	79.13	119.21	3198.	5.9	2.95	0.	1319.	9606.	0.00	7.	267.0	0.	0.
5.03	0.34	119.16	10765.	79.03	119.23	3028.	5.1	2.82	0.	1109.	7856.	0.00	6.	267.0	0.	0.
6.05	0.43	119.18	13714.	78.92	119.26	2871.	4.5	2.70	0.	930.	6428.	0.00	5.	267.0	0.	0.
7.07	0.52	119.19	16519.	78.79	119.29	2734.	3.9	2.60	0.	785.	5319.	0.00	5.	267.0	0.	0.
8.03	0.60	119.21	19038.	78.67	119.32	2619.	3.5	2.52	0.	674.	4490.	0.00	5.	267.0	0.	0.
9.03	0.69	119.22	21753.	78.54	119.34	2513.	3.2	2.44	0.	579.	3794.	0.00	4.	267.0	0.	0.
10.03	0.77	119.23	23966.	78.40	119.37	2416.	2.9	2.37	0.	499.	3228.	0.00	4.	267.0	0.	0.
11.03	0.85	119.24	26289.	78.25	119.40	2329.	2.6	2.30	0.	432.	2762.	0.00	4.	267.0	0.	0.
12.27	0.94	119.26	29055.	78.05	119.43	2231.	2.3	2.23	0.	364.	2294.	0.00	3.	267.0	0.	0.
13.15	1.01	119.27	30947.	77.90	119.46	2167.	2.2	2.19	0.	323.	2019.	0.00	3.	267.0	0.	0.
14.03	1.07	119.29	32985.	77.75	119.48	2107.	2.1	2.14	0.	287.	1782.	0.00	3.	267.0	0.	0.
15.03	1.15	119.30	34812.	77.57	119.51	2043.	1.9	2.10	0.	252.	1552.	0.00	2.	267.0	0.	0.
16.03	1.22	119.31	36377.	77.38	119.53	1983.	1.8	2.05	0.	221.	1348.	0.00	2.	267.0	0.	0.
17.00	1.29	119.32	38021.	77.19	119.56	1929.	1.7	1.99	0.	194.	1167.	0.00	2.	267.0	0.	0.
18.01	1.37	119.34	40922.	77.01	119.58	2814.	32.5	2.91	8105.	275.	2224.	0.00	2.	234.1	0.	0.
19.09	1.51	119.36	44749.	76.89	119.61	4143.	43.5	4.28	8951.	418.	4054.	0.00	3.	191.9	0.	0.
20.02	1.68	119.39	48953.	76.82	119.63	5654.	48.0	5.84	8031.	544.	6116.	0.00	3.	152.9	0.	0.
21.11	1.92	119.42	55115.	76.75	119.66	5822.	0.8	6.01	458.	422.	4829.	0.00	2.	148.1	0.	0.
22.00	2.11	119.44	60365.	76.70	119.68	5808.	0.3	6.00	438.	330.	3776.	0.00	1.	148.0	0.	0.
23.52	2.44	119.47	68043.	76.60	119.71	5664.	2.5	5.84	0.	233.	2386.	0.	0.	148.0	0.	0.

TIME	RANGE	BEARING	ALTITUDE	FLT/EL	FLT/AZ	VELOCITY	ACC	MACH	THRUST	DRAG	D/PRES	ETA	QALFA	WEIGHT	WINDN	WINDE
24.56	2.67	119.49	74332.	76.53	119.74	5586.	2.1	5.74	0.	173.	1759.	0.	0.	148.0	0.	0.
25.04	2.77	119.50	76932.	76.50	119.75	5554.	2.0	5.69	0.	151.	1533.	0.	0.	148.0	0.	0.
26.64	3.11	119.52	85497.	76.39	119.79	5462.	1.6	5.57	0.	98.	982.	0.	0.	148.0	0.	0.
27.28	3.24	119.53	88884.	76.34	119.80	5430.	1.5	5.52	0.	83.	826.	0.	0.	148.0	0.	0.
30.80	3.98	119.58	107179.	76.09	119.88	5279.	1.2	5.29	0.	33.	330.	0.	0.	148.0	0.	0.
32.08	4.24	119.60	113907.	75.99	119.91	5232.	1.1	5.18	0.	24.	237.	0.	0.	148.0	0.	0.
33.04	4.44	119.61	118564.	75.92	119.93	5198.	1.1	5.10	0.	19.	186.	0.	0.	148.0	0.	0.
35.92	5.04	119.65	132940.	75.69	120.00	5100.	1.0	4.89	0.	10.	93.	0.	0.	148.0	0.	0.
36.56	5.17	119.66	136896.	75.64	120.01	5080.	1.0	4.84	0.	9.	81.	0.	0.	148.0	0.	0.
37.20	5.30	119.66	139238.	75.59	120.03	5059.	1.0	4.80	0.	7.	70.	0.	0.	148.0	0.	0.
38.48	5.56	119.68	145483.	75.49	120.06	5018.	1.0	4.71	0.	6.	53.	0.	0.	148.0	0.	0.
45.52	7.01	119.76	178880.	74.88	120.21	4798.	1.0	4.47	0.	1.	13.	0.	0.	148.0	0.	0.
46.16	7.14	119.76	184838.	74.83	120.23	4778.	1.0	4.46	0.	1.	12.	0.	0.	148.0	0.	0.
47.44	7.40	119.78	187714.	74.71	120.25	4738.	1.0	4.46	0.	1.	9.	0.	0.	148.0	0.	0.
48.08	7.53	119.79	190833.	74.65	120.27	4719.	1.0	4.45	0.	1.	8.	0.	0.	148.0	0.	0.
53.84	8.70	119.85	216322.	74.09	120.39	4543.	1.0	4.50	0.	0.	3.	0.	0.	148.0	0.	0.
55.12	8.96	119.86	221889.	73.97	120.42	4504.	1.0	4.53	0.	0.	2.	0.	0.	148.0	0.	0.
56.40	9.22	119.87	227404.	73.83	120.44	4466.	1.0	4.55	0.	0.	2.	0.	0.	148.0	0.	0.
57.68	9.48	119.89	232869.	73.70	120.47	4427.	1.0	4.58	0.	0.	1.	0.	0.	148.0	0.	0.
58.96	9.73	119.90	238282.	73.56	120.50	4388.	1.0	4.61	0.	0.	1.	0.	0.	148.0	0.	0.
70.48	12.05	120.01	284099.	72.22	120.73	4043.	1.0	4.57	0.	0.	0.	0.	0.	148.0	0.	0.
73.04	12.57	120.04	294652.	71.89	120.79	3967.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
75.60	13.08	120.06	304802.	71.54	120.84	3891.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
78.16	13.59	120.09	313549.	71.19	120.89	3815.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
80.72	14.10	120.11	322492.	70.81	120.94	3740.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
96.08	17.14	120.25	378892.	68.22	121.22	3292.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
98.64	17.65	120.28	388817.	67.72	121.27	3218.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
101.20	18.15	120.30	388540.	67.19	121.31	3145.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
103.76	18.66	120.32	395661.	66.64	121.36	3072.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
106.32	19.16	120.34	402780.	66.06	121.40	3000.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
129.36	23.67	120.54	457797.	59.28	121.78	2367.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
134.48	24.66	120.58	467815.	57.26	121.85	2233.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
139.60	25.66	120.62	473831.	54.98	121.93	2102.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
144.72	26.65	120.66	485446.	52.40	122.00	1976.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
149.84	27.64	120.70	493061.	49.48	122.07	1854.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
200.05	37.33	121.04	525630.	-5.66	122.64	1207.	0.9	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
209.05	39.07	121.09	528328.	-18.10	122.71	1264.	0.9	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
227.05	42.54	121.19	511132.	-38.00	122.85	1528.	0.9	4.49	0.	0.	0.	0.	0.	148.0	0.	0.

ORIGINAL PAGE
OF POOR QUALITY

TIME	RANGE	BEARING	ALTITUDE	FLT/EL	FLT/AZ	VELOCITY	ACC	MACH	THRUST	DRAG	D/PRES	ETA	QALFA	WEIGHT	WINDN	WINDE
263.05	49.53	121.37	45[illegible]18.	-59.25	123.02	2373.	1.0	4.49	0.	0.	0.	0.	0.	148.0	0.	0.
365.05	70.03	121.67	8[illegible]218.	-76.26	122.89	5289.	0.5	5.38	0.	78.	771.	0.	0.	148.0	0.	0.
367.05	70.44	121.67	78919.	-76.40	122.88	5307.	0.2	5.43	0.	128.	1272.	0.	0.	148.0	0.	0.
368.05	70.64	121.67	78759.	-76.48	122.87	5307.	0.3	5.45	0.	164.	1632.	0.	0.	148.0	0.	0.
369.05	70.84	121.67	68[illegible]02.	-76.55	122.87	5297.	0.5	5.46	0.	210.	2091.	0.	0.	148.0	0.	0.
370.55	71.15	121.67	60[illegible]95.	-76.65	122.86	5262.	1.1	5.44	0.	301.	2992.	0.	0.	148.0	0.	0.
371.05	71.25	121.67	58[illegible]40.	-76.69	122.86	5242.	1.3	5.42	0.	338.	3356.	0.	0.	148.0	0.	0.
372.05	71.44	121.68	58261.	-76.76	122.85	5191.	1.9	5.36	0.	424.	4195.	0.	0.	148.0	0.	0.
373.05	71.64	121.68	48240.	-76.83	122.84	5119.	2.6	5.29	0.	527.	5187.	0.	0.	148.0	0.	0.
374.05	71.82	121.68	48299.	-76.91	122.84	5023.	3.4	5.19	0.	647.	6327.	0.	0.	148.0	0.	0.
375.05	72.01	121.68	38464.	-76.98	122.83	4899.	4.3	5.06	0.	782.	7587.	0.	0.	148.0	0.	0.
376.05	72.19	121.68	38763.	-77.06	122.82	4745.	5.2	4.85	0.	920.	8713.	0.	0.	148.0	0.	0.
377.05	72.36	121.68	29224.	-77.14	122.82	4563.	6.1	4.57	0.	1044.	9539.	0.	0.	148.0	0.	0.
378.05	72.52	121.68	24874.	-77.22	122.81	4355.	6.8	4.28	0.	1152.	10157.	0.	0.	148.0	0.	0.
379.05	72.67	121.68	20736.	-77.31	122.80	4126.	7.4	3.99	0.	1236.	10523.	0.	0.	148.0	0.	0.
380.05	72.82	121.68	16828.	-77.40	122.80	3881.	7.8	3.70	0.	1304.	10612.	0.	0.	148.0	0.	0.
381.05	72.95	121.68	13164.	-77.50	122.79	3629.	8.1	3.40	0.	1338.	10430.	0.	0.	148.0	0.	0.
382.05	73.08	121.68	9752.	-77.60	122.78	3365.	8.1	3.12	0.	1337.	10015.	0.	0.	148.0	0.	0.
383.01	73.19	121.68	6686.	-77.71	122.78	3115.	8.0	2.86	0.	1322.	9442.	0.	0.	148.0	0.	0.
384.01	73.29	121.68	8765.	-77.83	122.77	2863.	7.7	2.60	0.	1287.	8711.	0.	0.	148.0	0.	0.
385.01	73.39	121.69	1885.	-77.97	122.77	2621.	7.3	2.36	0.	1225.	7906.	0.	0.	148.0	0.	0.

	APOGEE			BURNOUT						IMPACT		
EL (DEG)	TIME (SEC)	ALT (FT)	RANGE (NM)	TIME (SEC)	ALT (FT)	RANGE (NM)	VEL (FT/SEC)	FLT/EL (DEG)	FLT/AZ (DEG)	TIME (SEC)	RANGE (NM)	AZ (DEG)
80.00	196.05	525976.	36.6	22.00	60165.	2.1	5808.36	76.70	119.68	385.44	73.4	121.69

THIS CONCLUDES THE CALCULATION.

ORIGINAL PAGE IS OF POOR QUALITY

APPENDIX D

CONTROL CARDS

Deck set up with object program on tape.

```
1        8            16
$        IDENT        153202, SENS-5D
$        EXECUTE      DUMP
$        LIMITS       100, 30K,,5K
$        TAPE         R*, X1D,,
$        FILE         02, X2R,  5L
$        FILE         03, X3R,  5L
$        DATA         05
$        INCODE       IBMF
```

Input data with NAMELIST Format:

```
$        DATA         01
$        INCODE       IBMF
```

Input data with FORMAT(V) Format

```
$        ENDJOB
```

***EOF

Deck set-up with object program on cards.

```
1       8           16
$       IDENT       153202, SENS-5D
$       OPTION      FORTRAN
```

Object Deck

```
$       EXECUTE     DUMP
$       LIMITS      100, 30K,, 5K
$       FILE        02, X2R,  5L
$       FILE        03, X3R,  5L
$       DATA        05
$       INCODE      IBMF
```

Input Data with NAMELIST Format

```
$       DATA        01
$       INCODE      IBMF
```

Input Data with FORMAT(V) Format

```
$       ENDJOB

***EOF
```

APPENDIX E

SOURCE PROGRAM

```
CMAIN                                                                  MAIN
C**** SENS-5D ***********************************************************
C     AUTHORS -- B. P. SINGH AND R. COOK.
C     DATE -- JUNE 23, 1975.
C
C     * THIS PROGRAM CALCULATES THE FOLLOWING --
C       (1) SUMMARY AT BURN-OUT, APOGEE, AND IMPACT IN 'BAI',
C       (2) F(Z) AND DF(Z) CURVES IN 'FZCURV',
C       (3) UNIT-WIND EFFECT,CORIOLIS DEFLN AND RANGE DERIV IN 'EFFECT',
C
C     * THE FOLLOWING UNITS ARE ADOPTED --
C       (1) INPUT/OUTPUT -- FOOT-POUND-SECOND-DEGREE,
C           EXCEPTION -- SURFACE RANGE IN NAUTICAL MILES (NM),
C       (2) CALCULATIONAL STEPS -- FOOT-POUND-SECOND-RADIAN,
C
C     * WIND PROFILE ----
C       (1) NORTH-TO-SOUTH OR EAST-TO-WEST DIRECTION IS POSITIVE.
C       (2) NO WIND MEANS NO WIND FIELD AT ALL.
C       (3) CONSTANT WIND MEANS ONE WIND FIELD FROM GROUND TO SOME
C           ALTITUDE. PARAMETERS USED ARE (WIND,WNAZ,ALOW,AHIGH),
C           DENOTING WIND SPEED, NORTH AZIMUTH, MINIMUM ALTITUDE,
C           AND MAXIMUM ALTITUDE RESPECTIVELY.
C       (4) VARIABLE WIND MEANS WIND VELOCITY VARIES WITH ALTITUDE,
C           PARAMETERS USED ARE (NLEV,ALTW,SPEED,DIR), DENOTING
C           NUMBER OF ALTITUDES, ALTITUDES, SPEED, AND NORTH
C           AZIMUTH MEASURED CLOCKWISE.
C
C     * THE FOLLOWING ARE THE LIMITS FOR ENTRIES --
C       (1) 20 PAY PAYLOADS,
C       (2) 61 ELEVATION ANGLES,
C       (3) 30 THRUSTING AND COASTING PHASES,
C       (4) 20 SPENT STAGES,
C       (5) 99 ENTRIES IN THRUST AND MACH TABLES,
C       (6) 99 WIND ALTITUDES.
C
C     * THE FOLLOWING TAPES OR FILES ARE DESIGNATED --
C       (1) ITAPE1 (ALWAYS 5) IS USED FOR READING DATA IN NAMELISTS,
C       (2) ITAPE2 IS USED FOR READING VEHICLE THRUST-MACH TABLES,
C       (3) ITAPE3 IS USED (INTERNAL) WRITING AND READING THRUST TABLES,
C       (4) ITAPE4 IS USED (INTERNAL) FOR SPENT STAGES,
C
C**** ***********************************************************************
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
C****
      INTEGER RKSTEP
      INTEGER ORDER
      REAL LATL,LONGL,LAT,LONG
      REAL MASS,MPROP,MPL,MDOT,MACH,LGG,LGP,IYY
      REAL IPT,IPAZ,IPR
      REAL MSEP
      DOUBLE PRECISION DRS,DPC
      CHARACTER NAME*80,LEFT*61,RIGHT*19
C
      DIMENSION SPEED(100),DIR(100)
C
      COMMON/STOR00/ ZERO,ONE,PI,QPI,HPI,TPI,CP(9)
      COMMON/STOR01/ G0,GR,OMEGA,GM,RE,RO,ESQ,ESQ1,ESQ1
      COMMON/STOR02/ PRESO,DENSO,TEMPO,SOUNDO,PRESP(9),TEMPB(9),
     ,               HALTB(9),TSLOPE(9),CONST,PRES,DENS,TEMP,SOUND
      COMMON/STOR03/ XO,YO,ZO,R,RXY,ALT,SR,HLIM
      COMMON/STOR04/ GDLATL,LATL,LONGL,THETAL,PHIL,GDLAT,LAT,LONG,THETA,
     ,               PHI,GAMAL,PSIL,GAMA,PSI,ELGDL,AZGDL,ELGD,AZGD,EL,
     ,               AZ,SRAZ
      COMMON/STOR05/ VT(3),VA(3),VEL,ACC,VQLIM,VLIM
      COMMON/STOR06/ RTI(9)
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTAI,TOO,RKSTEP,EPBIG,EPTINY,
     ,               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR08/ S(14),F(14,5),DRS(14),DPC(14),ERW(14),SP(6,20)
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,CA,AD,QS,
     ,               SLORE,CMP,LGG,LGP,IYY,SY,SZ
      COMMON/STOR10/ WE,WN,WLAST,WIND,WMAZ,ALOW,AHIGH,NLEV,ALEV(100),
     ,               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR11/ KPREV,HPREV,MPREV,APREV,LPREV(14),VPREV(14),
     ,               ORDER(14)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSC,NCON
      COMMON/STOR13/ APT,APAZ,APR,AER,BOT,BOA,BOR,BOVL,BOEL,BOAZ,IRT,
     ,               IPAZ,IPR
      COMMON/STOR14/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPUNCH,INDEX
      COMMON/STOR99/ NAME,ARRAY(800)
C
      COMMON NPST,NSEP,NSP,NM(14),KSP(20),RMASS,T3D,TAY(3),ANG(61),B(4),
     ,       PLM(20),PHT(32),TSER(32),MSEP(32),TSP(20),SPM(20),A(99,14)
C
      EQUIVALENCE (ARRAY(1),SPEED(1)),(ARRAY(101),DIR(1))
C
```

```
C
C
C**** VALUES OF CONSTANTS.
      DATA ZERO,ONE/ 1.0E-15, 1.0E+00/
      DATA PI,QPI,HPI,TPI/ 3.14159265, 7.85398164E-01, 1.57079633,
     ,     6.28318531/
      DATA ESQ,ESQ1,ESQI/ 6.69342100E-03, 9.93306579E-01, 1.00673853/
      DATA GO,OMEGA,GM,RE,RQ/ 32.174, 7.29211E-05, 1.4076576E+16,
     ,     2.08905780E+07, 2.09257410E+07/
      DATA CF/ 57.29577951, 1.74532925E-02, 6.07611550E+03,
     ,         1.64578833E-04, 3.10809970E-02, 0.3048, 3.2083990,
     ,         1.852, 5.39956804E-01/
      DATA NPMAX,NSMAX,NVAR/ 30, 20, 99/
      DATA DUMMY/ 99999.0/
C
C**** U. S. 1962 STANDARD ATMOSPHERIC TABLE.
      DATA CONST/34.163194E-3/
      DATA PRESO/2116.2/,DENSO/2.3769E-3/,TEMPO/288.15/,SOUNDO/1116.45/
      DATA PRESB/ 1.0,2.23361E-1,5.40328E-2,8.56663E-3,1.09455E-3,
     ,     5.82289E-4,1.79718E-4,1.0241E-5,1.6223E-6/
      DATA TEMPB/ 288.15,216.65,216.65,228.65,270.65,270.65,252.65,
     ,     180.65,180.65/
      DATA HALTB/ 0.0,11.0E+3,20.0E+3,32.0E+3,47.0E+3,52.0E+3,61.0E+3,
     ,     79.0E+3,88.743E+3/
      DATA TSLOPE/ -6.5E-3,0.0,1.0E-3,2.8E-3,0.0,-2.0E-3,-4.0E-3,0.,0./
C
C**** DEFAULT VALUES OF SOME INPUT DATA.
      DATA GDLATL/37.8480/, LONGL/-75.4736/
      DATA TZERO/0./, AZERO/0./, VZERO/0./
      DATA ITAPE1,ITAPE2,ITAPE3,ITAPE4/ 5, 1, 2, 3/
      DATA DTLA/0.01/, DTAI/1.00/, DMIN/0.0002/, DMAX/20.0/
      DATA EPTINY/0.0001/, EPBIG/0.001/
      DATA VOLIM/0.1/, VLIM/40.0/, HLIM/20.0/
      DATA ERW/ 1.1,1.0/
      DATA IPUNCH / 0/
      DATA ORDER/ 14*1/
      DATA JSPENT/0/
      DATA WIND/20.0/, WNAZ/0./, ALOW/0./, AHIGH/100000.0/
      DATA NLEV/12/, ALEV/0.,50.,100.,200.,500.,1000.,2000.,3000.,5000.,
     ,     10000.,50000.,89.100000./
C
C
```

```
C
C**** NAMELISTS
      NAMELIST/DLIST/ GDLATL,LONGL,TZERO,AZERO,VZERO,ITAPE1,ITAPE2,
     ,                ITAPE3,ITAPE4,DTLA,DTAI,DMIN,DMAX,EPTINY,EPBIG,
     ,                ERW,VOLIM,VLIM,WLIM,ORDER,IPUNCH
      NAMELIST/BLIST/ WPL,PLM,AZGDL,NANG,ANG,WIND,WNAZ,ALOW,AHIGH,
     ,                NLEV,ALTW,SPEED,DIR,IROT,IPRINT,JSPENT
      NAMELIST/FLIST/ WPL,AZGDL,ELGDL,WIND,WNAZ,NLEV,ALEV,IROT,IPRINT
      NAMELIST/ULIST/ WPL,AZGDL,NANG,ANG,WIND,ALOW,AHIGH,IROT,IPRINT
C
C
C**** INITIALIZE THE PROGRAM.
      PRINT 1000
      SAVEW = WIND
      SAVEA = WNAZ
      LEVEL = NLEV
      SAVEL = ALOW
      SAVEH = AHIGH
      READ(ITAPE1,DLIST)
      I = 0
      READ(ITAPE1,BLIST)
      IF(IPRINT.EQ.2) I=2
      READ(ITAPE1,FLIST)
      IF(IPRINT.EQ.2) I=2
      READ(ITAPE1,ULIST)
      IF(IPRINT.EQ.2) I=2
      IPRINT = I
      PRINT 1001
      CALL LIST(ITAPE1)
      PRINT 1002
      CALL LIST(ITAPE2)
      GDLATL = GDLATL*BF(2)
      LONGL = LONGL*BF(2)
      TAV(1) = TZERO
      TAV(2) = AZERO
      TAV(3) = VZERO
      CALL RDATA
      NCON = -1
      CALL SETIV
      DECODE (NAME,1003) LEFT,RIGHT
      IF(IPUNCH.NE.0) PUNCH 1050, LEFT
C
```

```
C**** SUMMARY AT BURN-OUT, APOGEE, AND IMPACT.
      IROT = 1
      NLEV = 0
      MLAST = 0
      WIND = 0.0
      WNAZ = SAVEA
      ALOW = SAVEL
      AHIGH = SAVEH
      IPRINT = 0
      AZGDL = DUMMY
      READ(ITAPE1,BLIST)
      IF(ABS(AZGDL-DUMMY) .LT. 1.0E-05) GO TO 200
      IF(WIND) 110,120,110
  110 MLAST = 3
      WNAZ = WNAZ*CF(2)
      CALL WCONST
      GO TO 140
  120 IF(NLEV) 200,140,130
  130 MLAST = NLEV
      DO 131 I=1,MLAST
      WAZ = DIR(I)*CF(2)
      WAZ = AMOD(WAZ+PI,TPI)
      VELE(I) = SPEED(I)*SIN(WAZ)
      VELN(I) = SPEED(I)*COS(WAZ)
  131 CONTINUE
  140 DO 150 J=1,NPL
      PLM(J) = PLM(J)*CF(5)
  150 CONTINUE
      AZGDL = AZGDL*CF(2)
      DO 160 J=1,NANG
      ANG(J) = ANG(J)*CF(2)
  160 CONTINUE
      CALL BAI(IROT)
      IEND = NSP
      IF(JSPENT .EQ. 0) GO TO 200
      IF(IPRINT.EQ.2) IPRINT=1
      CALL SPENT(IROT)
      IEND = 0
C
C**** F(Z) AND DF(Z) CURVES.
  200 CONTINUE
      IROT = 0
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      NLEV = LEVEL
      WIND = SAVEW
      WNAZ = SAVEA
      IPRINT = 0
      AZGDL = DUMMY
      READ(ITAPE1,FLIST)
      IF(ABS(AZGDL-DUMMY) .LT. 1.0E-05) GO TO 300
      MPL = WPL*CF(5)
      AZGDL = AZGDL*CF(2)
      ELGDL = ELGDL*CF(2)
      WNAZ = WNAZ*CF(2)
      MLAST = 3
      CALL FZCURV(IROT)
      IEND = NSP
C
C**** TOWER TILT, UNIT-WIND EFFECTS, AND CORIOLIS DEFLECTION
  300 CONTINUE
      IROT = 0
      WIND = SAVEW
      ALOW = SAVEL
      AHIGH = SAVEH
      IPRINT = 0
      AZGDL = DUMMY
      READ(ITAPE1,ULIST)
      IF(ABS(AZGDL-DUMMY) .LT. 1.0E-05) GO TO 400
      MPL = WPL*CF(5)
      AZGDL = AZGDL*CF(2)
      DO 310 J=1,NANG
      ANG(J) = ANG(J)*CF(2)
  310 CONTINUE
      MLAST = 3
      CALL EFFECT(IROT)
      IEND = NSP
C
C****
  400 IF(IEND .EQ. 0) GO TO 420
      DO 410 J=1,NSP
      READ(ITAPE3) B(3)
      READ(ITAPE3) MAX,(X(I,4),I=1,MAX)
      READ(ITAPE3) MAX,(X(I,5),I=1,MAX)
  410 CONTINUE
  420 READ(ITAPE3,END=460,ERR=450) J
```

```
      450 PRINT 1020
          GO TO 500
      460 PRINT 1050
      500 CONTINUE
C
     1000 FORMAT(1H1,48H SENS-5D CALCULATIONS BEGIN -----------------,//)
     1001 FORMAT(///,21H INPUT DATA SET NO. 1,//)
     1002 FORMAT(1H1,21H INPUT DATA SET NO. 2,//)
     1003 FORMAT(A61,A19)
     1020 FORMAT(///,33H END-OF-FILE NOT FOUND ON ITAPE3.)
     1030 FORMAT(19HSENS-5D TABLES FOR ,A61)
     1050 FORMAT(///,32H THIS CONCLUDES THE CALCULATION.)
          STOP
          END
```

```
      SUBROUTINE APCAL
C     CALCULATION AT THE APOGEE POINT.
C
      INTEGER RKSTEP
      REAL LATL,LONGL,LAT,LONG,IPT,IPAZ,IPR
      DOUBLE PRECISION DRS,DPC
      COMMON/STOR01/ G0,GR,OMEGA,GM,RE,RQ,ESQ,BSQI,ESQ1
      COMMON/STOR03/ X0,Y0,Z0,R,RXY,ALT,SR,HLIM
      COMMON/STOR04/ GDLATL,LATL,LONGL,THETAL,PHIL,GDLAT,LAT,LONG,THETA,
     ,               PHI,GAMAL,PSIL,GAMA,PSI,ELGDL,AZGDL,ELGD,AZGD,EL,
     ,               AZ,SRAZ
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     ,               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR08/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),SP(6,20)
      COMMON/STOR13/ APT,APAZ,APA,APR,BOT,BOA,BOR,BOVL,BOEL,BOAZ,IPT,
     ,               IPAZ,IPR
C
      RATIO = RE/R
      XS = RATIO*S(1)
      YS = RATIO*S(2)
      ZS = RATIO*S(3)
      CALL AZRAN(XS,YS,ZS,T)
      APT = T
      APA = ALT
      APR = SR
      APAZ = SRAZ
      RETURN
      END
```

```
      AR1 = LAT
      AR2 = LATL
      AR3 = LONG - LONGL
      DG = DCOS(AR1)
      DE = DG*DSIN(AR3)
      DF = DCOS(AR2)*DSIN(AR1) - DG*DSIN(AR2)*DCOS(AR3)
      SRAZ = DATAN2(DE,DF)
      IF(SRAZ.LT.0.) SRAZ=SRAZ+TRI
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE BAI(IBOT)
C     CALCULATIONS FOR BURNOUT, APOGEE AND IMPACT.
C
      INTEGER RKSTEP
      REAL LATL,LONGL,LAT,LONG,MASS,MPROP,MPL,MDOT,MACH,LCG,LCP,IYY,IPT,
     .     IPAZ,IPR,MSEP
      DOUBLE PRECISION DPS,DPC
      CHARACTER NAME*80
      EXTERNAL SOLV3D,SOLV5D
      DIMENSION SAVE(61,15)
      COMMON/STOR00/ ZERO,ONE,PI,QPI,HPI,TPI,CF(9)
      COMMON/STOR01/ G0,GR,OMEGA,GM,RE,RG,ESQ,ESQI,ESQ1
      COMMON/STOR03/ X0,Y0,Z0,R,RXY,ALT,SR,HLIM
      COMMON/STOR04/ GDLATL,LATL,LONGL,THETAL,PHIL,GDLAT,LAT,LONG,THETA,
     .               PHI,GAMAL,PSIL,GAMA,PSI,ELGDL,AZGDL,ELGD,AZGD,EL,
     .               AZ,SRAZ
      COMMON/STOR05/ VT(3),VA(3),VEL,ACC,VQLIM,VLIM
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     .               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR08/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),SP(6,20)
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,CA,AD,QS,
     .               SLOPE,CMP,LCG,LCP,IYY,SY,SZ
      COMMON/STOR10/ WE,WN,MLAST,WIND,WNAZ,ALOW,AHIGH,NLEV,ALEV(100),
     .               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSC,NCON
      COMMON/STOR13/ APT,APAZ,APA,APR,BOT,BOA,BOR,BOVL,BOEL,BOAZ,IPT,
     .               IPAZ,IPR
      COMMON/STOR14/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPUNCH,INDEX
      COMMON/STOR99/ NAME,ARRAY(800)
      COMMON NPST,NSEP,NSP,NM(14),KSP(20),RMASS,T3D,TAX(3),TANG(61),B(4),
     .       PLM(20),PHT(32),TSEP(32),MSEP(32),TSP(20),SPM(20),A(99,14)
      EQUIVALENCE (ARRAY(1),SAVE(1,1))
C
      SAVEO = OMEGA
      IF(IROT.EQ.0) OMEGA=1.0E-10
      DO 700 IP=1,NPL
      MPL = PLM(IP)
      PRINT 1000
      PRINT 1010, NAME,MPL*QO,AZGDL*CF(1)
      IF(MLAST .EQ. 0) PRINT 1011
      IF(MLAST .EQ. 3) PRINT 1012, WIND,WNAZ*CF(1),ALOW,AHIGH
      IF(NLEV .GT. 0) PRINT 1013
```

```
      IF(IROT.EQ.0) PRINT 1014
      IF(IROT.NE.0) PRINT $015
      DO 600 IA=1,NANG
      ELGDL = ANG(IA)
      IF(IPRINT.NE.0) PRINT 1016, ELGDL*CF(1)
C
C     BEGIN 5-D CALCULATIONS
      NCON = 0
      CALL SETIV
      IF(MLAST .GT. 0) NCON=2
      CALL RK4(14,$OLV5D)
  200 CALL PCM(14,$OLV5D)
      IF(T-T3D) 210,245,245
  210 CALL APSTEP($200,$220)
  220 CALL RK4(14,$OLV5D)
      CALL APSTEP($220,$225)
  225 IF(ABS(VT(3)) .GT. VLIM) GO TO 220
      GO TO 350
C
C     BEGIN 3-D CALCULATIONS
  245 KSTEP = 6
  250 CALL PCM(6,$OLV3D)
      CALL APSTEP($250,$320)
  320 CALL RK4(6,$OLV3D)
      CALL APSTEP($320,$325)
  325 IF(ABS(VT(3)) .GT. VLIM) GO TO 320
  350 CALL APCAL
      RKSTEP = 3
      DT = AMAX1(DT,DTAI)
      KSTEP = 2
      CALL RK4(6,$OLV3D)
  380 CALL PCM(6,$OLV3D)
      CALL IPSTEP($380,$390)
  390 CALL RK4(6,$OLV3D)
      IF(ALT.LT.HLIM) GO TO 400
      CALL IPSTEP($390,$390)
  400 CALL IPCAL
C
      SAVE(IA,01) = ANG(IA)*CF(1)
      SAVE(IA,02) = APT
      SAVE(IA,03) = APA
      SAVE(IA,04) = APB*CF(4)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SAVE(IA,05) = BOY
      SAVE(IA,06) = BOA
      SAVE(IA,07) = BOB*CF(4)
      SAVE(IA,08) = BOVL
      SAVE(IA,09) = BOEL*CF(1)
      SAVE(IA,10) = BOAZ*CF(1)
      SAVE(IA,11) = IPT
      SAVE(IA,12) = IPB*CF(4)
      SAVE(IA,13) = IPAZ*CF(1)
C
      IF(WSRENT.EQ.0 .OR. NSP.EQ.0) GO TO 600
      DO 500 IS=1,NSP
      READ(ITAPE3) B(3)
      READ(ITAPE3) MAX,(A(I,4),I=1,MAX)
      READ(ITAPE3) MAX,(A(I,5),I=1,MAX)
      WRITE(ITAPE4) TSP(IS),SPM(IS),B(3)
      WRITE(ITAPE5) MAX,(A(I,4),I=1,MAX)
      WRITE(ITAPE6) MAX,(A(I,5),I=1,MAX)
      WRITE(ITAPE7) (SP(I,IS),I=1,6)
  500 CONTINUE
  600 CONTINUE
      IF(IPRINT.NE.0)PRINT 1001
      PRINT 1020
      DO 610 IA=1,NANG
      PRINT 1030, (SAVE(IA,I),I=1,13)
  610 CONTINUE
      IF(IPUNCH .EQ. 0) GO TO 700
      PUNCH 1020, NANG,(SAVE(IA,1),IA=1,NANG)
      PUNCH 1091, (SAVE(IA,12),IA=1,NANG)
      PUNCH 1092, (SAVE(IA,09),IA=1,NANG)
      PUNCH 1093, (SAVE(IA,03),IA=1,NANG)
  700 CONTINUE
      OMEGA = SAVED
C
 1000 FORMAT(1H1,2HNASA WALLOPS FLIGHT CENTER,/,25H WALLOPS ISLAND, VIR
     ,GINIA,//,50H TRAJECTORY SUMMARY AT BURN-OUT, APOGEE AND IMPACT,//)
 1001 FORMAT(1H1)
 1010 FORMAT(13H VEHICLE   = ,A80,/,18H PAY LOAD  = ,F9.2,4H LBS,/,
     ,13H LAUNCH AZ = ,F7.2,4H DEG,/)
 1011 FORMAT(13H WIND      = ,4HZERO)
 1012 FORMAT(13H WIND      = ,F7.2,8H FT/SEC,,F7.2,13H DEG NORTH AZ,/,
     ,13X,5H FROM,F9.1,6H FT TO,F9.1,3H FT)
```

```
 1013 FORMAT(13H WIND       = ,30HVARIABLE WIND SUPPLIED BY USER)
 1014 FORMAT(13H EARTH      = ,18HNON-ROTATING MODEL)
 1015 FORMAT(13H EARTH      = ,14HROTATING MODEL)
 1016 FORMAT(1H1,55HDETAILED PRINT-OUT OF TRAJECTORY FOR LAUNCH ELEVATIO
     ,N =,F7.2,4H DEG,/,54H UNITS ARE F-P-S-DEGREE, EXCEPT RANGE(NM) AND
     , ACC(GO),,//)
 1020 FORMAT(///,19X,21HA   P   O   G   E   E,23X,27HB   U   R   N   O
     ,   U   T,23X,21HI   M   P   A   C   T,/,11X,20(1H*),5X,58(1H*),5X,
     ,29(1H*),/,2X,2HEL,8X,4HTIME,8X,3HALT,7X,5HRANGE,8X,4HTIME,8X,3HALT
     ,,7X,5HRANGE,8X,3HVEL,5X,6HFLT/EL,3X,6HFLT/AZ,7X,4HTIME,6X,5HRANGE,
     ,5X,2HAZ,7,1X,5H(DEG),6X,5H(SEC),7X,4H(FT),6X,4H(NM),7X,5H(SEC),7X,
     ,4H(FT),6X,4H(NM),4X,8H(FT/SEC),4X,5H(DEG),4X,5H(DEG),7X,5H(SEC),
     ,5X,4H(NM),5X,5H(DEG),//)
 1030 FORMAT(F6.2,F12.2,F12.0,F9.1,F12.2,F12.0,F9.1,F11.2,F9.2,F10.2,
     ,F12.2,F9.1,F9.2)
 1090 FORMAT(1H, LAUNCH ELEVATIONS (DEG) FOR IMPACT RANGE, B-O FLT-EL AN
     ,D APOGEE ALTI,/,13X,(9(1X,F6.2)))
 1091 FORMAT(1IMPACT RANGE IN NM,/,(F7.2,8(1X,F7.2)))
 1092 FORMAT(1B-O FLT/EL IN DEG,/,(F7.2,8(1X,F7.2)))
 1093 FORMAT(1APOGEE ALTITUDE IN FT,/,(F9.0,6(1X,F9.0)))
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE BOCAL
C     CALCULATION AT BURN-OUT POINT,
C
      INTEGER RKSTEP
      REAL IPT,IPAZ,IPR
      DOUBLE PRECISION DPS,DPC,AR1,AR2
      COMMON/STOR00/ ZERO,ONE,PI,QPI,HPI,TPI,CF(9)
      COMMON/STOR01/ G0,GR,OMEGA,GH,RE,RQ,ESQ,ESQI,ESQ1
      COMMON/STOR03/ X0,Y0,Z0,R,RXY,ALT,SR,HLIM
      COMMON/STOR02/ VT(3),VA(3),VEL,ACC,VOLIM,VLIM
      COMMON/STOR05/ RTI(9)
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     ;               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR08/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),BP(6,20)
      COMMON/STOR10/ WE,WR,MLAST,WIND,WNAZ,ALON,AHIGH,NLEV,ALEV(100),
     ;               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR15/ APT,APAZ,APR,ABR,BOT,BOA,BOR,BOVL,BOEL,BOAZ,IPT,
     ;               IPAZ,IPR
C
      S1=S(4)+OMEGA*S(2)
      S2=S(5)-OMEGA*S(1)
      VT(1) = RTI(1)*S1 + RTI(2)*S2 + RTI(3)*S(6)
      VT(2) = RTI(4)*S1 + RTI(5)*S2 + RTI(6)*S(6)
      VT(3) = RTI(7)*S1 + RTI(8)*S2 + RTI(9)*S(6)
      BOT = T
      BOA = ALT
      S1 = VT(1)**2 + VT(2)**2
      BOVL = SQRT(S1 + VT(3)**2)
      AR1=-VT(3)
      AR2=SQRT(S1)
      BOEL = DATAN(AR1/AR2)
      AR1=VT(2)
      AR2=VT(1)
      BOAZ=DATAN2(AR1,AR2)
      IF(BOAZ.LT.0?) BOAZ=BOAZ+TPI
      S1 = RE/R
      CALL AZRAN(S1*S(1),S1*S(2),S1*S(3),T)
      BOR = SR
      RETURN
      END
```

```
      SUBROUTINE EFFECT(IROT)
C     UNIT-WIND EFFECTS, CORIOLIS DEFLN AND RANGE DERIVATIVES
C
      REAL LATL,LONGL,LAT,LONG,MASS,MPROP,MPL,MDOT,MACH,LCG,LCP,IYY,IPT,
     ;     IPAZ,IPR,MSEP
      CHARACTER NAME*80
      DIMENSION SAVE(61,9)
      COMMON/STOR00/ ZERO,ONE,PI,QPI,HPI,TPI,CF(9)
      COMMON/STOR01/ G0,GR,OMEGA,GH,RE,RQ,ESQ,ESQI,ESQ1
      COMMON/STOR04/ GDLATL,LATL,LONGL,THETAL,PHIL,GDLAT,LAT,LONG,THETA,
     ;               PHI,GAMAL,PSIL,GAMA,PSI,ELGDL,AZGDL,ELGD,AZGD,EL,
     ;               AZ,SRAZ
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,CA,AD,QS,
     ;               SLOPE,CMP,LCG,LCP,IYY,SY,SZ
      COMMON/STOR10/ WE,WN,MLAST,WIND,WNAZ,ALON,AHIGH,NLEV,ALEV(100),
     ;               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSC,NCON
      COMMON/STOR13/ APT,APAZ,APA,APR,BOT,BOA,BOR,BOVL,BOEL,BOAZ,IPT,
     ;               IPAZ,IPR
      COMMON/STOR14/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPONCH,INDEX
      COMMON/STOR99/ NAME,ARRAY(800)
      COMMON NPST,NSEP,NSP,NM(14),KSP(20),RMASS,T3D,TAV(3),ANG(61),B(4),
     ;       PLM(20),PHT(32),TSEP(32),MSEP(32),TSP(20),SPMT(20),A(99,14)
      EQUIVALENCE (ARRAY(1),SAVE(1,1))
C
      PRINT 1000
      PRINT 1010, NAME,MPL,G0,AZGDL,CF(1),WIND,WIND*CF(6)
      IF(IROT.EQ.0) PRINT 1011
      IF(IROT.NE.0) PRINT 1012
      SAVEO = OMEGA
      DEL = -0.5*CF(2)
      CFC = CF(2)*CF(4)
C
      DO 100 J=1,NANG
      ELGDL= ANG(J)
      SAVE(J,1) = ELGDL*CF(1)
C
C     NO-WIND IMPACT AND CORIOLIS DEFLECTION -----------------------
      IF(IPRINT.NE.0) PRINT 1020, SAVE(J,1)
      OMEGA = 1.0E-10
      NCON = 0
      CALL SETIV
```

```
      CALL SPLASH
      UN = IPR
      PN = IPAZ
      IF(IPRINT.NE.0) PRINT 1021, SAVE(J,1)
      OMEGA = SAVEO
      CALL SETIV
      CALL SPLASH
      SAVE(J,4) = (IPR*COS(IPAZ)-UN*COS(PN))*CF(4)
      SAVE(J,5) = (IPR*SIN(IPAZ)-UN*SIN(PN))*CF(4)
      IF(IROT.NE.0) GO TO 10
      OMEGA = 1.0E-10
      GO TO 20
   10 UN = IPR
      PN = IPAZ
   20 SAVE(J,2) = UN*CF(4)
      SAVE(J,3) = PN*CF(1)
C
C     HEAD-WIND RANGE, ------------------------------------------
      IF(IPRINT.NE.0) PRINT 1023, SAVE(J,1)
      NCON = 2
      CALL SETIV
      WNAZ = PSI
      CALL WCONST
      CALL SPLASH
      TERM = IPR*(SIN(IPAZ)*SIN(PN)+COS(IPAZ)*COS(PN)) - UN
      SIGN = TERM/ABS(TERM)
      TERM = SQRT(IPR*IPR*UN*UN-2.0*IPR*UN*COS(IPAZ-PN))
      HEAD = (SIGN*TERM)/WIND
      SAVE(J,7) = HEAD*CF(4)
C
C     TAIL-WIND RANGE, ------------------------------------------
      IF(IPRINT.NE.0) PRINT 1024, SAVE(J,1)
      CALL SETIV
      WNAZ = PSI + PI
      IF(WNAZ.GE.TPI) WNAZ=WNAZ-TPI
      CALL WCONST
      CALL SPLASH
      TERM = IPR*(SIN(IPAZ)*SIN(PN)+COS(IPAZ)*COS(PN)) - UN
      SIGN = TERM/ABS(TERM)
      TERM = SQRT(IPR*IPR*UN*UN-2.0*IPR*UN*COS(IPAZ-PN))
      TAIL = (SIGN*TERM)/WIND
      SAVE(J,8) = TAIL*CF(4)
```

```
C
C     CROSS-WIND RANGE, ------------------------------------------
      IF(IPRINT.NE.0) PRINT 1025, SAVE(J,1)
      CALL SETIV
      WNAZ = PSI + HPI
      IF(WNAZ.GE.TPI) WNAZ=WNAZ-TPI
      CALL WCONST
      CALL SPLASH
      SIGN = 1.0
      TERM = SQRT(IPR*IPR*UN*UN-2.0*IPR*UN*COS(PAZ-PN))
      CROSS = (SIGN*TERM)/WIND
      SAVE(J,9) = CROSS*CF(4)
C
C     TOWER TILT, ------------------------------------------
      IF(IPRINT.NE.0) PRINT 1022, SAVE(J,1)
      ELGDL = ANGE(J) + DEL
      NCON = 0
      CALL SETIV
      CALL SPLASH
      SIGN = (IPR-UN)/ABS(IPR-UN)
      TERM = SQRT(IPR*IPR*UN*UN-2.0*IPR*UN*COS(PAZ-PN))
      TTILT = (SIGN*TERM)/DEL
      SAVE(J,6) = TTILT*CFB
  100 CONTINUE
C
      IF(IPRINT.NE.0) PRINT 1026
      PRINT 1030
      DO 110 J=1,NANG
      PRINT 1050, (SAVE(J,I),I=1,9)
  110 CONTINUE
      IF(IPUNCH .EQ. 0) GO TO 120
      PUNCH 1090, NANG,(SAVE(J,1),J=1,NANG)
      PUNCH 1091, (SAVE(J,7),J=1,NANG)
      PUNCH 1092, (SAVE(J,8),J=1,NANG)
      PUNCH 1093, (SAVE(J,9),J=1,NANG)
      PUNCH 1094, (SAVE(J,6),J=1,NANG)
      PUNCH 1095, (SAVE(J,4),J=1,NANG)
      PUNCH 1096, (SAVE(J,5),J=1,NANG)
  120 PRINT 1040
      CFC = CF(8)/CF(6)
      DO 130 J=1,NANG
      SAVE(J,2) = SAVE(J,2)*CF(8)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SAVE(J,4) = SAVE(J,4)*CF(8)
      SAVE(J,5) = SAVE(J,5)*CF(8)
      SAVE(J,6) = SAVE(J,6)*CF(8)
      SAVE(J,7) = SAVE(J,7)*CFC
      SAVE(J,8) = SAVE(J,8)*CFC
      SAVE(J,9) = SAVE(J,9)*CFC
      PRINT 1050, (SAVE(J,I),I=1,9)
  130 CONTINUE
      OMEGA = SAVED
C
 1000 FORMAT(1H1,25HNASA WALLOPS FLIGHT CENTER//,25H WALLOPS ISLAND, VIR
     ,GINIA,//,60H UNIT-WIND EFFECTS, CORIOLIS DEFLECTION AND RANGE DERI
     ,VATIVE,//)
 1010 FORMAT(13H VEHICLE   = ,A80,/,13H PAY LOAD  = ,F8.2,4H LBS,/,
     ,13H LAUNCH AZ = ,F7.2,4H DEG,/,13H WIND      = ,F7.2,9H FT/SEC (,
     ,F7.2,8H M/SEC),)
 1011 FORMAT(13H EARTH     = ,18HNON-ROTATING MODEL)
 1012 FORMAT(13H EARTH     = ,14HROTATING MODEL)
 1020 FORMAT(1H1,56HDETAILED TRAJECTORY FOR   NO  WIND CASE WITH ELEVATI
     ,ON =,F7.2,4H DEG,/,54H UNITS ARE F*P*S-DEGREE, EXCEPT RANGE(NM) AN
     ,D ACC(GO),,/,29H EARTH IS NON-ROTATING MODEL,,//)
 1021 FORMAT(1H1,56HDETAILED TRAJECTORY FOR   NO  WIND CASE WITH ELEVATI
     ,ON =,F7.2,4H DEG,/,54H UNITS ARE F*P*S-DEGREE, EXCEPT RANGE(NM) AN
     ,D ACC(GO),,/,25H EARTH IS ROTATING MODEL,,//)
 1022 FORMAT(1H1,56HDETAILED TRAJECTORY FOR RANGE DERV CASE WITH ELEVATI
     ,ON =,F7.2,4H DEG,/,54H UNITS ARE F*P*S-DEGREE, EXCEPT RANGE(NM) AN
     ,D ACC(GO),,//)
 1023 FORMAT(1H1,56HDETAILED TRAJECTORY FOR HEAD  WIND CASE WITH ELEVATI
     ,ON =,F7.2,4H DEG,/,54H UNITS ARE F*P*S-DEGREE, EXCEPT RANGE(NM) AN
     ,D ACC(GO),,//)
 1024 FORMAT(1H1,56HDETAILED TRAJECTORY FOR TAIL  WIND CASE WITH ELEVATI
     ,ON =,F7.2,4H DEG,/,54H UNITS ARE F*P*S-DEGREE, EXCEPT RANGE(NM) AN
     ,D ACC(GO),,//)
 1025 FORMAT(1H1,56HDETAILED TRAJECTORY FOR CROSS WIND CASE WITH ELEVATI
     ,ON =,F7.2,4H DEG,/,54H UNITS ARE F*P*S-DEGREE, EXCEPT RANGE(NM) AN
     ,D ACC(GO),,//)
 1026 FORMAT(1H1)
 1030 FORMAT(//,13H F*P*S SYSTEM,//,
     ,1X,9HLAUNCH EL,7X,15HNO-WIND  IMPACT,7X,20HCORIOLIS  DEFLECTION,
     ,5X,11HRANGE DERIV,16X,28HUNIT   -   WIND      EFFECTS,/,
     ,3X,5H(DEG),7X,9HRANGE(NM),3X,7HAZ(DEG),5X,9HNORTH(NM),3X,
     ,8HEAST(NM),6X,8H(NM/DEG),10X,12HHEAD(NM/FPS),5X,12HTAIL(NM/FPS),
```

```
     ,5X,13HCROSS(NM/FPS)//)
1040 FORMAT(//,13H M-R-S SYSTEM//,
     ,1X,9HLAUNCH EL,7X,15HNO-WIND  IMPACT,7X,20HCORIOLIS  DEFLECTION,
     ,5X,11HRANGE DERIV,16X,28HUNIT   -   WIND      EFFECTS,/,
     ,3X,5H(DEG),7X,9HRANGE(KM),3X,7HAZ(DEG),5X,9HNORTH(KM),3X,
     ,8HEAST(KM),6X,8H(KM/DEG),10X,12HHEAD(KM/MPS),5X,12HTAIL(KM/MPS),
     ,5X,13HCROSS(KM/MPS)//)
1050 FORMAT(F8.2,F14.2,F11.2,F14.2,F11.2,F15.5,10X,F10.5,7X,F10.5,7X,
     ,F10.5)
1090 FORMAT('N, LAUNCH ELEVATIONS (DEG) FOR UNIT-WIND EFFECTS, RANGE DE
     ,RIV AND CORIOLIS DEFL',/,I8,(9(1X,F6.2)))
1091 FORMAT('HEAD UNIT-WIND EFFECT (NM/FT/SEC)',/,(F7.3,8(1X,F7.3)))
1092 FORMAT('TAIL UNIT-WIND EFFECT (NM/FT/SEC)',/,(F7.3,8(1X,F7.3)))
1093 FORMAT('CROSS UNIT-WIND EFFECT (NM/FT/SEC)',/,(F7.3,8(1X,F7.3)))
1094 FORMAT('RANGE DERIV (NM/DEG)',/,(F7.3,8(1X,F7.3)))
1095 FORMAT('CORIOLIS DEFLECTION TO NORTH (NM)',/,(F7.3,8(1X,F7.3)))
1096 FORMAT('CORIOLIS DEFLECTION TO EAST (NM)',/,(F7.3,8(1X,F7.3)))
     RETURN
     END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE FZCURV(IROT)
C     CALCULATION OF WIND*WEIGHTING FACTOR AND BALLISTIC WIND FACTOR
C
      REAL LATL,LONGL,LAT,LONG*MASS,MPROP,MPL,MDOT,MACH,LCG,LCP,IYY,IPY,
     ,     IPAZ,IPR
      CHARACTER NAME*80
      DIMENSION RG(100),FZ(100),DFZ(100)
      COMMON/STOR00/ ZERO,ONE,PI,QPI,HPI,TPI,CF(9)
      COMMON/STOR01/ G0,GR,OMEGA,GM,RE,RO,ESQ,ESQI,ESQ1
      COMMON/STOR04/ GDLATL,LATL,LONGL,THETAL,PHIL,GDLAT,LAT,LONG,THETA,
     ,               PHI,GAMAL,PSIL,GAMA,PSI,ELGDL,AZGDL,ELGD,AZGD,EL,
     ,               AZ,SRAZ
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,CA,AD,QS,
     ,               SLOPE,CMP,LCG,LCP,IYY,SV,SZ
      COMMON/STOR10/ WE,WN,MLAST,WIND,WNAZ,ALOW,AHIGH,NLEV,ALEV(100),
     ,               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSO,NCON
      COMMON/STOR13/ ACT,APAZ,APA,APR,BOT,BOA,BOR,BOVL,BOEC,BOAZ,IPY,
     ,               IPAZ,IPR
      COMMON/STOR14/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPUNCH,INDEX
      COMMON/STOR99/ NAME,ARRAY(800)
      EQUIVALENCE (ARRAY(1),RG(1)),(ARRAY(101),FZ(1)),
     ,            (ARRAY(201),DFZ(1))
C
      PRINT 1000
      PRINT 1010, NAME,MRL*GO,ELGDL*CF(1),AZGDL*CF(1),WIND,WIND*CF(6),
     ,     WNAZ*CF(1)
      IF(IROT.EQ.0) PRINT 1011
      IF(IROT.NE.0) PRINT 1012
      SAVEO = OMEGA
      IF(IROT .EQ. 0) OMEGA=1.0E-10
C
C     INITIALIZE TO NO WIND
      IF(IPRINT.NE.0) PRINT 1020
      NCON = 0
      CALL SETIV
      CALL SPLASH
      UN = IPR
      PN = IPAZ
      RG(1) = UN
      FZ(1) = 0.
      DFZ(1) = 0.
```

```
C
C     INITIALIZE FOR MAXIMUM WIND ALTITUDE.
      ALOW = 0.
      AHIGH = ALEV(NLEV)
      IF(IPRINT.NE.0) PRINT 1021, ALOW,AHIGH
      NCON = 2
      CALL SETIV
      CALL NCONST
      CALL SPLASH
      DNOM = SQRT(IPR*IPR*UN*UN-2.0*IPR*UN*COS(IPAZ-PN))
      RG(NLEV) = IPR
      FZ(NLEV) =1.0
C
C     WIND UPTO OTHER INTERMEDIATE ALTITUDES.
      NENB = NLEV - 1
      DO 100 N=2,NEND
      AHIGH = ALEV(N)
      IF(IPRINT.NE.0) PRINT 1021, ALOW,AHIGH
      CALL SETIV
      CALL NCONST
      CALL SPLASH
      ANUM = SQRT(IPR*IPR*UN*UN-2.0*IPR*UN*COS(IPAZ-PN))
      RG(N) = IPR
      FZ(N) = ANUM/DNOM
      DFZ(N) = FZ(N) - FZ(N-1)
  100 CONTINUE
      DFZ(NLEV) = FZ(NLEV) - FZ(NEND)
C
      IF(IPRINT.NE.0) PRINT 1025
      PRINT 1030
      ANUM = CF(4)*CF(8)
      DO 110 N=1,NLEV
      DNOM = ALEV(N)*CF(5)
      PRINT 1040, N,ALEV(N),DNOM,RG(N)*CF(4),RG(N)*ANUM,FZ(N),DFZ(N),
     ,      ALEV(N),DNOM
  110 CONTINUE
      IF(IPUNCH .EQ. 0) GO TO 120
      PUNCH 1090, NLEV,(ALEV(N),N=1,NLEV)
      PUNCH 1091, (FZ(N),N=1,NLEV)
      PUNCH 1092, (DFZ(N),N=1,NLEV)
  120 OMEGA = SAVEO
C
```

```
 1000 FORMAT(1H1,28HNASA WALLOPS FLIGHT CENTER,/,23H WALLOPS ISLAND, VIR
     ;GINIA//,30H F(Z)-CURVE AND BALLISTIC WIND FACTORS,//)
 1010 FORMAT(13H VEHICLE    = ,A80,/,15H PAY LOAD   = ,F9.2,4H LBS,/,
     ;13H LAUNCH EL = ,F7.2,4H DEG,/,13H LAUNCH AZ = ,F7.2,4H DEG,/,
     ;13H WIND      = ,F7.2,7H FT/SEC,2H (,F6.2,9H M/SEC),F9.2,18H DEG A
     ,Z FROM NORTH)
 1011 FORMAT(15H EARTH     = ,18HNON-ROTATING MODEL)
 1012 FORMAT(13H EARTH     = ,14HROTATING MODEL)
 1020 FORMAT(1H1,92HDEVIATED TRAJECTORY FOR F(Z) WITH NO WIND AND WHERE,
     ;/,54H UNITS ARE M-M-S,DEGREE, EXCEPT RANGE(NM) AND ACC(GO),,Z/)
 1021 FORMAT(1H1,89HDEVIATED TRAJECTORY FOR F(Z) WITH WIND FROM,F10.0,
     ;4H TO ,F10.0,4Z/,59H UNITS ARE M-M-S,DEGREE, EXCEPT RANGE(NM) AND AC
     ;C(G0),,/)
 1025 FORMAT(1H1)
 1030 FORMAT(///,98,4HZ,9X,8HWIND ALT,12X,13HIMPACT  RANGE,11X,4HF(Z),9X
     ,,9HDF(Z),14X,8HWIND ALT,/,10X,4H(FT),7X,3HVM,8X,4H(NM),8X,4H(KM),
     ;19X,18HF(Z)-F(Z-1),8X,4H(FT),,9X,8H(M),,//)
 1040 FORMAT(14,F10.0,F11.0,F10.2,F14.2,F13.5,F13.7,F14.0,F11.0)
 1090 FORMAT(/N, ALTITUDE TABLE FOR F(Z) AND DELTA F(Z) CURVES*,/,
     ;13,(6(1X,F9.0)))
 1091 FORMAT('F(Z) CURVE',/,(F8.0,7(1X,F8.4)))
 1092 FORMAT('DELTA F(Z) CURVE',/,(F8.4,7(1X,F8.4)))
      RETURN
      END
```

```
      SUBROUTINE IPCAL
C     CALCULATION AT IMPACT POINT.
C
      INTEGER RKSTEP
      REAL IPT,IPAZ,IPB
      DOUBLE PRECISION DPS,DPC
      COMMON/STOR05/ X0,Y0,Z0,R,RXW,ALT,SR,HLIH
      COMMON/STOR04/ GDLNTL,LATL,LONGL,THETAL,PHIL,GDLAT,LAT,LONG,THETA,
     ;               PHI,GAMAL,PSIL,GAMA,PSI,ELGDL,AZGDL,ELGD,AZGD,EL,
     ;               AZ,SRAZ
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     ;               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR08/ S(14),F(14,5),DRS(14),DPC(14),ERW(14),BP(6,20)
      COMMON/STOR13/ APT,APAZ,APB,APR,BOT,BOA,BOR,BOVL,BOEL,BOAZ,IPT,
     ;               IPAZ,IPR
C
      CALL AZRAN(S(1),S(2),S(3),T)
      IPT = T
      IPR = SR
      IPAZ = SRAZ
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE IPSTEP(*,*)
C     ADJUSTMENT FOR STEP SIZE AS IMPACT CLOSES,
C
      INTEGER NKSTEP
      COMMON/STOR01/ G0,GR,OMEGA,GM,RE,RO,ESO,BSDI,ESQ1
      COMMON/STOR03/ X0,Y0,Z0,R,RXY,ALT,SR,HLIM
      COMMON/STOR05/ VY(3),VA(5),VEL,ACG,VOLIM,VLIM
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     ;               DMIN,DMAX,KSTEP,NGOOD
C
      G2 = G0/2.0
      TT = 8.0*DT
      DIST = (VEL+G2*TT)*TT
      IF(ALT .LT. DIST) KSTEP=10000
      TT = DT*2.0
      DIST = (VEL+G2*TT)*TT
      IF(DIST .LT. ALT) RETURN 1
      RKSTEP = 1
      TT = ALT/(VEL+G0*DT)
      DT = AMIN1(TT,DMAX)
      RETURN 2
      END
```

```
      SUBROUTINE LINT(F,X,FI,XI,ORDER,MAX,KB,KE,KKB,KKE)
C     LINEAR INTERPOLATION BY AITKEN ITERATION METHOD.
C
      INTEGER ORDER
      DIMENSION F(1),X(1),FI(1),XI(1)
      DIMENSION FFDET(11,11)
C
      IORD = ORDER
      IF(MAX-ORDER) 10,10,20
   10 N = ORDER - MAX
      ORDER = ORDER-N-1
   20 KI = KB
      DO 100 KK=KKB,KKE
      XNEW = XI(KK)
      DO 30 K=KI,KE
      KSAVE = K
      IF(X(K).GT.XNEW) GO TO 40
   30 CONTINUE
   40 IZERO = KSAVE - (ORDER+1)/2
      IF(IZERO .LT. 1) IZERO=1
      IF((IZERO+ORDER) .GT. MAX) IZERO=MAX-ORDER
      DO 50 JJ=1,ORDER
      N = JJ+ IZERO
      FFDET(1,JJ) = (F(IZERO)*(X(N)-XNEW)-F(N)*(X(IZERO)-XNEW))/
     ;(X(N)-X(IZERO))
   50 CONTINUE
      IF(ORDER.EQ.1) GO TO 70
      DO 65 I=2,ORDER
      I1 = I - 1
      L = I1 + IZERO
      DO 60 M=I,ORDER
      J = M + IZERO
      FFDET(I,M) = (FFDET(I1,I1)*(X(J)-XNEW) - FFDET(I1,M)*(X(L)-XNEW))/
     ;(X(J)-X(L))
   60 CONTINUE
   65 CONTINUE
   70 FI(KK) = FFDET(ORDER,ORDER)
      KI = KSAVE - 1
  100 CONTINUE
      ORDER = IORD
      RETURN
      END
                                                        LINT
```

```
      SUBROUTINE LISY(NFC)
C     PRINTS DATA STORED ON FILE NFC;
C
      CHARACTER NAME*80
      COMMON/STOR99/ NAME,ARRAY(800)
C
      REWIND NFC
  100 READ(NFC,1000,END=110) NAME
      PRINT 1000, NAME
      GO TO 100
  110 REWIND NFC
C
 1000 FORMAT(A80)
      RETURN
      END
```

```
      SUBROUTINE ABM(NF,SOLV)
C     MODIFIED ADAMS-BASHFORTH PREDICTOR-CORRECTOR METHOD,
C
      INTEGER RKSTEP
      DOUBLE PRECISION DPS,DPC,DPF
      DIMENSION P(14),C(14),PC(14)
      COMMON/STOR0/ T,DT,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     .              DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR0A/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),SP(6,20)
      COMMON/STOR1/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSC,NCON
      COMMON/STOR1A/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPUNCH,INDEX
      PARAMETER C1=55.0/24.0, C2=59.0/24.0, C3=37.0/24.0, C4=9.0/24.0,
     .          C5=19.0/24.0, C6=5.0/24.0, C7=1.0/24.0, C8=251.0/270.0,
     .          C9=19.0/270.0
C
C     PREDICTOR AND ITS MODIFIER,
      DO 100 I=1,NF
      DPF = C1*F(I,4)-C2*F(I,3)+C3*F(I,2)-C4*F(I,1)
      P(I) = DPS(I) + DT*DPF
      S(I) = P(I) - C8*DPC(I)
  100 CONTINUE
C
C     FUNCTION F(S,T+DT),
      T = T+DT
      CALL SOLV(S)
C
C     CORRECTOR, FINAL VALUE, AND MAXIMUM ERROR,
      EMAX = 0.
      DO 110 I=1,NF
      DPF = C4*F(I,5)+C5*F(I,4)-C6*F(I,3)+C7*F(I,2)
      C(I) = DPS(I) + DT*DPF
      PC(I) = P(I) - C(I)
      S(I) = C(I) + C9*PC(I)
      EMAX = EMAX + ERW(I)*ABS(PC(I))
  110 CONTINUE
      EMAX = EMAX/NF
C
C     STEP SIZE ADJUSTMENT ACCORDING TO ACCURACY REQUIREMENTS,
      IF((T+DT+1.E-05) .GE.TNP1) GO TO 400
      IF(EMAX .GT. EPBIG) GO TO 200
      IF(NGOOD .LT. KSTEP) GO TO 500
      IF(EMAX .LT. EPTINY) GO TO 300
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      GO TO 500
C
C     DECREASE THE STEP SIZE.
  200 TT = DT/2.0
      IF(TT .LT. DMIN) GO TO 500
      T = T-DT
      DT = TT
      IF((T+TT*4.3).GE.TNP1) DT=(TNP1-T)/4.0
      DO 210 I=1,NF
      S(I) = DPS(I)
  210 CONTINUE
      CALL RK4(NF,SOLV)
      RETURN
C
C     INCREASE THE STEP SIZE.
  300 TT = DT*2.0
      IF(TT .GT. DMAX) GO TO 500
      IF((T+TT*4.3).GE.TNP1) GO TO 500
      DT = TT
      DO 310 I=1,NF
      DPS(I) = S(I)
  310 CONTINUE
      CALL RK4(NF,SOLV)
      NGOOD = 1
      RETURN
C
C     PHASE CHANGES.
  400 TT = DT
      DO 410 I= 1,NF
      DPS(I) = S(I)
  410 CONTINUE
      RKSTEP = 1
      DT = TNP1*(1.0-1.0E-07) - T
      CALL RK4(NF,SOLV)
      T = TNP1
      IF(ABS(T-TB3) .LT. TBO*1.0E-06) CALL BOCAL
      IF(IPRINT .NE. 0) CALL TPRINT(5,1)
      RKSTEP = 3
      DT = TT
      CALL REPAZE
      CALL RK4(NF,SOLV)
      NGOOD = 1
```

```
      RETURN
C
C     INTEGRATION COMPLETE WITHIN ERROR BOUNDS.
  500 CALL SOLV(5)
      DO 510 I=1,NF
      DPS(I) = S(I)
      DPC(I) = PC(I)
      F(I,1) = F(I,2)
      F(I,2) = F(I,3)
      F(I,3) = F(I,4)
      F(I,4) = F(I,5)
  510 CONTINUE
      NGOOD = NGOOD+1
C
      IF(IPRINT.NE. 0) CALL TPRINT(5,0)
      RETURN
      END
```

```
      SUBROUTINE BDATA
C     READS ROCKET DATA AND WRITES ON A TEMPORARY TAPE,
C
      INTEGER ORDER
      REAL MSER
      LOGICAL IPR2
      CHARACTER NAME*80
      CHARACTER LINE*80,COL*1(80)
      DIMENSION F(99),X(99),FI(99),XI(99)
      COMMON/STOR00/ ZERO,ONE,PI,QP1,HPI,TPI,CF(9)
      COMMON/STOR01/ G0,AR,OMEGA,GM,RE,RO,ESO,ESQI,ESQ1
      COMMON/STOR02/ PBES0,DENS0,TEMP0,SOUND0,PRESB(9),TEMPB(9),
     ,               HALTB(9),TSLOPE(9),BONST,PRES,DENS,TEMP,SOUND
      COMMON/STOR11/ KPREV,NPREV,MPREV,APREV,LPREV(14),VPREV(14),
     ,               ORDER(14)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSM,NCON
      COMMON/STOR14/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPUNCH,INDEX
      COMMON/STOR99/ NAME,ARRAY(800)
      COMMON NPST,NSEP,NSP,NM(14),KSP(20),RMASS,T3D,TAV(3),ANG(61),B(4),
     ,       PLM(20),PHT(32),TSER(32),MSER(32),TSP(20),SPM(20),A(99,14)
      DATA SLCG,SLSY/2H0,/
      EQUIVALENCE (ITAPE2,IR),(ITAPE3,IW)
      EQUIVALENCE (ARRAY,F),(ARRAY(100),X),(ARRAY(199),FI),
     ,            (ARRAY(298),XI)
C
      I = NPMAX+2
      DO 1 J=1,I
      PHT(J) = 1.0E+10
      TSER(J) = 1.0E+10
      MSER(J) = 0.
    1 CONTINUE
      DO 2 J=1,NSMAX
      TSP(J) = 1.0E+10
    2 CONTINUE
      IPR2 = IPRINT .EQ. 2
C
      READ(IR,1000) NAME
      READ(IR,1000) LINE
      READ(IR,1001) RMASS,T3D
      READ(IR,1000) LINE
      READ(IR,1001) NPST,(PHT(I),I=1,NPST)
      READ(IR,1000) LINE
```

```
      READ(IR,1004) NSEP,(TSEP(I),I=1,NSEP)
      READ(IR,1008) LINE
      READ(IR,1004) NSEP,(MSEP(I),I=1,NSEP)
      RMASS = RMASS*CF(5)
      DO 10 I=1,NSEP
      MSEP(I) = MSEP(I)*CF(5)
   10 CONTINUE
      IF(.NOT.IPR2) GO TO 11
      PRINT 5000
      PRINT 5001, RMASS,TSD
      PRINT 5002, NPST,(PHT(I),I=1,NPST)
      PRINT 5003, NSEP,(TSEP(I),I=1,NSEP)
      PRINT 5004, NSEP,(MSEP(I),I=1,NSEP)
   11 CONTINUE

C     REGULAR PHASES
      DO 300 NP=1,NPST
      IF(IPR2) PRINT 5010, NP
      READ(IR,1008) LINE
      READ(IR,1009) LINE
      READ(IR,1004) (B(I),I=1,4)
      ADD = 0.
      DO 110 J=1,14
      READ(IR,1008) LINE
      IF(J.NE.2) GO TO 104
      DECODE(LINE,1002) COL
      DO 100 I=1,78
      IF(COL(I).EQ.'S'.AND.COL(I+1).EQ.'E'.AND.COL(I+2).EQ.'AA')
     .ADD = B(1)+PBESO
  100 CONTINUE
  104 READ(IR,1001) MAX,(A(I,J),I=1,MAX)
      IF(MAX.GT.1) GO TO 105
      MAX = 2
      A(2,J) = A(1,J)
  105 NM(J) = MAX
      IF(J.LT.7 .OR. J.GT.8) GO TO 110
      IF(MAX.NE.2) GO TO 110
      IF(A(1,7).LT.ZERO .AND. A(2,7).LT.ZERO) GO TO 107
  106 IF(A(1,8).LT.ZERO .AND. A(2,8).LT.ZERO) GO TO 108
      GO TO 110
  107 A(1,7) = SLCG
      A(2,7) = SLCG
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      GO TO 106
  108 A(1,8) = SIYY
      A(2,8) = SIYY
  110 CONTINUE
      SLCG = A(NM(X),7)
      SIYY = A(NM(8),8)
      DO 115 I=1,NVAR
      A(I,01) = A(I,01) * PHT(NP)
      A(I,02) = A(I,02) * KDD
      A(I,03) = A(I,03)*CF(2)
      A(I,06) = A(I,06) * PHT(NP)
      A(I,10) = A(I,10)*CF(1)
      A(I,12) = A(I,12)*CF(1)
  115 CONTINUE
C     INTERPOLATIONS BEGIN.
      IF(NM(6).EQ.2 .AND. ABS(A(1,6)*A(2,6)).LT.ZERO) GO TO 160
      DO 120 I=1,NVAR
      X(I) = A(I,6)
      XI(I) = A(I,1)
  120 CONTINUE
      DO 150 J=7,8
      MAX = NM(J)
      DO 130 I=1,MAX
      F(I) = A(I,J)
  130 CONTINUE
      CALL LINT(F,X,FI,XI,ORDER(8),MAX,1,NM(6),1,NM(1))
      MAX = NM(1)
      DO 140 I=1,MAX
      A(I,J) = FI(I)
  140 CONTINUE
      NM(J) = NM(1)
  150 CONTINUE
      GO TO 180
  160 CONTINUE
      MAX = NM(1)
      DO 170 I=3,MAX
      A(I,7) = A(1,7)
      A(I,8) = A(1,8)
  170 CONTINUE
      NM(7) = NM(1)
      NM(8) = NM(1)
  180 CONTINUE
```

```
      DO 230 K=9,15,2
      J = K + 1
      IF(NM(K).EQ.2 .AND. ABS(A(1,K)-A(2,K)).LT.ZERO) GO TO 210
      DO 190 I=1,NVAR
      F(I) = A(I,J)
      X(I) = A(I,K)
      XI(I) = A(I,4)
  190 CONTINUE
      MAX = NM(J)
      CALL LINT(F,X,FI,XI,ORDER(J),MAX,1,NM(K),1,NM(4))
      MAX = NM(4)
      DO 200 I=1,MAX
      A(I,J) = FI(I)
  200 CONTINUE
      NM(J) = NM(4)
      GO TO 230
  210 CONTINUE
      MAX = NM(4)
      DO 220 I=3,MAX
      A(I,J) = A(1,J)
  220 CONTINUE
      NM(J) = NM(4)
  230 CONTINUE
C     INTERPOLATIONS END.
      WRITE(IW) (B(I),I=1,4)
      IF(IPR2) PRINT 5011, (B(I),I=1,4)
      DO 290 J=1,14
      MAX = NM(J)
      WRITE(IW) MAX,(A(I,J),I=1,MAX)
      IF(IPR2) PRINT 5012, NP,J,MAX,(A(I,J),I=1,MAX)
  290 CONTINUE
  500 CONTINUE
C
C     SRENT STAGES.
      NSP = 0
  400 CONTINUE
      READ(IR,1000,END=450) LINE
      NSP = NSP+1
      IF(IPR2) PRINT 5020, NSP
      READ(IR,1000) LINE
      READ(IR,1001) TSP(NSP),SPM(NSP),B(3)
      SPM(NSP) = SPM(NSP)*GF(5)
```

```
      WRITE(IW) B(3)
      IF(IPR2) PRINT 5021, TSP(NSP),SPM(NSP),B(3)
      DO 410 J=4,5
      READ(IR,1000) LINE
      READ(IR,1001) MAX,(A(I,J),I=1,MAX)
      IF(MAX.GT.1) GO TO 405
      MAX = 2
      A(2,J) = A(J,1)
  405 NN(J) = MAX
      WRITE(IW) MAX,(A(I,J),I=1,MAX)
      IF(IPR2) PRINT 5012, NSP,J,MAX,(A(I,J),I=1,MAX)
  410 CONTINUE
      GO TO 400
  450 CONTINUE
C
 1000 FORMAT(A80)
 1001 FORMAT(V)
 1002 FORMAT(80A1)
 5000 FORMAT(1H1,21HINPUT DATA SET NO. 2,./,102H AFTER INTERPOLATIONS TO
     ; GET AERODYNAMIC DATA REFERRED TO ONE TIME VARIABLE AND ONE MACH N
     ;O. VARIABLE.)
 5001 FORMAT(/,14H ROCKET MASS =,1PE13.5,17H, BURN-OUT TIME =,1PE13.5)
 5002 FORMAT(/,19H NUMBER OF PHASES =,I2,15H, START TIMES =,/,
     ,(10(1PE13.5)))
 5003 FORMAT(/,25H NUMBER OF SPENT STAGES =,I2,15H, START TIMES =,/,
     ,(10(1PE13.5)))
 5004 FORMAT(/,25H NUMBER OF SPENT STAGES =,I2,15H, THEIR MASSES=,/,
     ,(10(1PE13.5)))
 5010 FORMAT(//,12H PHASE NO. =,I2,15H      ---------)
 5011 FORMAT(/,9H ARRAY B=,4(1PE13.5))
 5012 FORMAT(/,19H ARRAY A--NP,J,MAX=,3I5,/,(10(1PE13.5)))
 5020 FORMAT(//,18H SPENT STAGE NO. =,I2,15H      ---------)
 5021 FORMAT(/,14H TSP,SPM,B(3)=,/,3(1PE13.5))
      RETURN
      END
```

```
      SUBROUTINE REFAZE
C     UPDATING THE DATA AT PHASE CHANGE.
C
      INTEGER RKSTEP,ORDER
      REAL MASS,MPROP,MPL,MDOT,MACH,LGG,LCP,IYY,MSEP
      DOUBLE PRECISION DPS,DPC
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTA1,TBO,RKSTEP,EPBIG,EPTINY,
     ;               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR08/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),QP(6,20)
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,CA,AD,QS,
     ;               SLOPE,CMP,LCG,LCP,IYY,SY,SZ
      COMMON/STOR11/ KPREV,HPREV,MPREV,APREV,LPREV(14),VPREV(14),
     ;               ORDER(14)
      COMMON/STOR12/ NCMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSG,NCON
      COMMON/STOR13/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPUNCH,INDEX
      COMMON NRST,NSEP,NSP,NM(14),KSP(20),RMASS,R3D,TAV(3),TANG(61),B(4),
     ;       PLM(20),PHT(32),TBEP(32),MSEP(32),TSP(20),SPM(20),A(99,14)
C
      NR = NP + 1
      NP1 = NP + 1
      TNP1 = PHT(NR1)
      READ(ITAPE3) (B(I),I=1,4)
      B(3)=B(3)/2.0
      B(4) = B(4)**2
      DO 10 J=1,19
      DO 1 I=1,NVAR
      A(I,J) = 0.
    1 CONTINUE
      READ(ITAPE3) MAX,(A(I,J),I=1,MAX)
      NM(J) = MAX
      LPREV(J) = 1
      VPREV(J) = A(1,J)
   10 CONTINUE
      MDOT = 0.
      SUBM(NP) = MASS - A(1,3)
      S(7) = A(1,8)*SY
      S(8) = A(1,8)*SZ
      DPS(7)=S(7)
      DPS(8)=S(8)
C
      JS1 = JS+1
      IF(JS1.GT.NCMAX+1) RETURN
```

ORIGINAL PAGE IS OF POOR QUALITY

```
      IF(ABS(T-TSEP(JS1)).GT.1.0E-05) GO TO 100
      JS = JS + 1
      SUBM(NP) = SUBM(NP) + MSEP(JS1)
  100 DO 120 J=1,NP
      IF(ABS(T-TSP(J)) .GT. 1.0E-05) GO TO 120
      JSC = JSC+1
      DO 110 I=1,6
      SR(I,JSC) = S(I)
  110 CONTINUE
  120 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE RK4(NF,SOLV)
C     INTEGRATION BY RUNGE-KUTTA-SIMPSON METHOD OF FOURTH ORDER,
C     RALSTON COEFFICIENTS ARE USED FOR MINIMUM ERROR BOUNDS.
C
      INTEGER RKSTEP
      DOUBLE PRECISION DRS,DRC,DRF
      DOUBLE PRECISION G1(14),G2(14),G3(14),G4(14)
      COMMON/STOR01/ T,DT,TNP1,DTLA,DTA1,TBO,RKSTEP,EPBIG,EPTINY,
     ;               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR02/ S(14),F(14,5),DRS(14),DRC(14),ERW(14),SP(6,20)
      DATA A1/ 0.4/
      DATA B1,B2,B3/ 0.45973726, 0.29697760, 0.15875966/
      DATA C1,C2,C3/ 0.21810038, 3.05096470, 3.83286438/
      DATA D1,D2,D3,D4/ 0.17476028, 0.55148053, 1.20553547, 0.17118478/
C
      CALL SOLV(1)
      DO 150 J=1,RKSTEP
      TT = T
      DO 110 I=1,NF
      G1(I) = DT*F(I,J)
      S(I) = DRS(I) + B1*G1(I)
  110 CONTINUE
      T = TT + A1*DT
      CALL SOLV(5)
      DO 120 I=1,NF
      G2(I) = DT*F(I,5)
      S(I) = DRS(I) + B2*G1(I) + B3*G2(I)
  120 CONTINUE
      T = TT + B1*DT
      CALL SOLV(5)
      DO 130 I=1,NF
      G3(I) = DT*F(I,5)
      S(I) = DRS(I) + C1*G1(I) - C2*G2(I) + C3*G3(I)
  130 CONTINUE
      T = TT + DT
      CALL SOLV(5)
      DO 140 I=1,NF
      G4(I) = DT*F(I,5)
      S(I) = DRS(I) + D1*G1(I) - D2*G2(I) + D3*G3(I) + D4*G4(I)
      DRS(I) = S(I)
      DRC(I) = 0.
  140 CONTINUE
```

```
      IF(RKSTEP.EQ.1) GO TO 150
      JJ = J + 1
      CALL SOLV(JJ)
  150 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ROCKET(KODE)
C     INTERPOLATES BETWEEN THRUST AND MACH TABLES.
C
      LOGICAL KODE
      INTEGER RKSTEP
      REAL MASS,MPROP,MPL,MDOT,MACH,LCG,LCP,IYY,MSEP
      COMMON/STOR02/ PRESO,DENSO,TEMPO,SOUNDO,PRESB(9),TEMPB(9),
     ,               HALTB(9),TSLOPE(9),RONST,PRES,DENS,TEMP,SOUND
      COMMON/STOR05/ VT(3),VA(3),VEL,ACC,VOLIM,VLIM
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     ,               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,QA,AD,QS,
     ,               SLOPE,CMP,LCG,LCP,IYY,SY,SZ
      COMMON NPST,NSEP,NSP,NM(14),KSP(20),RMASS,T3D,TAV(3),ANG(61),B(4),
     ,       PLM(20),PHT(32),TSER(32),MSEP(32),TSP(20),SPM(28),A(99,14)
C
C
C     INTERPOLATION BETWEEN MACH NO TABLES.
      CA = 0.
      IF(SOUND .EQ. 0.) GO TO 121
      MACH = VEL/SOUND
      CALL RTERP(MACH,J,*,*110,*120)
      J1 = J + 1
      FRAC = (MACH - A(J,4))/(A(J1,4) - A(J,4))
      CA = A(J,5) + FRAC*(A(J1,5)-A(J,5))
      IF(KODE) GO TO 150
      CMP = A(J,10) + FRAC*(A(J1,10)-A(J,10))
      SLOPE = A(J,12) + FRAC*(A(J1,12)-A(J,12))
      LCP = A(J,14) + FRAC*(A(J1,14)-A(J,14))
      GO TO 150
  110 CA = A(1,5)
      IF(KODE) GO TO 150
      CMP = A(1,10)
      SLOPE = A(1,12)
      LCP = A(1,14)
      GO TO 150
  120 CA = A(NM(5),5)
  121 IF(KODE) GO TO 150
      CMP = A(NM(10),10)
      SLOPE = A(NM(12),12)
      LCP = A(NM(14),14)
C
```

ORIGINAL PAGE IS OF POOR QUALITY

```
C     INTERPOLATION BETWEEN TIME TABLES.
  150 IF(ABS(A(1,1)-A(2,1))) 200,220,200
  200 CALL RTERP(Y,J,1,$210,$220)
      J1 = J + 1
      DELT = 1.0/(A(J1,1)-A(J,1))
      DELM = A(J1,3)-A(J,3)
      FRAC = (Y-A(J,1))*DELT
      THR=A(J,2)+FRAC*(A(J1,2)-A(J,2))-B(1)*PRES
      IF(THR.LT.0.) THR=0.
      MPROP = A(J,3) + FRAC*DELM
      IF(KODE) RETURN
      MDOT = DELM*DELT
      LOG = A(J,7) + FRAC*(A(J1,7)-A(J,7))
      IYY = A(J,8) + FRAC*(A(J1,8)-A(J,8))
      RETURN
  210 THR = A(1,2)
      MPROP = A(1,3)
      IF(KODE) RETURN
      MDOT = 0.
      LOG = A(1,7)
      IYY = A(1,8)
      RETURN
  220 THR = 0.
      MPROP = 0.
      IF(KODE) RETURN
      MDOT = 0.
      LOG = A(NM(7),7)
      IYY = A(NM(8),8)
      RETURN
      END
*****W  1470 EQUALITY OR NON-EQUALITY COMPARISON MAY NOT BE MEANINGFUL IN LOGICAL IF EXPRESSIONS
```

```
      SUBROUTINE RTERP(V,L,I,***)
C     INTERPOLATION PROGRAM CALLED BY ROCKET.
C
      INTEGER ORDER
      REAL MSEP
      COMMON/STOR11/ KPREV,HPREV,MPREV,APREV,LPREV(14),VPREV(14),
     1               ORDER(14)
      COMMON NPST,NSEP,NSP,NH(14),MSP(20),RMASS,T3D,TAV(3),TANG(61),B(4),
     1       PLM(20),PHT(32),TSEP(32),MSEP(32),TSP(20),SPM(20),A(99,14)
C
      L = LPREV(I)
      MAX = NM(I)
      IF(V.GT.A(L,I) .AND. V.LE.A(L+1,I)) GO TO 200
      IF(V.LE.A(1,I)) RETURN 1
      IF(V.GE.A(MAX,I)) RETURN 2
      IF(V = VPREV(I)) 100,200,110
  100 JB = 2
      JE = L + 1
      GO TO 120
  110 JB = L + 1
      JE = MAX
  120 DO 130 J=JB,JE
      JSAVE = J
      IF(V . LT. A(J,I)) GO TO 140
  130 CONTINUE
  140 L = JSAVE - 1
  200 LPREV(I) = L
      VPREV(I) = V
      RETURN
      END
```

```
      SUBROUTINE RTIMAT
C     ROTATION MATRIX FROM TOPOCENTRIC(GEO-CEN) TO INERTIAL,
C
      DOUBLE PRECISION DRS,DPC,X,Y,Z,Q
      COMMON/STOR01/ G0,GR,OMEGAT,GH,RB,RQ,ESQ,ESQI,ESQ1
      COMMON/STOR05/ X0,Y0,Z0,R,RXY,ALT,SR,HLIA
      COMMON/STOR06/ RTI(9)
      COMMON/STOR08/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),SP(6,20)
C
      X = S(1)
      Y = S(2)
      Z = S(3)
      Q = X*X + Y*Y
      R = DSQRT(Q + Z*Z)
      RXY = DSQRT(Q)
      ALT = R - RB
      RI = 1.0D0/R
      RXYI = 1.0D0/RXY
      RTI(4) = -S(2)*RXYI
      RTI(5) = S(1)*RXYI
      RTI(6) = 0.
      RTI(7) = -S(1)*RI
      RTI(8) = -S(2)*RI
      RTI(9) = -S(3)*RI
      RTI(1) = RTI(9)*RTI(5)
      RTI(2) = -RTI(9)*RTI(4)
      RTI(3) = RXY*RI
      RETURN
      END
```

```
      SUBROUTINE SETIV
C     INITIALIZES ALL THE NECESSARY DATA BEFORE STARTING PHASE NO 1.
C     THIS ROUTINE INITIALIZES STEP DT BY INTEGRATING FIRST BY RK4 WITH
C     DT=DTLA ONE STEP AND THEN WITH DT=DTLA/2 TWO STEPS, AND
C     COMPARING AT T=T+DT INTEGRATION VARIABLES OBTAINED DIFFERENTLY,
C     IF ERROR LIES BEYOND THE ERROR BOUNDS A NEW STEP DT IS ITERATED;
C     NCON = -1 FIXES LAUNCH AND INERTIAL ORIGINS,
C          =  0 TAKES LAUNCH IN GD FRAME;
C          =  1 TAKES LAUNCH IN GC FRAME;
C          =  2 ONLY ASSUMES THAT WIND IS PRESENT.
C
      INTEGER RKSTEP,ORDER
      REAL LATL,LONGL,LAT,LONG,MASS,MPROP,MPL,HDOT,MACH,LCG,LCP,IYY,MSEP
      DOUBLE PRECISION DRS,DPC,AR1,AR2
      EXTERNAL SOLV5D
      DIMENSION SAVE(14)
      COMMON/STOR00/ ZERO,ONE,PI,QPI,HPI,TPI,CF(9)
      COMMON/STOR01/ G0,GR,OMEGA,GM,RE,RG,ESQ,BSQ1,ESQ1
      COMMON/STOR02/ PBES0,DENS0,TEMP0,SOUND0,PRESB(9),TEMPB(9),
     ,               HALTB(9),TSLOPE(9),CONST,PRES,DENS,TEMP,SOUND
      COMMON/STOR03/ X0,Y0,Z0,R,RXY,ALT,SR,HLIM
      COMMON/STOR04/ GDLATL,LATL,LONGL,THETAL,PHIL,GDLAT,LAT,LONG,THETA,
     ,               PHI,GAMAL,PSIL,GAMA,PSI,ELGDL,AZGDL,ELGD,AZGD,EL,
     ,               AZ,SRAZ
      COMMON/STOR05/ VT(3),VA(3),VEL,ACC,VGLIM,VLIM
      COMMON/STOR06/ RTI(9)
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTA1,TBO,RKSTEP,EPBIG,EPTINY,
     ,               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR08/ S(14),F(14,5),DRS(14),DPC(14),ERW(14),SP(6,20)
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,CA,AD,QS,
     ,               SLOPE,CMP,LCG,LCP,IYY,SY,SZ
      COMMON/STOR10/ WE,WN,MLAST,WIND,WNAZ,ALOW,AHIGH,NLEV,ALEV(100),
     ,               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR11/ KPREV,HPREV,MPREV,APREV,LPREV(14),VPREV(14),
     ,               ORDER(14)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NR,NP1,JSPENT,JS,JSQ,NCON
      COMMON/STOR14/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPUNCH,INDEX
      COMMON NPST,NSEP,NSP,NM(14),KSP(20),RMASS,T3D,TAV(3),ANG(61),B(4),
     ,       PLM(20),PHT(32),TSEP(32),MSEP(32),TSP(20),SPM(20),A(99,14)
C
      IF(NCON) 10,20,30
C
```

```
   10 CONTINUE
C     FIX LAUNCH AND INERTIAL ORIGINS.
C     ENTER HERE IF LAT AND LONG ARE CHANGED.
      AR1 = GDLATL
      AR2 = ESQ1*(DSIN(AR1)/DCOS(AR1))
      AR1 = DATAN(AR2)
      LATL = AR1
      THETAL = -LATL + HRI
      SM = DSIN(AR1)
      CM = DCOS(AR1)
      PHIL=LONGL
      IF(LONGL.LT.0.) PHIL=TPI+LONGL
      AR1 = LONGL
      SL = DSIN(AR1)
      CL = DCOS(AR1)
      EPS = GDLATL-LATL
      AR1 = EPS
      SEPS = DSIN(AR1)
      CEPS = DCOS(AR1)
      RE = RQ*SQRT(ESQ1/(1.0-ESQ*CM*CM))
      X0 = RE*CM*CL
      Y0 = RE*CM*SL
      Z0 = RE*SM
      IF(IPRINT.NE.2) RETURN
      PRINT 5000, NCON
      PRINT 5001, GDLATL,LATL,THETAL,EPS
      PRINT 5002, LONGL,PHIL
      PRINT 5003, RE,X0,Y0,Z0
      RETURN
C
   20 CONTINUE
C     INPUT LAUNCH ANGLES ARE IN GEO-DETIC FRAME.
C     ENTER HERE IF LAUNCH ANGLES ARE CHANGED.
      AR1 = ELGDL
      SEL = DSIN(AR1)
      CEL = DCOS(AR1)
      AR1 = AZGDL
      SAZ = DSIN(AR1)
      CAZ = DCOS(AR1)
      AR1=SEL*CEPS-SEPS*CEL*CAZ
      AR2=DSQRT(1.0D0-AR1*AR1)
      GAMA = DATAN2(AR1/AR2)
```

```
      AR1=CEL*SAZ
      AR2=CEPS*CEL*CAZ+SEPS*SEL
      PSI = DATAN2(AR1,AR2)
      IF(PSI.LT.0.) PSI=PSI+TPI
      EL = GAMA
      AZ = PSI
      ELGD = ELGDL
      AZGD = AZGDL
C
   30 CONTINUE
C     ENTER HERE IF LAT, LONG, AND LAUNCH ANGLES ARE NOT CHANGED.
      EL = GAMA
      AZ = PSI
      AR1 = GAMA
      ST = DSIN(AR1)
      CT = DCOS(AR1)
      AR1 = PSI
      SU = DSIN(AR1)
      CU = DCOS(AR1)
      AR1 = EL
      SEL = DSIN(AR1)
      CEL = DCOS(AR1)
      AR1 = AZ
      SAZ = DSIN(AR1)
      CAZ = DCOS(AR1)
      AR1 = SEPS*CT*CU + CEPS*ST
      AR2 = 1.0D0 - AR1**2
      ELGD = HPI
      IF(AR2 .LT. 1.0D-15) GO TO 31
      AR2 = DSQRT(AR2)
      ELGD = DATAN(AR1/AR2)
   31 AR1 = CT*SU
      AR2=CEPS*CT*CU-SEPS*ST
      AZGD=DATAN2(AR1,AR2)
      IF(AZGD.LT.0.) AZGD=AZGD+TPI
C
C
C     INITIALIZE THE INTEGRATION VARIABLES AND PARAMETERS.
C     ITERATE FOR GOOD INITIAL STEP DT.
      DT = DTLA
      RKSTEP = 1
  110 IF(RKSTEP .NE. 2) GO TO 120
```

ORIGINAL PAGE IS OF POOR QUALITY

```
      DT = DT/2.0
C
  120 NR = 1
      NR1 = NP + 1
      T = TAV(1)
      TNP1 = PHT(NR1)
      TBO = T3D
      ALT = TAV(2)
      R = RE + ALT
      VEL=TAV(3)  ;  IF(VEL.LT.VOLIM) VEL=VOLIM
      VSQ = VEL**2
      WN=0.  ;  WE=0.
      REWIND ITAPE3
      READ(ITAPE3) (B(I),I=1,4)
      B(3)=B(3)/2.0  ;  B(4)=B(4)**2
      DO 125 J=1,18
      DO 124 I=1,NVAR
      A(I,J) = 0.
  124 CONTINUE
      READ(ITAPE3) MAX,(X(I,J),I=1,MAX)
      NM(J) = MAX
      LRREV(J) = 1
      VPREV(J) = A(1,J)
  125 CONTINUE
      MDOT = 0.
      SUBM(NP) = BMASS + MPL + A(1,3)
      MRREV = 1
      HPREV = 0.
      KPREV = 1
      APREV = 0.
      KSTEP=10  ;  NGOOD=1
      JS=0  ;  JSQ=0
      INDEX = 0
      S(1) = R*CM*CL
      S(2) = R*CM*SL
      S(3) = R*SM
      V1=VEL*CEL*CAZ
      V2=VEL*CEL*SAZ
      V3=-VEL*SEL
      CALL RTIMAT
      S(4) = RTI(1)*V1 + RTI(4)*V2 + RTI(7)*V3 - OMEGA*S(2)
      S(5) = RTI(2)*V1 + RTI(5)*V2 + RTI(8)*V3 + OMEGA*S(1)
```

```
      S(6) = RTI(3)*V1 + RTI(6)*V2 + RTI(9)*V3
      S(7) = 0.
      S(8) = 0.
      S(09) = -CL*CU*CT+SM - CT*SL*SU + CM*CL*ST
      S(10) = -CU*CT*SL*SM + CL*CT*SU + CM*SL*ST
      S(11) = CM*CU*CT + SM*ST
      S(12) = CL*SM*SU - CU*SL
      S(13) = SL*SM*SU + CL*CU
      S(14) = -CM*SU
      DO 130 I=1,14
      DRS(I) = S(I)
  130 CONTINUE
C
      IF(RKSTEP .EQ. 3) GO TO 900
      CALL RK4(14,SOLVSD)
      IF(RKSTEP .EQ. 2) GO TO 150
      DO 140 I=1,14
      SAVE(I) = S(I)
  140 CONTINUE
      RKSTEP = 2
      GO TO 110
  150 EMAX = 0.
      DO 160 I=1,14
      ERR = 0.
      IF(ABS(S(I)).GT.1.0E-08) ERR=ABS(1.0-SAVE(I)/S(I))
      EMAX = AMAX1(EMAX,ERR)
  160 CONTINUE
      EMAX = EMAX/15.0
      IF(IPRINT.EQ.2) PRINT 5005, DT*2,EMAX,SAVE,S
      IF(EMAX - 1.0E-05) 170,170,180
  170 RKSTEP = 3
      GO TO 110
  180 IF(DT - DMIN) 170,170,190
  190 RKSTEP = 1
      GO TO 110
C
  900 IF(IPRINT.NE.2) RETURN
      PRINT 5010, NCON
      PRINT 5011, ELODL,GAMA,EL
      PRINT 5012, AZODL,PSI,AZ
      PRINT 5013, T,DT,NP,RKSTEP
      PRINT 5014, S
```

```
C
 5000 FORMAT(1H1,45HTHE ORIGIN AT THE LAUNCH SITE IS FIXED; NCON=,I2)
 5001 FORMAT(/,25H GDLATL,LATL,THETAL,EPS =,4(1PE13.5))
 5002 FORMAT(/,13H LONGL,PHIL =,2(1PE13.5))
 5003 FORMAT(/,14H RE,X0,Y0,Z0 =,4(1PE13.5))
 5005 FORMAT(/,' ITERATION FOR GOOD STEP SIZE -- DT,EMAX=',2E20.10,/,
     ,' S WITH DT ONE STEP AND S WITH DT/2 TWO STEPS',/,
     ,(10E13.5,/,4E13.5))
 5010 FORMAT(///,70H THE VEHICLE ATTITUDE AND INTEGRATION VARIABLES ARE
     ,INITIALIZED, NCON=,I2)
 5011 FORMAT(/,16H ELGDL,GAMA,EL =,3(1PE13.5))
 5012 FORMAT(/,15H A2GDL,PSI,AZ =,3(1PE13.5))
 5013 FORMAT(/,17H T,DT,NPARKSTEP =,1PE13.5,,1PE13.5,,2I5)
 5014 FORMAT(/,13H S(1,...,14) =,/,(10(1PE13.5)))
      RETURN
      END
```

```
      SUBROUTINE SOLV3D(K)
C     EQUATIONS OF MOTIONS IN 3 DIMENSIONS.
C
      REAL MASS,MPROP,MPL,MDOT,MACH,LCG,LCP,IYY,MSEP
      DOUBLE PRECISION DPS,DPC
      COMMON/STOR01/ G0,GR,OMEGA,GM,RE,RO,ESQ,ESQI,ESQ1
      COMMON/STOR02/ PBESO,DENSO,TEMPO,SOUNDO,PRESB(9),TEMPB(9),
     ,               HALTB(9),TSLOPE(9),CONST,PRES,DENS,TEMP,SOUND
      COMMON/STOR03/ XO,YO,ZO,R,RXY,ALT,SRHLIM
      COMMON/STOR05/ VT(3),VA(3),VEL,ACC,VOLIM,VLIM
      COMMON/STOR06/ RTI(9)
      COMMON/STOR08/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),BP(6,20)
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,CA,AD,QS,
     ,               SLOPE,CMP,LCG,LCP,IYY,SY,SZ
      COMMON/STOR10/ WE,WN,WLAST,WIND,WNAZ,ALOW,AHIGH,NLEV,ALEV(100),
     ,               ALTW(100),VELB(100),VELN(100)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSM,NCON
      COMMON NPST,NSEP,NSP,NM(14),KSP(20),RMASS,T3D,TAV(3),ANG(61),B(4),
     ,       PLM(20),PHT(32),TSEP(32),MSEP(32),TSP(20),SPM(20),A(99,14)
C
      CALL RTIMAT
      CALL ATMSPH
      IF(NCON.EQ.2) CALL WTERP
      GR = GM/(R*R)
      S1 = S(4) + OMEGA*S(2)
      S2 = S(5) - OMEGA*S(1)
      VT(1) = RTI(1)*S1 + RTI(2)*S2 + RTI(3)*S(6) - WN
      VT(2) = RTI(4)*S1 + RTI(5)*S2 + RTI(6)*S(6) - WE
      VT(3) = RTI(7)*S1 + RTI(8)*S2 + RTI(9)*S(6)
      VSQ = VT(1)**2 + VT(2)**2 + VT(3)**2
      VEL = SQRT(VSQ)
      QS = DENS*VSQ*B(3)
      CALL ROCKET(.TRUE.)
      MASS=SUBM(NP)+MPROP
      WTI = 1.0/MASS
      AD = CA*QS
      AF = THR - AD
      S1 = AF/VEL
      AFC1 = S1*VT(1)
      AFC2 = S1*VT(2)
      AFC3 = S1*VT(3)
      F(1,K) = S(4)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      F(2,K) = S(5)
      F(3,K) = S(8)
      F(4,K) = GR*RTI(7) + WTI*(RTI(1)*AFC1+RTI(4)*AFC2+RTI(7)*AFC3)
      F(5,K) = GR*RTI(8) + WTI*(RTI(2)*AFC1+RTI(5)*AFC2+RTI(8)*AFC3)
      F(6,K) = GR*RTI(9) + WTI*(RTI(3)*AFC1+RTI(6)*AFC2+RTI(9)*AFC3)
      RETURN
      END
```

```
      SUBROUTINE SOLV5D(K)
C     EQUATIONS OF MOTIONS IN 5 DIMENSIONS.
C
      REAL MASS,MPROP,MPL,MDOT,MACH,LCG,LCP,IYY,MSEP,NA,NB,LSM,LEXCG,
     1     JDCON
      DOUBLE PRECISION DPS,DPC,AR1,AR2,V1,V2,V3
      COMMON/STOR01/ G0,GR,OMEGA,GM,RE,RO,ESQ,ESQI,ESQ1
      COMMON/STOR02/ PRESO,DENSO,TEMPO,SOUNDO,PRESB(9),TEMPB(9),
     1               HALTB(9),TSLOPE(9),DONST,PRES,DENS,TEMP,SOUND
      COMMON/STOR03/ X0,Y0,Z0,R,RXY,ALT,SR,HLIM
      COMMON/STOR05/ VT(3),VA(3),VEL,ACC,VOLIM,VLIM
      COMMON/STOR06/ RTI(9)
      COMMON/STOR08/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),SP(6,20)
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,HR,MACH,CA,AD,QS,
     1               SLOPE,CMP,LCG,LCP,IYY,SY,SZ
      COMMON/STOR10/ WE,WN,MLAST,WIND,WNAZ,ALOW,AHIGH,NLEV,ALEV(100),
     1               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSC,NCON
      COMMON NPST,NSEP,NSP,NM(14),KSP(20),RMASS,T3D,TAV(3),ANG(61),B(4),
     1       PLM(20),PHT(32),TSEP(32),MSEP(32),TSP(20),SPM(20),A(99,14)
      EQUIVALENCE (S(9),RM11),(S(10),RM21),(S(11),RM31),(S(12),RM12),
     1            (S(13),RM22),(S(14),RM32)
C
      RM13 = RM21*RM32 - RM31*RM22
      RM23 = RM31*RM12 - RM11*RM32
      RM33 = RM11*RM22 - RM21*RM12
      CALL RTIMAT
      CALL ATMSPH
      IF(NCON.EQ.2) CALL WTERP
      GR = GM/(R*R)
      WEC = WE + OMEGA*RXY
      V1 = S(4) - RTI(1)*WN - RTI(4)*WEC
      V2 = S(5) - RTI(2)*WN - RTI(5)*WEC
      V3 = S(6) - RTI(3)*WN - RTI(6)*WEC
      VA(1) = RM11*V1 + RM21*V2 + RM31*V3
      VA(2) = RM12*V1 + RM22*V2 + RM32*V3
      VA(3) = RM13*V1 + RM23*V2 + RM33*V3
      VT(3) = RTI(7)*V1 + RTI(8)*V2 + RTI(9)*V3
      VSQ=VA(1)**2+VA(2)**2+VA(3)**2
      VEL=SQRT(VSQ)
      QS=DENS*VSQ*B(3)
      CALL ROCKET(.FALSE.)
```

```
      MASS=SUBM(NP)+MPBOP
      WTI=1.0/MASS
      AD=GA*QS
      AR = THR / AD
      AR1 = V(3)/V(1)
      ALFA = DATAN(AR1)
      AR1 = V(2)/V(1)
      BETA = DATAN(AR1)
      NA=SLOPE*ALFA*QS
      NB=SLOPE*BETA*QS
      LEXOG=ABS(B(2)-LOG)
      JDCON = MDOT*LEXOG
      PDCOEF = 0.5*CMP*QS*B(4)/VEL + JDCON*LEXOG
      JDCON = JDCON*2.0
      TERM = 1.0/IY
      SY = S(7)*TERM
      SZ = S(8)*TERM
      AG = NB+JDCON*SZ
      AH = NA+JDCON*SY
      LSM=ABS(LCG-LCP)
      F( 1,K) = S(4)
      F( 2,K) = S(5)
      F( 3,K) = S(6)
      F( 4,K) = GB*RY(7) + WTI*(RM11*AF+RM12*AG-RM13*AH)
      F( 5,K) = GB*RY(8) + WTI*(RM21*AF+RM22*AG-RM23*AH)
      F( 6,K) = GB*RY(9) + WTI*(RM31*AF+RM32*AG-RM33*AH)
      F( 7,K) = -NA*LSM + PDCOEF*SY
      F( 8,K) =  NB*LSM + PDCOEF*SZ
      F( 9,K) = RM12*SZ - RM13*SY
      F(10,K) = RM22*SZ - RM23*SY
      F(11,K) = RM32*SZ - RM33*SY
      F(12,K) = -RM11*SZ
      F(13,K) = -RM21*SZ
      F(14,K) = -RM31*SZ
      RETURN
      END
```

```
      SUBROUTINE SPENT(IROT)
C     CALCULATION OF TRAJECTORY FOR THE SPENT STAGES.
C
      INTEGER RKSTEP,ORDER
      REAL LATL,LONGL,LAT,LONG,MASS,MPROP,MPL,MDOT,MACH,LCG,LCP,IYY,IPT,
     ,     IPAZ,IPR,MSEP
      DOUBLE PRECISION DRS,DPC,AR1,AR2
      CHARACTER NAME*80
      EXTERNAL SOLV3D,SOLV5D
      DIMENSION SAVE(20,11)
      COMMON/STOR00/ ZERO,ONE,PI,QPI,HPI,TPI,CF(9)
      COMMON/STOR01/ G0,GR,OMEGA,GM,RE,RO,ESO,ESOI,ESQ1
      COMMON/STOR02/ PRESO,DENSO,TEMPO,SOUNDO,PRES(9),TEMPB(9),
     ,               HALTB(9),TSLOPE(9),DONST,PRES,DENS,TEMP,SOUND
      COMMON/STOR03/ X0,Y0,Z0,R,RXY,ALT,SR,HLIM
      COMMON/STOR04/ GDLATL,LATL,LONGL,THETAL,PHIL,GDLAT,LAT,LONG,THETA,
     ,               PHI,GAMAL,PSIL,GAMA,PSI,ELGDL,AZGDL,ELGD,AZGD,EL,
     ,               AZ,SRAZ
      COMMON/STOR05/ VT(3),VA(3),VEL,ACC,VOLIM,VLIM
      COMMON/STOR06/ RTI(9)
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     ,               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR08/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),SP(6,20)
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,CA,AD,QS,
     ,               SLORE,CMP,LCG,LCP,IYY,SY,SZ
      COMMON/STOR10/ WE,WN,MLAST,WIND,WNAZ,ALOW,AHIGH,NLEV,ALEV(100),
     ,               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR11/ KPREV,HPREV,MPREV,APREV,LPREV(14),VPREV(14),
     ,               ORDER(14)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSC,NCON
      COMMON/STOR13/ APT,APAZ,APA,APR,BOT,BOA,BOR,BOVL,BOEL,BOAZ,IPT,
     ,               IPAZ,IPR
      COMMON/STOR14/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPUNCH,INDEX
      COMMON/STOR99/ NAME,ARRAY(800)
      COMMON NPST,NSEP,NSP,NM(14),KSP(20),RMASS,T3D,TAV(3),ANG(61),B(4),
     ,       PLM(20),PHT(32),TSEP(32),MSEP(32),TSP(20),SPM(20),A(99,14)
      EQUIVALENCE (ARRAY(1),SAVE(1,1))
C
      SAVEO = OMEGA
      IF(IROT.EQ.0) OMEGA=1.0E-10
      REWIND ITAPE4
      DO 700 IP=1,NPL
```

ORIGINAL PAGE IS OF POOR QUALITY

```
      MBL = PLM(IP)
      PRINT 1000
      PRINT 1010, NAME,MBL,GO,AZGDL,CF(1)
      IF(NLAST.EQ.0) PRINT 1011
      IF(NLAST.EQ.0) PRINT 1012, WIND,WNAZ,CF(1),ALOW,AHIGH
      IF(NLEV.GT.0) PRINT 1013
      IF(IROT.EQ.0) PRINT 1014
      IF(IROT.NE.0) PRINT 1015
      IF(NSR.NE.0) GO TO 16
      PRINT 1016
      RETURN
   10 IF(IPRINT.EQ.0) PRINT 1020
      DO 600 IA=1,NANG
      ELGDL = ANG(IA)
    6
      DO 500 K=1,NWP
      B(1) = 0.
      B(2) = 0.
      B(4) = 0.
      READ(ITAPE4) TBP(K),SPM(K),B(3)
      B(3) = B(3)/2.0
      DO 30 J=1,14
      MAX = 2
      DO 20 I=1,NVAR
      A(I,J) =0.
   20 CONTINUE
      IF(J.LT.4 .OR. J.GT.9) GO TO 21
      READ(ITAPE4) MAX,(A(I,J),I=1,MAX)
   21 NN(J) = MAX
      LAREV(J) = 1
      VAREV(J) = A(1,J)
   30 CONTINUE
      READ(ITAPE4) (S(I),I=1,6)
      DO 40 I=1,6
      DAS(I) = S(I)
   40 CONTINUE
      NP = NPMAX+1
      NP1 = NP + 1
      JS = NP
      JSC = NSMAX
      T = TBP(K)
      TBPS = PHT(NP1)
```

```
      DT = DTLA
      SUBM(NP) = SPM(K)
      KPREV = 1
      HPREV = HALTB(1)
      MPREV = 1
      APREV = ZERO
      RKSTEP = 3
      KSTEP = 10
      NGOOD = 1
      CALL RTIMAT
      S1 = RE/R
      CALL AZRAN(S1*S(1),S1*S(2),S1*S(3),T)
      WE = 0.
      WN = 0.
      IF(NCON.EQ.2) CALL WTERP
      S1 = S(4) + OMEGA*S(2)
      S2 = S(5) - OMEGA*S(1)
      VT(1) = RTI(1)*S1 + RTI(2)*S2 + RTI(3)*S(6)
      VT(2) = RTI(4)*S1 + RTI(5)*S2 + RTI(6)*S(6)
      VT(3) = RTI(7)*S1 + RTI(8)*S2 + RTI(9)*S(6)
      VSQ = VT(1)**2 + VT(2)**2 + VT(3)**2
      VEL = SQRT(VSQ)
      AR1 = -VT(3)
      AR2= SQRT(VT(1)**2+VT(2)**2)
      ELV = DATAN(AR1/AR2)
      AR1 = VT(2)
      AR2 = VT(1)
      AZM = DATAN2(AR1,AR2)
      IF(AZM.LT.0.) AZM=AZM+TPI
      SAVE(K,1) = TSP(K)
      SAVE(K,2) = SPM(K)*GO
      SAVE(K,3) = ALT
      SAVE(K,4) = SR*CF(4)
      SAVE(K,5) = SRAZ*CF(1)
      SAVE(K,6) = VEL
      SAVE(K,7) = ELV*CF(1)
      SAVE(K,8) = AZM*CF(1)
C
      INDEX = 0
      IF(IPRINT.NE.0) PRINT 1017,  K,ELGDL*CF(1)
      CALL RK4(6,SOLV3D)
   50 CALL PCM(6,SOLV3D)
```

```
      CALL IPSTEP($50,$60)
   60 CALL RK4(6,$OLV3D)
      IF(ALT.LT.HLIM) GO TO 70
      CALL IPSTEP($60,$60)
   70 CALL IRCAL
      SAVE(K,9) = T
      SAVE(K,10)= $R*CF(X)
      SAVE(K,11)= $RAZ*CF(1)
  500 CONTINUE
      IF(IPRINT.EQ.0) GO TO 505
      PRINT 1001
      PRINT 1020
  505 DO 510 K=1,NSP
      PRINT 1030, ELADL*CF(1),K,(SAVE(K,I),I=1,11)
  510 CONTINUE
  800 CONTINUE
  900 CONTINUE
      OMEGA = SAVE0
C
 1000 FORMAT(1H1,26HNASA WALLOPS FLIGHT CENTER,/,25H WALLOPS ISLAND, VIR
     ,GINIA,//37H TRAJECTORY SUMMARY FOR SPENT STAGES,//)
 1001 FORMAT(1H1)
 1010 FORMAT(13H VEHICLE    = ,A80,/,18H PAY LOAD  = ,F9.2,21H LBS (PAREN
     ,T VEHICLE),/,13H LAUNCH AZ = ,F7.2,9H DEG (,F)7/)
 1011 FORMAT(13H WIND       = ,4HZERO)
 1012 FORMAT(13H WIND       = ,F7.2,8H FT/SEC,,F7.2,13H DEG NORTH AZ,/,
     ,13X,5H FROM,F9.1,6H FT TO,F9.1,8H FT)
 1013 FORMAT(13H WIND       = ,30HVARIABLE WIND SUPPLIED BY USER)
 1014 FORMAT(13H EARTH      = ,18HNON-ROTATING MODEL)
 1015 FORMAT(13H EARTH      = ,14HROTATING MODEL)
 1016 FORMAT(///,19H MISSING INPUT DATA,//,41H DRAG TABLE NOT SUPPLIED F
     ,OR SPENT STAGE)
 1017 FORMAT(1H1,39HDETAILED PRINT-OUT OF SPENT STAGE ,I2,11H TRAJECTOR
     ,Y,/,37H WHERE PARENT VEHICLE LAUNCH EL WAS ,F7.2,4H DEG,/,
     ,54H UNITS ARE FPS-S-DEGREE) EXCEPT RANGE(NM) AND ACC(G),,//)
 1020 FORMAT(43X,28HS  E  P  A  R  A  T  I  O  N,40X,16HI  N  P  A  C  T
     ,,/,80X,77(1H*),10X,23(1H*),/,7H LAUNCH,4X,3HNO.,6X,4HTIME,5X,6HWEI
     ,GHT,6X,3HALT,7X,5HRANGE,6X,2HAZ,8X,3HVEL,6X,5HFLT/EL,4X,6HFLT/AZ,
     ,10X,4HTIME,5X,5HRANGE,6X,2HAZ,/,8H EL(DEG),12X,5H(SEC),4X,6H(SLUG)
     ,,6X,4H(FT),6X,4H(NM),6X,5H(DEG),3X,8H(FT/SEC),5X,5H(DEG),5X,5H(DEG
     ,),10X,5H(SEC),4X,4H(NM),6X,5H(DEG),/,/)
 1030 FORMAT(F8.2,I5,F11.2,F10.2,F10.0,F11.2,F10.2,F12.2,2F10.2,F15.2,
```

```
169          ,F9.2,F10.2)
170           RETURN
171           END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE SPLASH
C     CALCULATES TRAJECTORIES FOR WIND EFFECTS.
C
      INTEGER RKSTEP
      EXTERNAL SOLV3D,SOLV5D
      COMMON/STOR03/ X0,Y0,Z0,R,BXY,ALT,SR,HLIM
      COMMON/STOR05/ VT(3),VA(3),VEL,ACC,VOLIM,VLIM
      COMMON/STOR07/ T,DY,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     ;               DMIN,DMAX,KSTEP,NGOOD
C
      CALL RK4(14,SOLV5D)
  100 CALL RCM(14,SOLV5D)
      IF(T-TBO) 100,110,110
  110 KSTEP = 6
  200 CALL RCM(6,SOLV3D)
      IF(VT(3) .LT.0.0) GO TO 200
      KSTEP = 2
  300 CALL RCM(6,SOLV3D)
      CALL IPSTEP($300,$400)
  400 CALL RK4(6,SOLV3D)
      IF(ALT .LT. HLIM) GO TO 500
      CALL IPSTEP($400,$400)
  500 CALL IPCAL
      RETURN
      END
```

```
      SUBROUTINE TPRINT(K,L)                                            TPRINT
C     DETAILED PRINT-OUT OF THE TRAJECTORY.
C
      INTEGER NKSTEP
      REAL LATL,LONGL,LAT,LONG,MASS,MPROP,MPL,MDOT,MACH,LCG,LCP,IYY,MSEP
      DOUBLE PRECISION DPS,DPC,AR1,AR2
      DIMENSION CSP(2)
      COMMON/STOR00/ ZERO,ONE,PI,QPI,HPI,TPI,CF(9)
      COMMON/STOR01/ GO,GR,OMEGA,GM,RE,RO,ESO,ESO1,ESO1
      COMMON/STOR03/ XO,YO,ZO,R,RXY,ALT,SRHLIM
      COMMON/STOR04/ GDLATL,LATL,LONGL,THETAL,PHIL,GDLAT,LAT,LONG,THETA,
     .               PHI,GAMAL,PSIL,GAMA,PSI,ELGDL,AZGDL,ELGD,AZGD,EL,
     .               AZ,SRAZ
      COMMON/STOR05/ VT(3),VA(3),VEL,ACC,VELIM,VLIM
      COMMON/STOR06/ RT(9)
      COMMON/STOR07/ T,DT,TNP1,DTLA,DTAI,TBO,RKSTEP,EPBIG,EPTINY,
     .               DMIN,DMAX,KSTEP,NGOOD
      COMMON/STOR08/ S(14),F(14,5),DPS(14),DPC(14),ERW(14),SP(6,20)
      COMMON/STOR09/ MASS,MPROP,MPL,SUBM(32),MDOT,THR,MACH,CA,AD,QS,
     .               SLOPE,CMP,LCG,LCP,IYY,SY,SZ
      COMMON/STOR10/ WE,WN,MLAST,WIND,WMAZ,ALOW,AHIGH,NLEV,ALEV(100),
     .               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR12/ NPMAX,NSMAX,NPL,NANG,NVAR,NP,NP1,JSPENT,JS,JSC,NCON
      COMMON/STOR14/ ITAPE1,ITAPE2,ITAPE3,ITAPE4,IPRINT,IPUNCH,INDEX
      COMMON NRST,NSEP,NSP,NM(14),KSP(20),BMASS,T30,TAV(3),ANG(61),B(4),
     .       PLM(20),PHT(32),TSEP(32),MSEP(32),TSP(20),SPM(20),A(99,14)
      DATA IPREV,CSP/ 0, 1H0, 1H1/
C
      CALL FXORT(71,1,1,0)
      IF(INDEX .EQ. 0) PRINT 1000, CSP(IPRINT)
      I = T
      IF(I .EQ. 0) GO TO 1
      IF(L .EQ. 1) GO TO 1
      IF(I .EQ. IPREV) RETURN
    1 IPREV = T
      INDEX = INDEX + 1
      IF(MOD(INDEX,38).EQ.0) PRINT 1000, CSP(2)
C
      ETA = 0.
      IF(T .GT. TBO) GO TO 15
      AR1 = SQRT(VA(2)**2 + VA(3)**2)
      AR2 = ABS(VA(1))
```

```
      ETA = DATAN(AR1/AR2)
      S1=S(4)*OMEGA*S(2)
      S2=S(5)-OMEGA*S(1)
      VT(1) = RTI(1)*S1 + RTI(2)*S2 + RTI(3)*S(6) + WN
      VT(2) = RTI(4)*S1 + RTI(5)*S2 + RTI(6)*S(6) + WE
      VT(3) = RTI(7)*S1 + RTI(8)*S2 + RTI(9)*S(6)
   15 VT(1) = VT(1) + WN
      VT(2) = VT(2) + WE
      S1 = VT(1)**2 + VT(2)**2
      VELO = SQRT(VT(3)**2 + S1)
      AR1 = -VT(3)
      AR2 = SQRT(S1)
      ELE = DATAN(AR1/AR2)
      AR1 = VT(2)
      AR2 = VT(1)
      AZI = DATAN2(AR1,AR2)
      IF(AZI.LT.0.) AZI=AZI+TPI
      S1 = RE/R
      CALL AZRAN(S1*S(1),S1*S(2),S1*S(3),T)
      Q = 0.5*QS/B(3)
      QALFA = Q*ETA*CF(1)
      ACC = 0.
      J = -2
      DO 20 I=1,3
      J=J+3
      S1 = RTI(J)*F(4,K) + RTI(J+1)*F(5,K) + RTI(J+2)*F(6,K) -
     .   (OMEGA**2)*(RTI(J)*S(1)+RTI(J+1)*S(2))
     .     2.0*OMEGA*(RTI(J+1)*S(4)-RTI(J)*S(5))
      ACC = ACC + S1**2
   20 CONTINUE
      ACC = SQRT(ACC)
      PRINT 1010, T,SR*CF(4),SRAZ*CF(1),ALT,ELE*CF(1),AZI*CF(1),VELO,
     . ACC/G0,MACH,THR,AD,Q,ETA*CF(1),QALFA,MASS*G0,WN,WE
C
 1000 FORMAT( A1,!   TIME  RANGE  BEARING  ALTITUDE  FLT/EL  FLT/AZ  VEL
     .,OCITY  ACC  MACH  THRUST   DRAG. D/PGES  ETA   QALFA  WEIGHT  WIN.
     .,DN  WINDE!,Z/)
 1010 FORMAT(1X,F7.2,2F8.2,F10.0,2F8.2,E10.0,F8.1,E6.2,F8.0,F7.0,F8.0,
     .F7.2,F7.0,F8.1,2F7.1)
      RETURN
      END
```

```
      SUBROUTINE WCONST
C     CONSTANT WIND TABLE.
C
      COMMON/STOR00/ ZERO,ONE,PI,QPI,HPI,TPI,CF(9)
      COMMON/STOR10/ WE,WN,MLAST,WIND,WNAZ,ALOW,AHIGH,NLEV,ALEV(100),
     .               ALTW(100),VELE(100),VELN(100)
C
      WAZ = AMOD(WNAZ+PI,TPI)
      ALTW(1) = ALOW
      ALTW(2) = ALOW + 0.1
      ALTW(3) = AHIGH
      VELE(1) = 0
      VELE(2) = WIND*SIN(WAZ)
      VELE(3) = VELE(2)
      VELN(1) = 0
      VELN(2) = WIND*COS(WAZ)
      VELN(3) = VELN(2)
      RETURN
      END
```

```
      SUBROUTINE WTERP
C     INTERPOLATES NORTH AND EAST WIND COMPS FROM WTABLE.
C
      INTEGER ORDER
      COMMON/STOR03/ X0,Y0,Z0,R,RXY,ALT,SR,HLIM
      COMMON/STOR10/ WE,WN,MLAST,WIND,WMAZ,ALOW,AHIGH,NLEV,ALEV(100),
     ;               ALTW(100),VELE(100),VELN(100)
      COMMON/STOR11/ KPREV,HPREV,MPREV,APREV,LPREV(14),VPREV(14),
     ;               ORDER(14)
      EQUIVALENCE (ALT,A)
C
      M = MPREV
      IF(A.GT.ALTW(M) .AND. A.LE.ALTW(M+1)) GO TO 200
      IF(A .GT. ALTW(MLAST)) GO TO 220
      IF(A .LT. ALTW(1)) GO TO 220
      IF(A - APREV) 100,210,110
  100 JB = 2
      JE = M+1
      GO TO 120
  110 JB = M+1
      JE = MLAST
  120 DO 130 J=JB,JE
      JSAVE = J
      IF(A .LT. ALTW(J)) GO TO 140
  130 CONTINUE
  140 M = JSAVE - 1
  200 MPREV = M
      APREV = A
      FRAC = (A-ALTW(M))/(ALTW(M+1)-ALTW(M))
      WE = VELE(M) + FRAC*(VELE(M+1)-VELE(M))
      WN = VELN(M) + FRAC*(VELN(M+1)-VELN(M))
  210 RETURN
  220 WE = 0.
      WN = 0.
      RETURN
      END
```