PODEMS - A POINT DEFENSE MISSILE SIMULATION

by

Michael H. Redlin

March 1973

Approved for Public Release; Distribution Unlimited
ABSTRACT

A Point Defense Missile Simulation has been developed. This report describes the concept of such a missile, the basic features of the simulation program including the integration routine and the jet reaction controllers, and provides a FORTRAN coded source program.
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>II. Computer Program Explanations</td>
<td>3</td>
</tr>
<tr>
<td>A. Definitions of Coordinate Systems</td>
<td>3</td>
</tr>
<tr>
<td>B. Missile Position and Orientation</td>
<td>3</td>
</tr>
<tr>
<td>C. Quaterions</td>
<td>5</td>
</tr>
<tr>
<td>D. Differential Equations for Rigid Body</td>
<td>7</td>
</tr>
<tr>
<td>E. Definition of Relative Wind Orientation</td>
<td>8</td>
</tr>
<tr>
<td>F. Aerodynamic Data</td>
<td>8</td>
</tr>
<tr>
<td>G. Integration Routine</td>
<td>9</td>
</tr>
<tr>
<td>H. Other Subroutines</td>
<td>10</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>24</td>
</tr>
<tr>
<td>APPENDIX A - Computer Program Listing</td>
<td>25</td>
</tr>
<tr>
<td>APPENDIX B - Listing of Program Variables</td>
<td>57</td>
</tr>
<tr>
<td>INITIAL DISTRIBUTION LIST</td>
<td>64</td>
</tr>
<tr>
<td>FORM DD 1473</td>
<td>65</td>
</tr>
</tbody>
</table>
LIST OF SYMBOLS

\[ \begin{align*} 
&c_i \quad \text{Corrector for state variables at time step } i \\
&C_1' \quad \text{Moment coefficients of baseline missile expressed in pitch axes} \\
&C_{m'} \quad \\
&C_n' \quad \\
&C_x' \quad \text{Force coefficients of baseline missile expressed in pitch axes} \\
&C_y' \quad \\
&C_z' \quad \\
&d \quad \text{Missile diameter, ft} \\
&e_0 - e_3 \quad \text{Quaterion parameters} \\
&h \quad \text{Step size for fast integration speed, secs} \\
&h_s \quad \text{Step size for slow integration speed, secs} \\
&I_x, I_y \quad \text{Longitudinal and transverse moments of inertia, slugs-ft}^2 \\
&K_m, K_n \quad \text{Moment amplification factors due to JRC's} \\
&K_y, K_z \quad \text{Force amplification factors due to JRC's} \\
&m \quad \text{Missile mass, slugs} \\
&m_i \quad \text{Modified value of state variables at time step } i \\
&m'_i \quad \text{Derivatives of state variables evaluated with } m_i \\
&p, q, r \quad \text{Missile angular rates about body axes, rad/sec} \\
&p_i \quad \text{Predicted value of state variables at time step } i \\
&r_{MT} \quad \text{Range from target to missile, ft} \\
&S \quad \text{Reference area, ft}^2 \\
&T_{11-T33} \quad \text{Elements of transformation matrix from inertial axes to body axes} \\
&T_r \quad \text{Thrust of rocket, lbs} \\
&T_{JRC} \quad \text{Thrust of JRC's, lbs} \\
u, v, w \quad \text{Components of missile's inertial velocity in body axes, ft/sec} 
\end{align*} \]
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_T, v_T, w_T$</td>
<td>Components of total relative wind velocity in body axes, ft/sec</td>
</tr>
<tr>
<td>$u_w, v_w, w_w$</td>
<td>Components of true wind velocity in body axes, ft/sec</td>
</tr>
<tr>
<td>$V$</td>
<td>Magnitude of relative wind velocity, ft/sec</td>
</tr>
<tr>
<td>$x, y, z$</td>
<td>Body axes coordinate system</td>
</tr>
<tr>
<td>$x_{MT}, y_{MT}, z_{MT}$</td>
<td>Target position relative to missile in body axes, ft</td>
</tr>
<tr>
<td>$x', y', z'$</td>
<td>Pitch axes coordinate system</td>
</tr>
<tr>
<td>$X, Y, Z$</td>
<td>Inertial axes coordinate system</td>
</tr>
<tr>
<td>$X_I, Y_I, H_T$</td>
<td>North-south, east-west, and height inertial position of missile, ft</td>
</tr>
<tr>
<td>$X_F, Y_F, Z_F$</td>
<td>Total forces along missile body axes, lbs</td>
</tr>
<tr>
<td>$X_M, Y_M, Z_M$</td>
<td>Total moments along missile body axes, lbs-ft</td>
</tr>
<tr>
<td>$X_T, Y_T, H_T$</td>
<td>North-south, east-west, and height inertial position of target, ft</td>
</tr>
<tr>
<td>$x_{cg}$</td>
<td>Position of missile center of gravity, ft</td>
</tr>
<tr>
<td>$x_{JRC}$</td>
<td>Position of missile JRC thrusters, ft</td>
</tr>
<tr>
<td>$\dot{y}_i, \ddot{z}_i$</td>
<td>Predicted values of state variables at time step $i$</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>Missile angle of attack, degrees</td>
</tr>
<tr>
<td>$\epsilon$</td>
<td>Quaternion orthogonality error</td>
</tr>
<tr>
<td>$\epsilon_B$</td>
<td>Angular error between gyro spin axis and target line of sight, rad</td>
</tr>
<tr>
<td>$\epsilon_C$</td>
<td>Angular error between coil housing axis and target line of sight, rad</td>
</tr>
<tr>
<td>$\theta, \phi, \psi$</td>
<td>Standard missile Euler angles, deg</td>
</tr>
<tr>
<td>$\theta_B$</td>
<td>Angular error associated with seeker gimbal ring, rad</td>
</tr>
<tr>
<td>$\sigma_A, \sigma_B$</td>
<td>Target elevation and azimuth angles relative to missile axes</td>
</tr>
<tr>
<td>$\phi_1$</td>
<td>Seeker roll angle, rad</td>
</tr>
<tr>
<td>$\phi_J$</td>
<td>Orientation of relative wind with respect to thruster axes, deg</td>
</tr>
<tr>
<td>$\phi_w$</td>
<td>Orientation of relative wind with respect to missile axes, deg</td>
</tr>
<tr>
<td>$\psi_2$</td>
<td>Seeker coil housing look angle, rad</td>
</tr>
</tbody>
</table>
\( \gamma_3 \)  Angular error associated with seeker gimbal ring, rad

\( \Omega_c \)  Angular rotation rate of coil housing, rad/sec

\( \Omega_{G_y} \)  Angular rotation rate of gyro spin axis about its \( y \) axis, rad/sec

\( \Omega_{G_z} \)  Angular rotation rate of gyro spin axis about its \( z \) axis,

\( \Omega_Y \)  Angular rotation rate of seeker yoke, rad/sec
<table>
<thead>
<tr>
<th>LIST OF FIGURES</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Definition of Missile Fixed Coordinate System</td>
<td>14</td>
</tr>
<tr>
<td>2. Definition of Pitch Axes Coordinate System</td>
<td>14</td>
</tr>
<tr>
<td>3. Relative Wind Orientation</td>
<td>15</td>
</tr>
<tr>
<td>4. Relative Wind Orientation With Respect to JRC Thruster Axis</td>
<td>16</td>
</tr>
<tr>
<td>5. $C_z$, vs $\alpha$ for $\xi_w = 0, \pi/4, \pi/2$</td>
<td>17</td>
</tr>
<tr>
<td>6. $C_m$, vs $\alpha$ for $\xi_w = 0, \pi/4, \pi/2$</td>
<td>18</td>
</tr>
<tr>
<td>7. $K_z$ vs $\alpha$ for $\phi_J = 0, \pi/2, \pi$</td>
<td>19</td>
</tr>
<tr>
<td>8. $K_m$ vs $\alpha$ for $\phi_J = 0, \pi/2, \pi$</td>
<td>19</td>
</tr>
<tr>
<td>9. Dual Speed Integration for One Large Step $h_s$</td>
<td>20</td>
</tr>
<tr>
<td>10. Definitions of JRC Orientations</td>
<td>21</td>
</tr>
<tr>
<td>11. Definition of Target Azimuth and Elevation Angles</td>
<td>22</td>
</tr>
<tr>
<td>12. Definition of Missile Parameters</td>
<td>23</td>
</tr>
</tbody>
</table>
INTRODUCTION

A Point DEfense Missile Simulation (PODEMS) program has been developed and this report offers a description of its basic features, structure, and requirements. Although rather straightforward in nature, this program provides the basic framework from which further simulations of increased complexity and sophistication can be easily implemented.

The concept of a point defense missile as defined by this effort can be best understood by analyzing a typical flight. The seeker of the missile initially acquires a low-altitude, high-speed, incoming target. The surface-to-air missile launches vertically, and then immediately performs a rapid pitch-over maneuver toward the target with a consequent altitude gain of less than 500 feet. The primary controllers for this phase of the flight are two pairs of diametrically opposed jets (jet reaction controllers-JRC) aligned perpendicular to both one another and the missile axis of symmetry. Upon attainment of an approximately horizontal flight path, the primary controller of the missile transfers to typical aerodynamic surfaces (CANARDS) which then guide the missile to intercept. The maneuvers of lift-off and pitch-over, for which the time frame is 1. - 1.5 secs, are of primary interest and therefore are the object of this simulation.

The important features of the simulation are: acceptance of a vertical launch configuration, implementation of JRC controlled maneuvers, a detailed simulation of a large-look angle seeker, and a dual speed integration routine.

The basis of the simulation is a dual speed integration routine using Hamming's predictor-modifier-corrector formulation for the recursion equations. The user has the option of specifying which state variables are integrated with the two different steps size h and $h_s$. Additionally, because of the singularities evident in the Euler angles, four quaterion parameters are
employed to uniquely represent the missile attitude for all possible orientations.

All of the missile parameters, including the aerodynamic data for both the baseline missile and the JRC's, are listed within the report. However, the reader is cautioned against the assumption that a particular missile is being simulated for the data are only representative of this type of missile.

The intention of this effort was to provide a general basic structural program capable of simulating a missile as a rigid body with the specific subroutines for the aerodynamic data, rigid body parameters, etc. to be supplied by the user as required.
A. Definitions of Coordinate Systems

1. Inertial Coordinate System

An inertially fixed coordinate system \((X,Y,Z)\) is attached to the earth with the origin at ground zero, the \(X\) axis indicating north, the \(Y\) axis indicating east and the \(Z\) axis indicating the local vertical (positive downward).

2. Missile Fixed Coordinate System

A body fixed coordinate system \((x,y,z)\) is located with its origin at the missile center of gravity, the \(x\) axis as the missile's axis of symmetry (positive pointing forward), the \(y\) axis rotated negative \(45^\circ\) from the right-hand pitch canard, and the \(z\) axis rotated accordingly. See Figure 1.

3. Pitch Axis Coordinate System

The origin and the \(x'\) axis of the pitch axes coordinate system \((x',y',z')\) are coincident with their counterparts in the missile fixed axes system, while the \(z'\) axis always coincides with the projection of the relative wind vector onto the \(y, z\) plane. The angle \(\phi\) indicates the relative rotation of \((x',y',z')\) with respect to \((x,y,z)\). See Figure 2.

B. Missile Position and Orientation

The coordinates \(X, Y, HT\) locate the missile center of gravity with respect to the inertial coordinate system in the north-south, east-west, and height above ground zero directions respectively.

The orientation of the missile axes with respect to the inertial system is monitored using the standard Euler angles \(\gamma, \theta, \phi\) (yaw, pitch, roll) with the order of rotation as given. The resulting transformation matrix from inertial coordinates to missile coordinates is
\[
\begin{bmatrix}
\mathbf{x} \\
\mathbf{y} \\
\mathbf{z}
\end{bmatrix} =
\begin{bmatrix}
T_{11} & T_{12} & T_{13} \\
T_{21} & T_{22} & T_{23} \\
T_{31} & T_{32} & T_{33}
\end{bmatrix}
\begin{bmatrix}
\mathbf{X} \\
\mathbf{Y} \\
\mathbf{Z}
\end{bmatrix}
\]

where

\[\begin{align*}
T_{11} &= \cos \psi \cos \theta \\
T_{12} &= \sin \psi \cos \theta \\
T_{13} &= -\sin \theta \\
T_{21} &= \cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi \\
T_{22} &= \sin \psi \sin \theta \sin \phi + \cos \psi \cos \phi \\
T_{23} &= \cos \theta \sin \phi \\
T_{31} &= \cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi \\
T_{32} &= \sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi \\
T_{33} &= \cos \theta \cos \phi
\end{align*}\]

In the actual simulation, the Euler angles are not employed because of their singularity at \( \theta = \pm 90 \). However they are calculated and outputted to aid the program user in visualization of the missile orientation. The following equalities define the Euler angles when \( \theta \neq \pm 90 \):

\[\begin{align*}
\theta &= \sin^{-1} (-T_{13}) \\
\psi &= \tan^{-1} \left( \frac{T_{12}}{T_{11}} \right) \quad (4 \text{ quadrant } \tan^{-1}) \\
\phi &= \tan^{-1} \left( \frac{T_{23}}{T_{33}} \right) \quad (4 \text{ quadrant } \tan^{-1})
\end{align*}\]

However, when \( \theta = \pm 90 \), \( \psi \) and \( \phi \) are undefined and the following convention is adopted:

\[\begin{align*}
\psi &= 0 \\
\phi &= \tan^{-1} \left( \frac{T_{21}}{T_{31}} \right) \quad (4 \text{ quadrant } \tan^{-1})
\end{align*}\]
C. Quaternions

To avoid the singularity of the Euler angles at $\theta = \pm 90$, the quaternion system of four coordinates is adopted. The introduction of an extra coordinate into the system removes the singularity but requires the addition of a constraint equation on the four parameters.

The four coordinates are $e_0, e_1, e_2, e_3$ with the constraint of $e_0^2 + e_1^2 + e_2^2 + e_3^2 = 1$ (orthogonality). The elements of the previously mentioned transformation matrix are functions of these coordinates.

\[
\begin{align*}
T_{11} &= e_0^2 + e_1^2 - e_2^2 - e_3^2 \\
T_{12} &= 2(e_1 e_2 + e_0 e_3) \\
T_{13} &= 2(e_1 e_3 - e_0 e_2) \\
T_{21} &= 2(e_1 e_2 - e_0 e_3) \\
T_{22} &= e_0^2 + e_2^2 - e_1^2 - e_3^2 \\
T_{23} &= 2(e_2 e_3 + e_0 e_1) \\
T_{31} &= 2(e_1 e_3 + e_0 e_2) \\
T_{32} &= 2(e_2 e_3 - e_0 e_1) \\
T_{33} &= e_0^2 + e_3^2 - e_1^2 - e_2^2
\end{align*}
\]

The differential equations for the quaternion parameters as functions of the missile angular rates $(p,q,r)$ are:
\[ \dot{e}_0 = -\frac{1}{2} (e_1 p + e_2 q + e_3 r) \]
\[ \dot{e}_1 = \frac{1}{2} (e_0 p - e_3 q + e_2 r) \]
\[ \dot{e}_2 = \frac{1}{2} (e_3 p + e_0 q - e_1 r) \]
\[ \dot{e}_3 = \frac{1}{2} (-e_2 p + e_1 q + e_0 r) \]

Mechanization of the constraint equation is achieved by defining an error

\[ \epsilon = 1 - (e_0^2 + e_1^2 + e_2^2 + e_3^2) \]

which is a measure of the violation of the constraint and applying a correction factor to each differential equation which reduces the error. With a value of \( K = 1 \) the equations remain correctly constrained within \( |\epsilon| \leq 10^{-6} \).

\[ \dot{e}_0 = -\frac{1}{2} (e_1 p + e_2 q + e_3 r) + K e_0 \epsilon \]
\[ \dot{e}_1 = \frac{1}{2} (e_0 p - e_3 q + e_2 r) + K e_1 \epsilon \]
\[ \dot{e}_2 = \frac{1}{2} (e_3 p + e_0 q - e_1 r) + K e_2 \epsilon \]
\[ \dot{e}_3 = \frac{1}{2} (-e_2 p + e_1 q + e_0 r) + K e_3 \epsilon \]

The required initial conditions on \( e_0, e_1, e_2, e_3 \) are given as functions of the initial \( \psi, \theta, \phi \) by

\[ e_0 = \cos (\psi/2) \cos (\theta/2) \cos (\phi/2) + \sin (\psi/2) \sin (\theta/2) \sin (\phi/2) \]
\[ e_1 = \cos (\psi/2) \cos (\theta/2) \sin (\phi/2) - \sin (\psi/2) \sin (\theta/2) \cos (\phi/2) \]
\[ e_2 = \cos (\psi/2) \sin (\theta/2) \cos (\phi/2) + \sin (\psi/2) \cos (\theta/2) \sin (\phi/2) \]
\[ e_3 = -\cos (\psi/2) \sin (\theta/2) \sin (\phi/2) + \sin (\psi/2) \cos (\theta/2) \cos (\phi/2) \]
D. Differential Equations for Rigid Body

With \( X_F, Y_F, Z_F \) defined as the total forces on the missile expressed in missile axes \( x,y,z \) respectively and \( X_M, Y_M, Z_M \) defined as the total moments about the missile center of gravity expressed in the same axis system, the differential equations of motion are:

\[
\begin{align*}
\dot{u} &= rv - qw + \frac{X_F}{m} \\
\dot{v} &= pw - ru + \frac{Y_F}{m} \\
\dot{w} &= qu - pv + \frac{Z_F}{m} \\
\dot{p} &= \frac{X_M}{I_x} \\
\dot{q} &= -\frac{pr (I_x - I_y) + Y_M}{I_y} \\
\dot{r} &= +\frac{pq (I_x - I_y) + Z_M}{I_y}
\end{align*}
\]

\[
\begin{bmatrix}
\dot{x}_T \\
\dot{y}_T \\
\dot{z}_T
\end{bmatrix} = T^{-1} \begin{bmatrix}
u \\ v \\ -w
\end{bmatrix}
\]

\[
\begin{align*}
\dot{e}_0 &= -\frac{1}{2} (e_1 p + e_2 q + e_3 r) + K e_0 \\
\dot{e}_2 &= \frac{1}{2} (e_0 p - e_3 q + e_2 r) + K e_1 \\
\dot{e}_3 &= \frac{1}{2} (e_3 p + e_0 q - e_1 r) + K e_2 \\
\dot{e}_4 &= \frac{1}{2} (-e_2 p + e_1 q + e_0 r) + K e_3
\end{align*}
\]

It is assumed that the missile is symmetric about the \( x \) axis, no cross products of inertia exist and \( I_y = I_z \). \( T^{-1} = T \) where \( T \) is the transformation matrix from \((X,Y,Z)\) to \((x,y,z)\).
E. Definition of the Relative Wind Orientation

Two angles $\alpha$ and $\theta_w$ define the orientation of the relative wind vector with respect to the missile axis system as shown in Figure 3, where $u_T$, $v_T$, $w_T$ are the $x, y, z$ components of the resultant wind and $V = \sqrt{u_T^2 + v_T^2 + w_T^2}$. Along each axis the resultant wind component is the difference of the missile inertial velocity and the true wind for that axis.

$$u_T = u - u_w$$

$$v_T = v - v_w$$

$$w_T = w - w_w$$

Additionally, $\theta_J$ defines the orientation of the resultant wind with respect to each individual JRC (Jet Reaction Controller). Positive $\theta_J$ is defined as shown in Figure 4.

F. Aerodynamic Data

The missile aerodynamics are divided into two distinct categories: (1) aerodynamic coefficients for the baseline missile (no JRC) and (2) amplification factors which represent the effect of the JRC thrusters. Both sets of data are functionally dependent on $\alpha$ and either $\theta_w$ or $\theta_J$ but not on Mach number.

The baseline aerodynamic coefficients $C_{x'}$, $C_{y'}$, $C_{z'}$, $C_{JRC}$, $C_{m'}$, $C_{n'}$ are given as shown in Figures 5-6. For the present simulation $C_{x'} = C_{y'} = C_{JRC} = C_{m'} = C_{n'} = 0$.

The effects of the JRC thrusters are summarized by amplification factors as follows (y force and moment amplification factors are used as examples)

$$K_y = \frac{Y_F|_{JRC\ on} - Y_F|_{JRC\ off}}{T_{JRC}}$$

$$K_m = \frac{Y_M|_{JRC\ on} - Y_M|_{JRC\ off}}{T_{JRC} (X_{JRC})}$$
The specific values of $K_y$, $K_z$, $K_m$, $K_n$ as programmed are shown in Figures 7-8. For this simulation $K_y = K_n = 0$. Additionally, the effects of the JRC jets are assumed to be independent.

G. Integration Routine

Hamming's predictor, modifier, corrector set of recursion equations are used for the dual speed numerical integration of the problem state variables. The following is a brief explanation of the equations.

For a system of $n$ ordinary differential equations

$$y' = f(x, y)$$

where

$$y' = dy/dx,$$

a sequence of the solution variables

$$y_i = y(x_i) \quad i = 1, 2, \ldots$$

can be expressed as a function of previous $y_i$ and $y_i'$. With $h = x_{i+1} - x_i$ Hamming's method is:

**PREDICT:**

$$p_{i+1} = y_{i-3} + \frac{4h}{3} (2y_i' - y_{i-1}' + 2y_{i-2}' - 2y_i)$$

**MODIFY:**

$$m_{i+1} = p_{i+1} - \frac{\frac{112}{121}}{} (p_i - c_i)$$

$$m_i' = f(x_{i+1}, m_i)$$

**CORRECT:**

$$c_{i+1} = \frac{1}{8} [9y_i - y_{i-2} + 3h (m_i' + 2y_i' - y_{i-1}')]$$

**FINAL VALUE:**

$$y_{i+1} = c_{i+1} + \frac{9}{121} (p_{i+1} - c_{i+1})$$
Each advance of \( h \) in the independent variable \( x \) requires two evaluations of \( y' \), once for the predictor and once for the corrector. The method is numerically stable with truncation errors to the order of \( h^5 \).

The values of \( y \) and \( y' \) from the past three intervals are necessary, thus a starting technique is required. The conventional application of a 4th order Runge-Kutta integration method on the first three steps was discarded in favor of calculating the required state variables by a Euler backstep. Specifically, for \( i = 0 \),

\[
\begin{align*}
y_{-3} &= y_0 - 3hy'_0 \\
y_{-2} &= y_0 - 2hy'_0 \\
y_{-1} &= y_0 - hy'_0 \\
y'_{-2} &= y'_{-1} = y'_0
\end{align*}
\]

This method suffers from inaccuracy when \( y'_{-2} \) and \( y'_{-1} \) differ appreciably from \( y'_0 \). However, in this simulation, no variation in the solution was detected from the application of the less accurate Euler backstep when compared with a Runge-Kutta starter.

An additional complexity was introduced by the requirement of a dual speed integration algorithm because of computational time considerations. Now there are two systems of differential equations:

\[
\begin{align*}
y' &= f(x,y,z) \\
z' &= f(x,y,z)
\end{align*}
\]

with the \( z \) equations requiring smaller time steps than the \( y \) equations for the same accuracy criteria. With \( h \) and \( h_s \) defined as the smaller
and larger step sizes respectively, figure 9 depicts the sequencing of the algorithm for one step of $h_s$. A ratio of $h_s/h = 5$ is chosen for illustration although this is variable at the operator's option.

H. Other Subroutines

CONTROL SYSTEM

The JRC's were assumed to be the primary controlling elements for the initial missile trajectory and, therefore, the canard deflection are identically zero for this phase of the flight.

Two control equations govern the action of the JRC's, one for each pair of opposing jets. Figure 10 defines the jet numbers and orientations. For illustration the control of jets 1 and 3 is presented. A variable $FRMTZ$ is defined as a function of missile-target relative position and rates. The exact specification for this equation is the operators responsibility. When $FRMTZ > 0$ jet 3 is on while jet 1 is off. If $FRMTZ < 0$ the reverse is true, and when $FRMTZ = 0$ both jets are off.

As an example of a possible control equation consider

$$FRMTZ = \sigma_A + K \dot{\sigma}_A$$

where $\sigma_A$ is defined in Figure 11.

SEEKER

This program incorporates a simulation of a large look angle version of a present day seeker. The simulation was supplied by the manufacturer and was only slightly modified to interface correctly. The system description will not be discussed here, only the inputs and outputs of the subroutine.

The following information is required by subroutine SEEKER: $X_I, Y_I, HT, XT_I, YT_I, HTT, [T], p, q, r, \dot{p}, \dot{q}, \dot{r}$. The subroutine returns: $\hat{\psi}_1, \psi_2, \psi_3, \theta_4, \epsilon_B, \epsilon_C, \hat{\theta}_1, \dot{\psi}_2, \Omega_y, \Omega_c, \Omega_{gy}, \Omega_{gz}$ for outputting if desired.
RIGID BODY PARAMETERS

All rigid body parameters (mass, $I_x$, $I_y$, C.G. position) are linearly interpolated between the initial values at lift-off and the final values when the missile thrust motor is expended. The instantaneous position of the center of gravity $x_{cg}$ is defined relative to the reference point for the aerodynamic data as in Figure 12. The following table indicates the parameters as used in the program.

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>LIFT OFF</th>
<th>BURN OUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m$</td>
<td>6.742</td>
<td>4.710</td>
</tr>
<tr>
<td>$I_y$</td>
<td>65.1</td>
<td>48.1</td>
</tr>
<tr>
<td>$I_x$</td>
<td>.420</td>
<td>.245</td>
</tr>
<tr>
<td>$x_{cg}$</td>
<td>-.321</td>
<td>.406</td>
</tr>
</tbody>
</table>

$S = .13635 \text{ ft}^2$

$d = .4167 \text{ ft}$

$x_{JRC} = 2.434 \text{ ft}$

$T_{JRC} = 400.1 \text{ lbs}$

$T_r = 3000. \text{ lbs}$

ATMOSPHERE

Both the density and acoustical velocity of air as functions of height are generated within ATMOS. A linear interpolation of these parameters is based on data from an ICAO Standard Atmosphere Table at heights of 0. and 1000. ft.

Additionally values for the X and Y components of surface winds maybe entered as constant or functions of altitude depending on the operator's preference.
TARGET

Subroutine target calculates the time history trajectory of the target as a function of its initial inertial position, constant inertial velocity components and time.

THRUST

Missile thrust is assumed to be a constant THR for a duration of burn TBURN, after which THR = 0 .
Figure 1. Definition of Missile Fixed Coordinate System

End View Looking Forward

Figure 2. Definition of Pitch Axes Coordinate System

End View Looking Forward
\[ \alpha = \cos^{-1} \left( \frac{u_T}{V} \right) \]

\[ \xi_w = \cos^{-1} \left( \frac{w_T}{(v_T^2 + w_T^2)^{\frac{1}{2}}} \right) \]

If \( \alpha = 0 \), \( \xi_w = 0 \)

Figure 3. Relative Wind Orientation
Figure 4. Relative Wind Orientation With Respect to the JRC Thruster Axis.
Figure 5. $C_z'$ vs $\alpha$ for $\bar{\alpha}_w = 0, \pi/4, \pi/2$.

$\alpha$ in degrees
Figure 6. $C_{m}$ vs $\alpha$ for $\xi_{w} = 0$, $\pi/4$, $\pi/2$. 

$C_{m}$ vs $\alpha$ for $\xi_{w} = 0$, $\pi/4$, $\pi/2$. 

Figure 6. $C_{m}$ vs $\alpha$ for $\xi_{w} = 0$, $\pi/4$, $\pi/2$. 

18
\[ J = \frac{n}{2} \]

\[ \alpha \text{ in degrees} \]

**Figure 7.** \( K_z \) vs \( \alpha \) for \( \hat{\phi}_J = 0, \pi/2, \pi \).

\[ \hat{\phi}_J = \pi/2, \pi \]

\[ \hat{\phi}_J = 0 \]

\[ \alpha \text{ in degrees} \]

**Figure 8.** \( K_m \) vs \( \alpha \) for \( \hat{\phi}_J = 0, \pi/2, \pi \).
1. At \( x = x_i \) evaluate \( z'_i, y'_i \) and predict \( \widetilde{z}_{i+1} \) and \( \widetilde{y}_{i+5} \) using \( h \) and \( h_s \) respectively.

2. At \( x = x_{i+1} \) evaluate \( z'_{i+1}, y'_{i+1} \) and correct \( z_{i+1} \) and \( y_{i+5} \) using \( h \) and \( h_s \) respectively.

3. With \( y_{i+5} \) now fixed, at \( x = x_{i+1}, x_{i+2}, \ldots \) evaluate only \( z'_{i+1}, z'_{i+2}, \ldots \) and sequentially predict \( \widetilde{z}_{i+2}, \widetilde{z}_{i+3}, \ldots \) and correct \( z_{i+2}, z_{i+3}, \ldots \) until \( x = x_{i+5} \).

4. Repeat steps (1) - (3) for successive increments of \( h_s \).

Figure 9. Dual Speed Integration for One Large Step \( h_s \).
Figure 10. Definition of JRC Orientation.
\[
\sigma_A = \tan^{-1}(\frac{z_{mT}}{x_{mT}})
\]

\[
\sigma_B = \tan^{-1}(\frac{y_{mT}}{x_{mT}})
\]

Figure 11. Definition of Target Azimuth and Elevation Angles.
Figure 12. Definition of Missile Parameters.
REFERENCES


APPENDIX A

PROGRAM LISTING

The following is a FORTRAN listing of the simulation program designed for compatibility with the United Computing Service, Inc. time sharing system. A typical input/output listing is also included.
**Program SXDF**

**Common Variables**

- Y(9), DY(9), XH, WINDX, WINDY, WINDZ, RMA
- CFY, XF, YF, ZF, THR, XG, RMA, FCY, Y, YM, ZM, X, Y, Z

**Communications**

1. **COMMON/TRANS/T1**
2. **COMMON/EQUIVALENCE**
3. **program** 03/28/73.
4. **CALL COMMON/TRANS/T1**

**Input/Output**

- VSND, SFHY, HTE, GUE, ALPHA, PHI, XMAX, TUR, RHO, VEL, FSI, THT
- COMMON/TRANS/T1, COMMON/EQUIVALENCE

**Variables**

- CX, DY, Z, CX, CY, C, CY, C, CY

**Common Variables**

- OR, OR, OR, OR

**Main Routine**

- CALL COMMON/TRANS/T1

**Input/Output**

- VSND, SFHY, HTE, GUE, ALPHA, PHI, XMAX, TUR, RHO, VEL, FSI, THT

**Communications**

1. **COMMON/TRANS/T1**
2. **COMMON/EQUIVALENCE**
3. **program** 03/28/73.
4. **CALL COMMON/TRANS/T1**

**Initialization**

- IRT2 = 2 * IRATIO

**Main Routine**

- CALL COMMON/TRANS/T1

**Input/Output**

- VSND, SFHY, HTE, GUE, ALPHA, PHI, XMAX, TUR, RHO, VEL, FSI, THT

**Communications**

1. **COMMON/TRANS/T1**
2. **COMMON/EQUIVALENCE**
3. **program** 03/28/73.
4. **CALL COMMON/TRANS/T1**

**Initialization**

- IRT2 = 2 * IRATIO

**Main Routine**

- CALL COMMON/TRANS/T1
00690  EFS=1.*T11+T22+T33-4.*E0*E0
00700  E0D=-.5*(E1*F+E2*G+E3*R)+EPS*E0
00710  E1D=+.5*(E0*F+E2*R-E3*Q)+EPS*E1
00720  E2D=+.5*(E0*G+E3*F-E1*R)+EPS*E2
00730  E3D=+.5*(E0*R+E1*Q-E2*P)+EPS*E3
00740  IT IS NOT NECESSARY TO REEVALUATE ALL OF THE DERIVATIVES WHEN
00750  "IRATE" EQUALS 1 IN THIS SITUATION, OTHERWISE IT EQUALS 0.
00770  WHEN "IRATE" EQUALS 1, PORTIONS OF THE LOOP ARE BYPASSED TO
00780  PREVENT RECOMPUTATION OF NON-UPDATED VARIABLES.
00790  IF (IRATE.EQ.1) GO TO 20
00800  COMPUTE TARGET POSITION.
00810  CALL TARGET
00820  COMPUTE MISSILE THRUST.
00830  CALL THRUST
00840  COMPUTE THE INSTANTANEOUS VALUES OF THE RIGID BODY PARAMETERS.
00850  CALL RBPRMT
00860  COMPUTE THE PERTINENT ATMOSPHERIC PARAMETERS.
00870  CALL ATMOS
00880  20 CONTINUE
00890  SIMULATE THE SEEKER DYNAMICS. THE SEEKER VARIABLES MUST BE
00900  INTEGRATED WITH THE SMALL TIME STEP TO AVOID COMPUTATIONAL
00910  INSTABILITIES, AND HENCE ARE EVALUATED DURING EACH PASS THROUGH
00920  THE LOOP.
00930  CALL SEEKER
00940  BYPASS PORTIONS OF THE LOOP BASED ON "IRATE".
00950  IF (IRATE.EQ.1) GO TO 30
00960  DETERMINE THE MISSILE CONTROL VARIABLES.
00970  CALL CTLSYS
00980  ADD THE EFFECTS OF ATMOSPHERIC WINDS.
00990  UT=U-WINDXB
01000  VT=V-WINDYB
01010  WT=W-WINDZB
01020  CALCULATE MISSILE VELOCITY, MACH NUMBER, AND DYNAMIC PRESSURE.
01030  VEL2=UT*UT+VT*VT+WT*WT
01040  VEL=SGRT(VEL2)
01050  VEL1=SGRT(VEL2-UT*UT)
01060  GUE=.5*RH0*VEL2
01070  RMACH=VEL/VSND
01080  DETERMINE ALPHA AND PHI OF THE WIND AT THE SINGULARITY OF
01090  ALPHA=0, PHI IS DEFINED AS 0.
01100  IF (VEL.GT.0.) FRMT=UT/VEL
01120  IF (ABS(FRMT).GE.1.0) FRMT=SIGN(CON1,FRMT)
01130  ALPHA=CON4*ACOS(FRMT)
01140  PHIW=0.
01150  SPHIW=0.
01160  CPHIW=1.
01170  IF (VEL1.EQ.0.) GO TO 10
01180  SPHIW=VT/VEL1
01190  CFHW=WT/VEL1
01200  IF (ABS(SPHIW).GE.1.0) SPHIW=SIGN(CON1,SPHIW)
01210  IF (ABS(CPHIW).GE.1.0) CPHIW=SIGN(CON1,CPHIW)
01220  PHIW=CON4*ACOS(CPHIW)
01230  IF (SPHIW.LT.0.) PHIW=-PHIW
01240  DETERMINE THE AERODYNAMIC FORCES AND MOMENTS.
01250  10 CALL AERO
01260  SUM THE AERO, THRUST, AND GRAVITY FORCES.
01270  XF=XFA+THR+T13*GRAV*RMASS
EVALUATE THE MISSILE STATE VARIABLE DERIVATIVES.

UD = \( R \times V - C \times U + XF \)/RMASS
WD = \( C \times U - I \times V + ZF \)/RMASS
PD = XM/XXI
FRMT = \( F \times (XXI - YYI) \)
CD = \(- R \times FRMT + YM) / YYI
RD = \( G \times FRMT + ZM) / YYI
XID = \( U \times T1 + V \times T21 + W \times T31 \)
YID = \( U \times T12 + V \times T22 + W \times T32 \)
HTD = \(- (U \times T13 + V \times T23 + W \times T33) \)

DETERMINE WHETHER OUTPUT IS DESIRED. "IFPRINT"=0 INDICATES THAT THE INTEGRATION ROUTINE HAS ONLY PREDICTED AND HAS NOT CORRECTED, THEREFORE THE OUTPUT IS MEANINGLESS. "IFPRINT"=1 INDICATES THAT OUTPUTING IS POSSIBLE.

IF (IFPRINT.EQ.0) GO TO 6
DETERMINE IF THE DESIRED OUTPUT INTERVAL IS SATISFIED.
IF (MOD(NSTEF,NFFINT).GT.0) GO TO 6
CONVERT THE QUATIONS TO EULER ANGLES FOR OUTPUTING.
SINGULARITY OF \( \Theta = \pm 90 \), \( FSI \) IS DEFINED AS \( =0 \) AND \( PHI \) IS CALCULATED.

IF (ABS(T13).GE.1.0) T13=SIGN(CON1,T13)
THT=CON4*ASIN(-T13)
FSI=CON4*ATAN2(T12,T11)
PHI=CON4*ATAN2(T23, T33)
GO TO 5
THETA=CON4*ATAN2(T21, T31)

KEEP TRACK OF THE NUMBER OF TIMES OUTPUTING IS PERFORMED.
ICOUNT=ICOUNT+1
CONVERT RADIANS TO DEGREES FOR OUTPUTING.
FSI20=FSI2*CON4
THT40=THT4*CON4
HI10=HI1*CON4
EFBO=EFB*CON4
FSI30=FSI3*CON4
EFBCO=EFBC*CON4
WHITE THE OUTPUT VARIABLES
WRITE(2) X,U,V,W,XI,YI,HT,F,G,R,FSI,THT,PHI,
ALPHA,FHI4,HI10,FSI20,FSI30,THT40,EPBO,EFBCO,
WYX,WZC,WGY,WGZ,FH11D,FH12D, (IJEK1(I),I=1,4)
DETERMINE WHETHER OR NOT ALL VARIABLES ARE TO BE UPDATED DURING THE NEXT PASS THROUGH THE LOOP. ALL VARIABLES ARE EVALUATED ON THE FIRST AND SECOND PASSES AND ONLY THE Z(I) VARIABLES ARE EVALUATED ON ALL SUBSEQUENT PASSES UNTIL AN "IRATIO" NUMBER OF STEPS HAVE BEEN TAKEN.
IF (K.GT.2) IRATE=1
CALL THE INTEGRATION ROUTINE.
2 CALL INTEG
STOF THE PROGRAM WHEN THE REQUIRED TIME HAS ELAPSED.
IF (X.LT.(XMAX+H*IRATIO)) GO TO 3
CALL THE PROGRAM TO SORT AND SEQUENCE THE OUTPUT VARIABLES FOR TTY COMPATIBILITY.
01870 REWIND 2
01880    CALL OUTPUT1
01890    END
SUBROUTINE INIT
COMMON Y(9),DY(9),X,H,WINDX,WINDY,WINDZ,RMACH
COMMON CH1W,XP1,Y1,Z1,TH1,XCG,RMMA,EPC,EY1,XM,YM,ZM,XXI,YYI
COMMON VSND,SFIW,HTE,CUE,ALPHA,PH1W,XMAX,TBURN,RHO,VEL,PSI,THT,
COMMON FH1,Z(14),DZ(14)
COMMON NDIM,IPRINT,NSTEP,NPRINT,ICOUNT,IRATE,IRATIO,NDIMF
COMMON INITL/XT0,YT0,HTT0,XTD,YTD,HTTD
EQUVALENCE (E0,Z(11)),(E1,Z(12)),(E2,Z(13)),(E3,Z(14))
100 100 FORMAT(*INFUT H,XMAX,NPRINT,IRATIO*)
110 110 FORMAT(*INFUT XT0,YT0,HTT0,XTD,YTD,HTTD*)
120* INITIALIZE INTEGRATION ROUTINE CONTROL VARIABLES.
130 IRATE=0
140 IPRINT=1
160* SET DIMENSIONS OF Y(I) AND Z(I) RESPECTIVELY. IF DIMENSIONS
170* ARE INCREASED ALSO INCREASE THE STORAGE LOCATION DIMENSIONS
180* SUBROUTINE INTEG.
190 NDIM=9
200 NDIMF=14
210 PRINT 100
220* READ FROM TTY THE STEP SIZE, RUN TIME, OUTPUT INTERVAL,STEP SIZE
230* RATIO.
240 READ , H,XMAX,NPRINT,IRATIO
250* READ FROM TTY THE TARGET POSITION AND VELOCITY.
260 PRINT 110
270 READ ,XT0,YT0,HTT0,XTD,YTD,HTTD
280* TIME=0.
290 NSTEF=0
300 X=0.
310* INITIALIZE ALL STATE VARIABLES TO 0.
320 DO 1 I=1,NDIM
330 Y(I)=0.
340 1 DY(I)=0.
350 DO 2 I=1,NDIMF
360 Z(I)=0.
370 2 DZ(I)=0.
380 HTE=6.
390 TBURN=4.5
400* INITIAL EULER ANGLES FOR VERTICAL ORIENTATION.
410 PSI=0.
420 THT=3.141592653/2.
430 PHI=0.
440* COMPUTE INITIAL QUATERNION VALUES FROM EULER ANGLES.
450 E0=+COS(PSI/2.)*COS(THT/2.)*COS(PHI/2.)+SIN(PSI/2.)*SIN(THT/2.)*SIN(PHI/2.)
460+ (THT/2.)*SIN(PHI/2.)
470 E1=+COS(PSI/2.)*COS(THT/2.)*SIN(PHI/2.)*SIN(PSI/2.)*SIN(THT/2.)*COS(PHI/2.)
480+ (THT/2.)*COS(PHI/2.)
490 E2=+COS(PSI/2.)*SIN(THT/2.)*COS(PHI/2.)*SIN(PSI/2.)*SIN(THT/2.)*COS(PHI/2.)
500+ (THT/2.)*SIN(PHI/2.)
510 E3=-COS(PSI/2.)*SIN(THT/2.)*COS(PHI/2.)*SIN(PSI/2.)*COS(THT/2.)*SIN(PHI/2.)
520+ (THT/2.)*COS(PHI/2.)
530 RETURN
540 END
SUBROUTINE TRNSMT

COMMON Y(9),DY(9),X,H,WINDXB,WINDYB,WINDZB,RMACH

COMMON CPHIW,XFA,YFA,ZFA,THR,XCG,RCMASS,PCAN,YCAN,XM,YM,ZM,XXI,YYI

COMMON VSND,SPHIW,HTE,GUE,ALPHA,PHIW,XMAX,TBURN,RHO,VEL,PSI,THT

COMMON PHII,Z(14),DZ(14)

COMMON NDIM,IPRINT,NSTEF,NPRINT,ICOUNT,IRATE,IRATIO,NDIMF

COMMON/TRANS/T11,T12,T13,T21,T22,T23,T31,T32,T33

EQUIVALENCE (E0,Z(11)),(E1,Z(12)),(E2,Z(13)),(E3,Z(14))

CALCULATE THE ELEMENTS OF THE TRANSFORMATION MATRIX FROM THE QUATERNION VARIABLES.

T11=E0*E0+E1*E1-E2*E2-E3*E3
T12=2.*(E1*E2+E0*E3)
T13=2.*(E1*E3-E0*E2)
T21=T12-4.*E0*E3
T22=T11-2.*(E1*E1-E2*E2)
T23=2.*(E2*E3+E0*E1)
T31=T13+4.*E0*E2
T32=T23-4.*E0*E1
T33=T11-2.*(E1*E1-E3*E3)

RETURN

END
SUBROUTINE TARGET

COMMON Y(9),DY(9),X,H,WINDXB,WINDYB,WINDZB,RMACH
+CFHIW,XFA,YFA,ZFA,THR,XCRG,RA,PCAN,YCAND,XYM,XY,ZM,XXI,YYI
+VSND,SFH,W,HTE,QUE,ALPHA,PHIW,XMAX,TBURN,RHO,VEL,PSI,THT,
+FIW,Z(4),DZ(4)
COMMON NDIM,IPRINT,NPRINT,ICOUNT,IRATE,IRATI0,NDIMF

COMMUNICATIONS WITH SUBROUTINE INIT
COMMON/IINIT/T0,T0,H0,XTD,YTD,HTD
COMMON/TRANS/T11,T12,T13,T21,T22,T23,T31,T32,T33
COMMON/SEEKR/XTI,YTI,HTTI

DATA CONS/57.29577951/

COMPUT THE INERTIAL POSITION OF THE TARGET
XTI=XT0+X*XTD
YTI=YT0+X*YTD
HTTI=HTT0+X*HTTD

COMPUTE THE TARGET POSITION RELATIVE TO THE MISSILE IN THE INERTIAL AXIS SYSTEM.
XIMT=XTI-XI
YIMT=YTI-YI
HTMT=HTTI-HT

TRANSFORM THIS INTO MISSILE BODY COORDINATES.
XMT=T11*XIMT+T12*YIMT-T13*HTMT
YMT=T21*XIMT+T22*YIMT-T23*HTMT
ZMT=T31*XIMT+T32*YIMT-T33*HTMT

XMTD=T11*XTD+T12*YTD-T13*HTTD-U*ZMT+R*YMT
YMTD=T21*XTD+T22*YTD-T23*HTTD-V*XMT+P*ZMT
ZMTD=T31*XTD+T32*YTD-T33*HTTD-W-P*YMT+Q*XMT

COMPUTE THE AZIMUTH AND ELEVATION ANGLES OF THE TARGET AS SEEN BY THE MISSILE AND THEIR RESPECTIVE RATES.
SIGA=ATAN2(ZMT*XMT)*CONS
SIGB=ATAN2(YMT,XMT)*CONS
SIGAD=CONS*(XMT*ZMTD-XMTD*ZMT)/(XMT*XMT+ZMT*ZMT)
SIGBD=CONS*(XMT*YMTD-YMTD*YMT)/(XMT*XMT+YMT*YMT)

RETURN

END
10 SUBROUTINE THRUST
20 COMMON Y(9),DY(9),X,H,WINDXB,WINDYB,WINDZB,RMACH
30+ CFHIW,XFA,YFA,ZFA,THR,XCG,RMASS,PCAN,XM,YM,ZM,XXI,YYI
40+ VSND,SPHIW,HTE,QUE,ALPHA,PHIW,XMAX,TBURN,RHO,VEL,FSI,THT,
50+ PHI,Z(14),DZ(14)
60 COMMON NDIM,IPRINT,NSTEP,NPRINT,ICOUNT,IRATE,IRATIO,NDIMF
70 THR=3000.
80 IF (X.GT.TBURN) THR=0.
90 RETURN
100 END
SUBROUTINE RBFRMT

COMMON Y(9),DY(9),X,H,WINDXB,WINDYB,WINDZB,RMACH

COMMON NDIM,NDIMF,IPRINT,NSTEP,NPRINT,ICOUNT,IRATE,IRATIO,NSTEF,NFRINT,ICOUNT,IRATE,IRATIO

DATA RMASS0/6.742/,YYI0/65.1/,XXI0/.420/,XCG0/-.321/,RMASSD/-4506/,YYID/-3.778/,XXID/-.03889/,XCGD/.16152/

IF (X.GT.TBURN) GO TO 1

IF THE MAIN ROCKET IS STILL ON, THE MISSILE'S MASS, MOMENTS

VALUE PLUS AN AVERAGE RATE OF CHANGE TIMES THE ELAPSED TIME.

RMASS=RMASS0+X*RMASSD

XXI=XXI0+X*XXID

YYI=YYI0+X*YYID

XCG=XCG0+X*XCGD

RETURN

END
SUBROUTINE ATMOS

COMMON Y(9),DY(9),X,H,WINDXB,WINDYB,WINDZB,WMACH

,CFH1W,XFA,YFA,ZFA,THR,XCG,RMAS,PCAN,YCAN,XM,YM,ZM,XXI,YYI

,VSND,SFHIW,HTE,QUE,ALPHA,PHIW,XMAX,TBURN,RHO,VEL,PSI,THT

PHI,Z(14),DZ(14)

COMMON NDIM,IFRINT,NSTEP,NPRINT,I_COUNT,IRATE,IRATIO,NDIMF

COMMON/TRANS/T11,T12,T13,T21,T22,T23,T31,T32,T33

EQUIVALENCE (HT,Y(6))

RHO=0.023769-HT*.000000688

VSND=1116.89-HT*.00384

SET THE INERTIAL COMPONENTS OF THE WIND.

WINDXI=0.

WINDYI=0.

TRANSFORM INTO MISSILE AXES.

WINDXB=WINDXI*T11+WINDYI*T12

WINDYB=WINDXI*T21+WINDYI*T22

WINDZB=WINDXI*T31+WINDYI*T32

RETURN

END
SUBROUTINE SEEKER

COMMON Y(9),DY(9),X,H,WINDXB,WINDYB,WINDZB,RMACH

COMMON/TRANS/T1,T2,T3,T13,T21,T22,T23,T31,T32,T33

COMMON/TARG/XTI,YTI,HTTI

COMMON/SEEKR/EFB,SPHI1,CHI1,CPHI

100 EQUIVALENCE (EL,Z(1)),(ER,Z(2)),(WXZ,Z(3)),(PHI1,Z(4)),
(WCZ,Z(5)),(FSIZ,Z(6)),(WGY,Z(7)),(WYZ,Z(8)),(FSIZ,Z(9)),
(THT4,Z(10)),(EDL,DZ(1)),(ERD,DZ(2)),(WYXD,DZ(3)),(PHI1,DZ
4)),(WGD,DZ(5)),(FSID,DZ(6)),(WGD,DZ(7)),(WGD,DZ(8)),

120 (FSID,DZ(9)),(THT4,DZ(10)),(FX,DZ(7)),(QY,DZ(8)),(RZ,DZ(9))

130 (FX,DY(7)),(FY,DY(8)),(RD,DY(9))

140 EQUIVALENCE (XI,Y(4)),(YI,Y(5)),(HT,Y(6))

150 DATA INIT/0/*T1FSI/.0125/*T2PSI/.0033/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5/*RKPSI/-960./*EPSIDB/2.5/*RKBPSI/.046/*RN2/-1./*RPSI/1.5*/

170 DATA XMTI=XTI-XI

300 YMTI=YTI-YI

310 ZMTI=-CHTTI-HTTI

320 RMT=SQRT(XMTI*XMTI+YMTI*YMTI+ZMTI*ZMTI)

330 RNXI=XMTI/RMT

340 RNYI=YMTI/RMT

350 RNZI=ZMTI/RMT

360 RNXI=T11*RNXI+T12*RNYI+T13*RNZI

370 RNMY=T21*RNXI+T22*RNYI+T23*RNZI

380 RNMZ=T31*RNXI+T32*RNYI+T33*RNZI

390 IF (INIT.EQ.1) GO TO 1

400 INIT=1

410 ISWSZ=0

420 SFSI2=SQRT(RNMY*RNMY+RNZI*RNZI)

430 CPSI2=ATAN(SFSI2,RNMX)

440 FHI=0.

450 IF (SFSI2*NEG.0.) PHI1=ATAN2(RNMY,RNMX)

460 CONTINUE

470 SPHI1=SN(PHI1)

480 CHI1=COS(PHI1)

490 CPSI2=SN(CHI1)

500 CPSI3=COS(CHI1)

510 CPSI3=SN(CHI1)

520 CPSI3=COS(CHI1)

530 STHT4=SN(STHT4)

540 CTHT4=COS(STHT4)

550 SPSI1=SPS12*CPSI3+CPHI2*SPS13

560 CPSI1=CPHI2*CPSI3-SPS12*SPS13

570 EBPBS1=RNMX*SPS12+RNMY*CPHI1*CPHI2+RNZI*SPI1*CPHI3

580 EFBTHI=(-RNMX*CPHI2*STHT4+RNMY*CPHI1*STHT4+RNZI*CPHI1*STHT4)

590 CONTINUE
RNCY = -RNMX*SPSI2 + RNMY*CPSI2*CFHI1 + RNMZ*CPSI2*SFHI1
RNCZ = -RNMY*SPHI1 + RNMZ*CPHI1
EFBC = SQRT(RNCY*RNCY + RNCZ*RNCZ)
EFB = SQRT(EFBSI*EFBSI + EPBHT*EPBHT)
RLAMCG = SQRT(THT4*THT4 + PSI3*PSI3)

* CALCULATION OF GAIN COMPENSATION IN ROLL AXIS DRIVE
IF (ABS(PSI2) .LT. .579) GO TO 10
VCCOMP = SIGN(1.0, PSI2)
GO TO 30
10 IF (ABS(PSI2) .LT. .174) GO TO 20
VCCOMP = SIGN(3.3, PSI2)
GO TO 30
20 VCCOMP = SIGN(10.5, PSI2)
CONTINUE
30 THRESH = .10
IF (ISWSZ .EQ. 1) THRESH = .05
IF (RLAMCG .GT. THRESH) GO TO 40
IF (ABS(PSI2) .GE. THRESH) GO TO 40
EFSR = -EPS1
GO TO 50
40 EFSR = -THT4*VCCOMP
CONTINUE
50 RPHI = EFSR / T2PHI

* ROLL AXIS DRIVE TORQUE MOTOR
ERD = (EFSR - ER) / T2PHI
EPHID = (T1PHI*ERD + ER) * RKPHI
EPHIDL = BOUND(EPHIDL, EPHID)
EPHNET = EPHIDL - PHI1D * RKPHI
RIPHI = EPHNET / RPHI
TYXM0T = RKTPHI * BOUND(RIPHI1B, RIPHI)

* OUTER LOOK AXIS DRIVE TORQUE MOTOR
EPSL = PSI3
ELD = (EPSL - EL) / T2PSI
EPSI = (T1PSI*ELD + EL) * RKPSI
EPSIDL = BOUND(EPSIDB, EPSI)
EPSNET = EPSIDL - RKPSI * RNS2 * PSI2D
RIPSI = EPSNET / RP'SI
TCZMOT = RKTPSI * BOUND(RIPSI1B, RIPSI)

* GYRO TORQUE CONTROL EQUATIONS
TGYENL = RKTT*EPBPSI - RKPTE*PSI3
TGZENL = -RKTT*EPBTHT + RKPTE*THT4
TGYE = BOUND(TGYEB, TGYENL)
TGZE = BOUND(TGZEB, TGZENL)

* SEEKER YOKE DYNAMICS
WYY = Q*CPHI1 + R*SPHI1
WYZ = -Q*SPHI1 + R*CPHI1
WYD = OD*CPHI1 + RD*SPHI1 - PHI1D*(Q*SPHI1 - R*CPHI1)
WZD = -OD*SPHI1 + RD*CPHI1 - PHI1D*(Q*CPHI1 + R*SPHI1)
TYXF1 = FRIC(T1, PHI1D)
TYX1 = TYXM0T + TYXF1
TYX = TYX2 + TYXU + TYX1
TYY = RIYY + WYD + WYZ + WXX + WYY + WXX + WYY
TYZ = RIYZ + WZD + WXX + WYY + WXX + WYY
TYY1 = TYY - TYY2 - TYYU
TYZ1 = TYZ - TYZ2 - TYZU
TMX1 = -TYX1
TMY1 = -(TYY1*CPHI1 - TYZ1*SPHI1)
TMZ1 = -(TYY1*SPHI1 + TYZ1*CPHI1)
RICC = RICX * PHI1 + RICY * SFHI1
RIYX = RIYX + RICC
WYX = (TYX - WYY * WYZ + (RIYZ - RIYY)) / RIYX
PHI1D = WYX - F

SEEKER COIL HOUSING DYNAMICS

WCX = WYX * CPSI2 + WYY * SPSI2
WCY = WYX * SFHI2 + WYY * CPSI2
TCZF2 = FRICT (F2, FSI2D)
TCZS2 = STOPS (CKSTP2, FSI2L, FSI2)
TCZZ = (TCZMOT - (RN2 - 1.0) * RIDFSI * WYZD) + RN2 + TCZF2 + TCZS2 - RKC2

FSI2

TCZ = TCZ2 + TCZU + TCZ3 + TCZE
WCZD = (TCZ - WCX * WCY + (RICY - RICX)) / (RICZ + RN2 * RN2 * RIFFSI)

PHI1D = WYX - F

SEEKER COIL HOUSING DYNAMICS

TYX2 = -(TCX2 * CPSI2 - TCY2 * SPSI2)
TYY2 = -(TCX2 * SPSI2 + TCY2 * CPSI2)
TYZ2 = -TCZ2

Gyro Dynamics

TGY = TGY5 + TGYE + TGYU
TGZ = TGZ5 + TGZE + TGZU

TGYEFF = TGY + HS * WGY - (RKHG / RIGZ) * (TGZ + HS * WGY)

TGZEFF = TGZ + HS * WGY + (RKHG / RIGY) * (TGY - HS * WGY)

WGYD = TGYEFF / (RIGY + RKHG * RIGZ)

WGZD = TGZEFF / (RIGZ + RKHG * RIGY)

Gyro Housing Dynamics

WHY = WGY

WRY = WRY

WHZ = WGR

WRX = WCX * CPSI3 + WCY * SPSI3

WRZ = (WHZ - WRX * STHT4) / CTHT4

PHI3D = WRZ - WCZ

THVT4D = WHY - WRY

TRZF3 = FRICT (F3, FSI3D)

TRZS3 = STOPS (CKSTP3, FSI3L, FSI3)

TRZ3 = TRZS3 + TRZF3

TRZ4 = -(TRZ3 + TRZU)

THYF4 = FRICT (F4, THT4D)

THYS4 = STOPS (CKSTP4, THT4L, THT4)

THX4 = -THXU

THY4 = THYF4 + THYS4

THZ4 = -(TRZ4 + THX4 * STHT4) / CTHT4

THY5 = -(THYU + THY4)

THZ5 = -(THZ4 + THZU)

TGY5 = -THY5

TGZ5 = -THZ5

TRX4 = -(THX4 * CTHT4 + THZ4 * STHT4)

TRY4 = -THY4

TRX3 = -(TRX4 + TRXU)

TRY3 = -(TRY4 + TRYU)

TCZ3 = -TRZ3

TCX3 = -(TRX3 * CPSI3 - TRY3 * SPSI3)

TCY3 = -(TRX3 * SPSI3 + TRY3 * CPSI3)

TCYE = -TGYE
1780  TCZE=-TGZE
1790  RETURN
1800  END
1810* FUNCTION ROUTINE FOR A LIMITER.
1820 FUNCTION BOUND(XL,X)
1830 BOUND=X
1840 IF (ABS(X) .GE. XL) BOUND=SIGN(XL,X)
1850 RETURN
1860 END
FUNCTION ROUTINE FOR FRICTION.

FUNCTION FRICT(FL,X)

IF (X) 1,2,3

1 FRICT=FL
RETURN

2 FRICT=0.
RETURN

3 FRICT=-FL
RETURN

END
FUNCTION ROUTINE FOR STOPS WITH COMPLIANCE.

FUNCTION STOPS(SLOPE, XT, X)

STOPS=0.

DEL=ABS(X)-XT

IF (DEL.GE.0.) STOPS=SLOPE*SIGN(DEL, X)

RETURN

END
SUBROUTINE SEEKER

1810* FUNCTION ROUTINE FOR A LIMITER.
1870* FUNCTION ROUTINE FOR FRICTION.
1970* FUNCTION ROUTINE FOR STOPS WITH COMPLIANCE.
SUBROUTINE CTLSYS

COMMON Y(9),DY(9),X,H,WINDX,WINDY,WINDZ,RMACH

CFHIW,XFA,YFA,ZFA,THR,XCG,MASS,PCAN,YCAN,XM,YM,ZM,XXI,YYI

COMMON SPHIW,HTE,CUE,ALPHA,PHIW,XMAX,TBURN,RHO,VEL,PSI,THT

COMMON NDIM,IPRINT,NSTEP,NPRINT,ICOUNT,IRATE,IRATIO,NDIMF

COMMON/CTLSYS/ IJETC4)

COMMON/SEEKR/EFB,SPHI1,CFHI1,EPBC

COMMUNICATIONS WITH SUBROUTINE TARGET

EQUIVALENCE (HT,Y(6)),(PSI,Z(6)),(PSI2,DZ(6))

EQUIVALENCE (PHI1,D(4))

DATA TBJRC/1.2/

DATA CON1/.18/

ENABLE THE CONTROL SYSTEM AT THE ENABLE HEIGHT.

IF (HT-HTE) 1,2,2

1 IJRC=0

TSJRC=X

GO TO 3

2 IJRC=1

DETERMINE WHETHER THE JRC BURN TIME HAS BEEN EXCEEDED, AND IF

SO, DISENABLE THE JRC'S.

IF ((X-TSJRC)*STJRC) IJRC=0

3 IJRC 4,4,5

IJET(1)=0

IJET(2)=0

IJET(3)=0

IJET(4)=0

GO TO 6

DETERMINE THE JRC STATES BASED ON THE CONTROL LAW.

PRMTY CONTROLS THE JETS ON THE Y AXIS WHILE PRMTZ CONTROLS

THE JETS ON THE Z AXIS. "IJET(1)"=1 SIGNIFIES THAT JET 1 IS ON.

IJET(1)=1

IJET(3)=0

GO TO 10

IJET(1)=0

IJET(3)=1

10 IF (PRMTY) 11,12,13

IJET(2)=1

IJET(4)=0

GO TO 6

IJET(2)=0

IJET(4)=1

CONTINUE

THE CANARD DEFLECTIONS ARE SET TO ZERO

PCAN=0.

YCAN=0.

RETURN

END
10 SUBROUTINE AERO
20 COMMON Y(9),DY(9),X,H,WINDXB,WINDYB,WINDZB,RMACH
30+ ,CFHW,XFA,YFA,ZFA,THR,XCG,RMASS,PCAN,YCAN,XM,YM,ZM,XXI,YYI
40+ ,VSND,SPHIW,HTE,CUE,ALPHA,PHIW,XMAX,TBURN,RHO,VEL,FSI,THT,
50+ PHI,Z(14),DZ(14)
60 COMMON NDIM,IPRINT,NSTEP,NPRINT,ICOUNT,IRATE,IRATIO,NDIMF
70* COMMUNICATION WITH AROBSE
80 COMMON/AROB/XFB,YFB,ZFB,XMB,YMB,ZMB
90* COMMUNICATION WITH AROJRC
100 COMMON/AROJ/YFJ,ZFJ,YMJ,ZMJ
110* DETERMINE BASELINE AERODYNAMIC FORCES AND MOMENTS.
120 CALL AROBSE
130* DETERMINE FORCES AND MOMENTS DUE TO THE JRC.
140 CALL AROJRC
150* SUM THE FORCES.
160 XFA=XFB
170 YFA=YFB+YFJ
180 ZFA=ZFB+ZFJ
190* SUM THE MOMENTS AND TRANSFER REFERENCE POINT TO THE
200* MISSILE CG.
210 XM=XMB
220 YM=YMB+YMJ+ZFA*XCG
230 ZM=ZMB+ZMJ-YFA*XCG
240 RETURN
250 END
SUBROUTINE AROBSE

COMMON Y (9), DY (9), XH, WINDB, WINDY, WINDZ, RMACH
COMMON CFHIV, XFA, YFA, ZFA, THR, XCG, RMAS, FCAN, YCAN, XM, YM, ZM, XXI, YYI
COMMON VSND, SPHIW, HTE, CUE, ALFA, PHiW, XMAX, TBURN, RHO, VEL, FSI, THT,
COMMON FHI, Z(14), DZ(14)
COMMON NDI, FPRINT, NSTEP, NPRINT, ICOUNT, IRATE, IRATIO, NDIMF
COMMON /AER01/ DFHI, DALPHA
COMMON /AROB/ XFB, YFB, ZFB, XMB, YMB, ZMB
DIMENSION CXB(25,3), CYFB(25,3), CZPB(25,3), CLB(25,3),
     CFHB(25,3), CNPB(25,3)

DATA CXB/75*0.0/
DATA CYFB/75*0.0/
DATA CZFB/75*0.0/
DATA CLB/75*0.0/
DATA CFHB/75*0.0/
DATA CNPB/75*0.0/
DATA CON1/45.0/, CON2/2.5/, CON3/90.0/
DATA DIA/.4166666667/, S/.1363538470/

FHIWT=AMOD(ABS(PHIW), CON3)
FHIWTE=FHIWT/CON1+1.0
IF=PHIWT
DFHI=PHIWT-IF
ALP=ALPHA/CON2+1.0
IA=ALPHA
IF (IA>GE.25) IA=24
DALPHA=ALPHA-IA
CALL INTER2(CXB (IA, IF), CXB (IA+1, IF), CXB (IA, IF+1),
     CXP (IA+1, IF+1), CXB )
CALL INTER2(CYFB(IA, IF), CYFB(IA+1, IF), CYFB(IA, IF+1),
     CYF(IA+1, IF+1), CYFB )
CALL INTER2(CZFB(IA, IF), CZFB(IA+1, IF), CZFB(IA, IF+1),
     CZF(IA+1, IF+1), CZFB )
CALL INTER2(CLB (IA, IF), CLB (IA+1, IF), CLB (IA+1, IF+1),
     CMF (IA+1, IF+1), CMF )
CALL INTER2(CNF(IA, IF), CNF(IA+1, IF), CNF(IA, IF+1),
     CNF(IA+1, IF+1), CNF )
CYP=CYP*CFHIV+CZF*SPHIW
CZ=-CYP*SFHIV+CZP*CPHIU
CM=CMF*CFHIV+CNF*SPHIW
CN=-CMF*SFHIW+CNF*CFHIW
XFB=GUE*S*CX
YFB=GUE*S*CY
ZFB=GUE*S*CZ
XMB=GUE*S*DIA*CL
YMB=GUE*S*DIA*CM
ZMB=GUE*S*DIA*CN
RETURN
END
SUBROUTINE AROJRC

COMMON Y(9), DX(9), XH, WINDXB, WINDYB, WINDZB, RMACH
COMMON YFA, YFB, YFC, WTH, XCR, XAMM, FC, YCAN, XMY, YZM, ZXI, YI
COMMON USND, SHW, HTE, CUE, ALPHA, PHIB, XMI, MAX, TBR, RHO, VEL, PSI, THT
COMMON Z(14), DZ(14)
COMMON NDIM, IPRINT, NSTEP, NPRINT, ICONT, IRATE, IRATIO, NDIMF
COMMON/AERO/ DP, DPH, DALPHA
COMMON/CTLSYJ/IJETC4)
COMMON/AFOJ/YFJ, ZFJ, YMJ, ZMJ

DIMENSION RKB(25,3), RKB(25), RKB(25,3), RKB(25,3)
DIMENSION CPE(4), SPE(4)
DATA RKYB/75.0/
DATA RKZB/
DATA RKMB/
DATA RKNB/
DATA TJRC/450.0/
DATA CON1/90.0/
DATA FHIE1/0.0/
DATA CPE/1.0/
DATA SPE/0.0/
ALPHAE=ALPHA/CON2+1.0
IA=ALPHAE
IF (IA.GE.25) IA=24
DALPHA=ALPHAE-IA
RKY=0.
RKZ=0.
RKM=0.
HKN=0.
DO 1 I=1,4
IF (IJET(I).EQ.0) GO TO 1
FHIE=FHIE-FHIE1-CON1*(I-1.0)
CF=CPE(I)
SF=SPE(I)
IF (ABS(FHIE).GT.180.) FHIE=FHIE-SIGN(360.,FHIE)
FHJTE=FHJTE/CON1+1.0
IF=FHJTE
IF (IF.GE.3) IF=2
DFHI=FHJTE-IF
CALL INTER2(RKB(IA,IF), RKB(IA+1,IF), RKB(IA,IF+1),
RKB(IA+1,IF+1)), RKYJ
CALL INTER2(RKZB(IA,IF), RKB(IA+1,IF), RKB(IA,IF+1),
RKB(IA+1,IF+1)), RKNJ
CALL INTER2(RKMB(IA,IF), RKB(IA+1,IF), RKB(IA,IF+1),
RKB(IA+1,IF+1), RKNJ)
CALL INTER2(RKB(IA,IF), RKB(IA+1,IF), RKB(IA,IF+1),
RKB(IA+1,IF+1)), RKNJ
IF (FHIE.GE.0.0) GO TO 2
RKY=-RKYJ+RKYJ*CF+RKZJ*SP
RKZ=-RKZJ*RKYJ*SP+RKZJ*CF

48
590  RKM=RKM+RKM*CP +RKNJ*SP
600  RKN=RKN-RKM*SF +RKNJ*CP
610  1 CONTINUE
620  YFJ=RKY*TJRC
630  ZFJ=RKZ*TJRC
640  YMJ=RKM*TJRC*XJET
650  ZMJ=RKN*TJRC*XJET
660  RETURN
670  END
SUBROUTINE OUTFUT1
COMMON Y(9),DY(9),X,H,WINDXB,WINDYB,WINDZB,RMACH
COMMON CYC,YCYC,Z(14),DYC,DYCYC,X*H,WINDXB*UI,NDYB*WINDZB*
RMACH*CPHIV*XFA*YFA*ZFA*THR*XCG*RMASS*PCAN*YCAN*XM*YM*ZM*XXI*YYI
COMMON NDIM,IPRINT,NSTEP,NPRINT,ICOUNT,I RATE,IRATIO,NDIMF
DIMENSION OUT(40,40),IOUT(40,4)
100 FORMAT(F6.3*6F8.2)
101 FORMAT(F6.3*3F8.2*3F7.2)
102 FORMAT(F6.3*8F7.2)
103 FORMAT(//2X**TIME**5X**U**7X**V**7X**W**7X**XI**6X**YI**6X**HT**)
104 FORMAT(//2X**TIME**5X**U**3X**V**3X**W**3X**XI**3X**YI**3X**HT**)
105 FORMAT(//2X**TIME**2X**U**3X**V**3X**W**3X**XI**3X**YI**3X**HT**)
106 FORMAT(F6.3*4(5X,11))
107 FORMAT(//**STEP SIZE=**F6.4**IRATIO=**I2)
108 FORMAT(//**STEP SIZE=**F6.4**IRATIO=**I2)
109 FORMAT(//**STEP SIZE=**F6.4**IRATIO=**I2)

READ ALL THE OUTFUTED VARIABLES.
READ(2,*) (OUT(I,J),J=1,27),(IOUT(I,J),J=1,4)
OUTFUT THESE VARIABLES TO THE TTY IN APPROPRIATE COLUMNS WITH HEADINGS.

PRINT 103
DO 1 I=1,ICOUNT
1 PRINT 100, (OUT(I,J),J=1,7)
PRINT 104
DO 2 I=1,ICOUNT
2 PRINT 101, OUT(I,1), (OUT(I,J),J=8,13)
PRINT 105
DO 3 I=1,ICOUNT
3 PRINT 102, OUT(I,1), (OUT(I,J),J=14,21)
PRINT 106
DO 5 I=1,ICOUNT
5 PRINT 102, OUT(I,1), (OUT(I,J),J=22,27)
PRINT 107
DO 4 I=1,ICOUNT
4 PRINT 106, OUT(I,1), (IOUT(I,J),J=1,4)
PRINT 200, H,IRATIO
STOP
END
SUBROUTINE INTEG
COMMON Y(9),DY(9),X,H,WINDX,WINDY,WINDZ,RMACH
CPIH,XFA,YFA,ZFA,THR,XCG,MRM,A,CAN,PCAN,XM,YM,ZM,XXI,YYI,
USND,SPHIW,HTE,CUE,ALPHA,PIHW,XMAX,TBURN,RHO,VEL,FSI,THT,
FHI,Z(14),DZ(14)
COMMON NDIM,IPRINT,NSTEP,PRINT,IFUNC,IRET,IRATIO,NDIMF
* STORAGE LOCATIONS FOR THE REQUIRED FAST STATE VARIABLES AND
* THEIR DERIVATIVES. "AUX" FOR Y(I) AND "BUX" FOR Z(I).
DIMENSION AUX(1),AUX2(1),AUX3(1),AUX4(1),AUX5(1),AUX6(1),
AUX7(1),AUX8(1),BUX(14),BUX2(14),BUX3(14),
BUX4(14),BUX5(14),BUX6(14),BUX7(14),BUX8(14)
DATA IENTR/1/,IENTRF/2/
IF (IRATE.EQ.1) GO TO 100
DECIDE WHETHER TO 1) PREDICT REQUIRED STARTING VALUES BY
AN EULER BACKSTEP METHOD, 2) PERFORM PREDICTION OF THE Y(I)
VARIABLES DURING INTEGRATION, OR 3) CORRECT THE Y(I) VARIABLES
DURING INTEGRATION.
IF (IENTR-2) 20,21,22
DETERMINE REQUIRED STARTING VALUES USING AN EULER BACKSTEP, AND
STORE INTO APPROPRIATE LOCATIONS.
20 HS=IRATIO*H
H1=3.*HS
H2=4./3.*HS
G1=3.*H
G2=4./3.*H
Cl=112./121.
C2=9./121.
DO 16 I=1,NDIM
AUX2(I)=Y(I)-HS*DY(I)
AUX3(I)=Y(I)-2.*HS*DY(I)
AUX4(I)=Y(I)-3.*HS*DY(I)
AUX6(I)=DY(I)
AUX7(I)=DY(I)
16 AUX8(I)=0.
DO 26 I=1,NDIMF
BUX2(I)=Z(I)-H*DZ(I)
BUX3(I)=Z(I)-2.*H*DZ(I)
BUX4(I)=Z(I)-3.*H*DZ(I)
BUX6(I)=DZ(I)
BUX7(I)=DZ(I)
26 BUX8(I)=0.
PREDICT THE Y(I) VARIABLES.
21 DO 17 I=1,NDIM
AUX1(I)=Y(I)
AUX5(I)=DY(I)
DELT=AUX4(I)+H3*(AUX5(I)+AUX5(I)-AUX6(I)+AUX7(I)+AUX7(I))
Y(I)=DELT-C1*AUX8(I)
17 AUX8(I)=DELT
100 IENTR=3
GO TO 100
CORRECT THE Y(I) VARIABLES.
22 DO 18 I=1,NDIM
DELT=1.25*(9.*AUX1(I)-AUX3(I)+H1*(DY(I)+AUX5(I)+AUX5(I))
-AUX6(I)))
AUX8(I)=AUX8(I)-DELT
18 Y(I)=DELT+C2*AUX8(I)
100 IENTR=3
GO TO 100
AUX4(I) = AUX3(I)
AUX3(I) = AUX2(I)
AUX2(I) = AUX1(I)
IENTR = 2

NOW TO REPEAT FOR THE Z(I) VARIABLES.

DETERMINE WHETHER TO PREDICT OR CORRECT THE Z(I) VARIABLES.

100 IF (IENTR.EQ.3) GO TO 32

PREDICT THE Z(I) VARIABLES.

DO 27 I = 1, NDIMF
  BUX1(I) = Z(I)
  BUX5(I) = DZ(I)

DELT = BUX4(I) + G3*(BUX5(I) + BUX5(I) - BUX6(I) + BUX7(I) + BUX7(I))
Z(I) = DELT - C1*BUX8(I)
DO 27 BUX8(I) = DELT

UPDATE THE VARIABLE TIME BY "H" AFTER EACH PREDICTION.

NSTEP = NSTEP + 1
X = NSTEP*H
IFRINT = 0
IENTRF = 3
RETURN

CORRECT THE Z(I) VARIABLES.

DO 28 I = 1, NDIMF
  DELT = .125*C9.1*BUX1(I) - BUX3(I) + G1*(DZ(I) + BUX5(I) + BUX5(I)) - BUX6(I))
  BUX6(I) = BUX8(I) - DELT
  Z(I) = DELT + C2*BUX8(I)
DO 28 BUX8(I) = DELT

RETURN

END
SUBROUTINE INTER2(C00, C10, C01, C11, CB)

20* PERFORMS A TWO DIMENSIONAL INTERPOLATION BETWEEN THE FOUR CORNERS OF THE SQUARE C00-C11 AND RETURNS THE ANSWER BY CB.

40* COMMUNICATIONS WITH SUBROUTINE AERO WHICH CALLS INTER2.

50* DPHI AND DALPHA ARE THE INTERPOLATION INCREMENTS.

60 COMMON/AERO1/DPHI, DALPHA

70 C1=C00+DALPHA*(C10-C00)

80 C2=C01+DALPHA*(C11-C01)

90 CB=C1+DPHI*(C2-C1)

100 RETURN

110 END
FORTRAN, OLD, SXDF

FORTRAN, OLD, SXDF

READY.

RUN, M = 11000

03/28/73  17:05:04

PROGRAM SXDF

INFUT H, XMAX, NPRINT, IRATIO

? .0005, 2, 200, 10

INFUT XT0, YT0, HTT0, XTD, YTD, HTTD

? 9000, 3000, 100, 0, 0, 0

<table>
<thead>
<tr>
<th>TIME</th>
<th>U</th>
<th>V</th>
<th>W</th>
<th>XI</th>
<th>YI</th>
<th>HT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.</td>
<td>0.</td>
<td>0.</td>
<td>0.</td>
<td>0.</td>
<td>0.</td>
<td>0.</td>
</tr>
<tr>
<td>0.00</td>
<td>41.41</td>
<td>0.</td>
<td>0.00</td>
<td>0.00</td>
<td>0.</td>
<td>2.07</td>
</tr>
<tr>
<td>0.20</td>
<td>63.14</td>
<td>1.29</td>
<td>1.29</td>
<td>0.02</td>
<td>0.02</td>
<td>8.29</td>
</tr>
<tr>
<td>0.30</td>
<td>124.61</td>
<td>-9.71</td>
<td>-7.74</td>
<td>6.1</td>
<td>5.55</td>
<td>18.68</td>
</tr>
<tr>
<td>0.40</td>
<td>157.77</td>
<td>-18.84</td>
<td>-47.56</td>
<td>2.57</td>
<td>1.36</td>
<td>33.11</td>
</tr>
<tr>
<td>0.50</td>
<td>166.45</td>
<td>-20.10</td>
<td>-114.34</td>
<td>7.15</td>
<td>2.96</td>
<td>50.89</td>
</tr>
<tr>
<td>0.60</td>
<td>175.92</td>
<td>-36.26</td>
<td>-150.15</td>
<td>15.35</td>
<td>5.63</td>
<td>71.00</td>
</tr>
<tr>
<td>0.70</td>
<td>196.31</td>
<td>-47.08</td>
<td>-166.24</td>
<td>27.98</td>
<td>9.39</td>
<td>91.98</td>
</tr>
<tr>
<td>0.80</td>
<td>227.44</td>
<td>-52.29</td>
<td>-169.56</td>
<td>45.23</td>
<td>14.44</td>
<td>112.76</td>
</tr>
<tr>
<td>0.90</td>
<td>266.06</td>
<td>-51.83</td>
<td>-162.85</td>
<td>67.10</td>
<td>20.97</td>
<td>132.50</td>
</tr>
<tr>
<td>1.00</td>
<td>300.77</td>
<td>-49.27</td>
<td>-150.62</td>
<td>93.59</td>
<td>29.04</td>
<td>150.66</td>
</tr>
<tr>
<td>1.10</td>
<td>353.54</td>
<td>-45.26</td>
<td>-136.15</td>
<td>124.68</td>
<td>38.69</td>
<td>166.91</td>
</tr>
<tr>
<td>1.20</td>
<td>400.89</td>
<td>-39.66</td>
<td>-118.95</td>
<td>160.36</td>
<td>49.92</td>
<td>181.18</td>
</tr>
<tr>
<td>1.30</td>
<td>449.40</td>
<td>-34.79</td>
<td>-100.35</td>
<td>200.65</td>
<td>62.74</td>
<td>193.53</td>
</tr>
<tr>
<td>1.40</td>
<td>498.01</td>
<td>-30.25</td>
<td>-86.66</td>
<td>245.59</td>
<td>77.18</td>
<td>204.17</td>
</tr>
<tr>
<td>1.50</td>
<td>552.44</td>
<td>-14.23</td>
<td>-36.64</td>
<td>295.19</td>
<td>93.24</td>
<td>213.61</td>
</tr>
<tr>
<td>1.60</td>
<td>601.57</td>
<td>11.12</td>
<td>37.54</td>
<td>349.48</td>
<td>110.67</td>
<td>222.88</td>
</tr>
<tr>
<td>1.70</td>
<td>645.79</td>
<td>27.02</td>
<td>80.41</td>
<td>408.31</td>
<td>129.98</td>
<td>233.29</td>
</tr>
<tr>
<td>1.80</td>
<td>698.37</td>
<td>16.60</td>
<td>45.12</td>
<td>471.48</td>
<td>150.43</td>
<td>245.71</td>
</tr>
<tr>
<td>1.90</td>
<td>749.20</td>
<td>-10.08</td>
<td>-34.29</td>
<td>539.20</td>
<td>172.30</td>
<td>259.58</td>
</tr>
<tr>
<td>2.00</td>
<td>796.43</td>
<td>-23.79</td>
<td>-76.56</td>
<td>611.76</td>
<td>195.74</td>
<td>273.45</td>
</tr>
<tr>
<td>TIME</td>
<td>P</td>
<td>C</td>
<td>R</td>
<td>PSI1</td>
<td>PSI2</td>
<td>THT1</td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>90.00</td>
</tr>
<tr>
<td>0.100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>90.00</td>
</tr>
<tr>
<td>0.200</td>
<td>0</td>
<td>-0.52</td>
<td>0.52</td>
<td>45.00</td>
<td>89.32</td>
<td>45.00</td>
</tr>
<tr>
<td>0.300</td>
<td>0</td>
<td>-2.43</td>
<td>1.31</td>
<td>41.34</td>
<td>77.76</td>
<td>41.40</td>
</tr>
<tr>
<td>0.400</td>
<td>0</td>
<td>-4.24</td>
<td>-1.16</td>
<td>18.61</td>
<td>59.74</td>
<td>20.22</td>
</tr>
<tr>
<td>0.500</td>
<td>0</td>
<td>-3.25</td>
<td>1.12</td>
<td>11.62</td>
<td>35.64</td>
<td>14.61</td>
</tr>
<tr>
<td>0.600</td>
<td>0</td>
<td>-1.80</td>
<td>1.03</td>
<td>14.15</td>
<td>20.65</td>
<td>15.77</td>
</tr>
<tr>
<td>0.700</td>
<td>0</td>
<td>-1.00</td>
<td>0.64</td>
<td>16.17</td>
<td>11.83</td>
<td>16.32</td>
</tr>
<tr>
<td>0.800</td>
<td>0</td>
<td>-5.35</td>
<td>0.19</td>
<td>17.03</td>
<td>6.90</td>
<td>16.46</td>
</tr>
<tr>
<td>0.900</td>
<td>0</td>
<td>-2.55</td>
<td>0.33</td>
<td>2.55</td>
<td>16.54</td>
<td>0</td>
</tr>
<tr>
<td>1.000</td>
<td>0</td>
<td>-2.33</td>
<td>0.03</td>
<td>17.90</td>
<td>2.55</td>
<td>16.54</td>
</tr>
<tr>
<td>1.100</td>
<td>0</td>
<td>-1.15</td>
<td>0.11</td>
<td>16.13</td>
<td>1.45</td>
<td>16.55</td>
</tr>
<tr>
<td>1.200</td>
<td>0</td>
<td>-1.10</td>
<td>0.09</td>
<td>18.23</td>
<td>1.12</td>
<td>16.55</td>
</tr>
<tr>
<td>1.300</td>
<td>0</td>
<td>-0.99</td>
<td>0.07</td>
<td>18.40</td>
<td>1.03</td>
<td>16.55</td>
</tr>
<tr>
<td>1.400</td>
<td>0</td>
<td>-0.66</td>
<td>0.16</td>
<td>18.39</td>
<td>0.92</td>
<td>16.55</td>
</tr>
<tr>
<td>1.500</td>
<td>0</td>
<td>-0.39</td>
<td>0.18</td>
<td>18.30</td>
<td>5.77</td>
<td>16.54</td>
</tr>
<tr>
<td>1.600</td>
<td>0</td>
<td>-0.40</td>
<td>17.99</td>
<td>13.22</td>
<td>16.49</td>
<td>0</td>
</tr>
<tr>
<td>1.700</td>
<td>0</td>
<td>-0.09</td>
<td>17.68</td>
<td>17.58</td>
<td>16.41</td>
<td>0</td>
</tr>
<tr>
<td>1.800</td>
<td>0</td>
<td>-0.98</td>
<td>0.31</td>
<td>17.66</td>
<td>14.66</td>
<td>16.40</td>
</tr>
<tr>
<td>1.900</td>
<td>0</td>
<td>-1.07</td>
<td>0.37</td>
<td>17.90</td>
<td>7.83</td>
<td>16.45</td>
</tr>
<tr>
<td>2.000</td>
<td>0</td>
<td>-0.02</td>
<td>0.01</td>
<td>18.12</td>
<td>4.35</td>
<td>16.47</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TIME</th>
<th>ALPHA</th>
<th>PHIW</th>
<th>PHI1</th>
<th>PSI2</th>
<th>PSI3</th>
<th>THT4</th>
<th>EFBC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>71.57</td>
<td>89.40</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0.100</td>
<td>0</td>
<td>0</td>
<td>71.54</td>
<td>89.28</td>
<td>0.13</td>
<td>0</td>
<td>0.03</td>
</tr>
<tr>
<td>0.200</td>
<td>1.25</td>
<td>45.00</td>
<td>71.57</td>
<td>88.83</td>
<td>0.02</td>
<td>0</td>
<td>0.05</td>
</tr>
<tr>
<td>0.300</td>
<td>5.69</td>
<td>128.55</td>
<td>71.36</td>
<td>78.33</td>
<td>-0.10</td>
<td>-0.17</td>
<td>0</td>
</tr>
<tr>
<td>0.400</td>
<td>17.96</td>
<td>158.39</td>
<td>70.92</td>
<td>59.40</td>
<td>-0.10</td>
<td>-0.03</td>
<td>0.03</td>
</tr>
<tr>
<td>0.500</td>
<td>34.90</td>
<td>170.03</td>
<td>64.24</td>
<td>36.08</td>
<td>-0.16</td>
<td>-0.32</td>
<td>0</td>
</tr>
<tr>
<td>0.600</td>
<td>41.28</td>
<td>166.42</td>
<td>62.46</td>
<td>21.07</td>
<td>-0.18</td>
<td>-0.09</td>
<td>0</td>
</tr>
<tr>
<td>0.700</td>
<td>41.39</td>
<td>163.93</td>
<td>62.35</td>
<td>12.18</td>
<td>-0.19</td>
<td>-0.10</td>
<td>0.02</td>
</tr>
<tr>
<td>0.800</td>
<td>37.96</td>
<td>162.86</td>
<td>62.43</td>
<td>7.30</td>
<td>-0.20</td>
<td>-0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>0.900</td>
<td>32.71</td>
<td>162.35</td>
<td>61.75</td>
<td>4.64</td>
<td>-0.16</td>
<td>-0.04</td>
<td>0.03</td>
</tr>
<tr>
<td>1.000</td>
<td>27.17</td>
<td>161.89</td>
<td>62.99</td>
<td>3.10</td>
<td>-0.17</td>
<td>-0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>1.100</td>
<td>22.09</td>
<td>161.61</td>
<td>62.93</td>
<td>2.03</td>
<td>-0.16</td>
<td>-0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>1.200</td>
<td>17.37</td>
<td>161.56</td>
<td>62.97</td>
<td>1.59</td>
<td>-0.06</td>
<td>-0.05</td>
<td>0.04</td>
</tr>
<tr>
<td>1.300</td>
<td>13.30</td>
<td>160.88</td>
<td>63.81</td>
<td>1.67</td>
<td>-0.10</td>
<td>-0.26</td>
<td>0.03</td>
</tr>
<tr>
<td>1.400</td>
<td>10.44</td>
<td>160.75</td>
<td>63.05</td>
<td>1.51</td>
<td>-0.06</td>
<td>-0.19</td>
<td>0.03</td>
</tr>
<tr>
<td>1.500</td>
<td>4.07</td>
<td>158.77</td>
<td>63.43</td>
<td>6.23</td>
<td>-0.20</td>
<td>-0.97</td>
<td>0.02</td>
</tr>
<tr>
<td>1.600</td>
<td>3.72</td>
<td>16.50</td>
<td>71.49</td>
<td>13.76</td>
<td>-0.23</td>
<td>-0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>1.700</td>
<td>7.48</td>
<td>18.57</td>
<td>71.23</td>
<td>18.22</td>
<td>-0.21</td>
<td>-0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>1.800</td>
<td>3.94</td>
<td>20.20</td>
<td>70.60</td>
<td>15.76</td>
<td>-0.21</td>
<td>-0.02</td>
<td>0.02</td>
</tr>
<tr>
<td>1.900</td>
<td>2.73</td>
<td>163.62</td>
<td>69.92</td>
<td>9.06</td>
<td>-0.17</td>
<td>-0.04</td>
<td>0.03</td>
</tr>
<tr>
<td>2.000</td>
<td>5.34</td>
<td>161.37</td>
<td>69.50</td>
<td>5.64</td>
<td>-0.15</td>
<td>-0.00</td>
<td>0.03</td>
</tr>
</tbody>
</table>

55
<table>
<thead>
<tr>
<th>TIME</th>
<th>WYOKE</th>
<th>WCOIL</th>
<th>WGY</th>
<th>WGZ</th>
<th>PHI1D</th>
<th>PSI2D</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>1.00</td>
<td>-0.02</td>
<td>0.03</td>
<td>0.02</td>
<td>0.03</td>
<td>0.02</td>
<td>0.03</td>
</tr>
<tr>
<td>2.00</td>
<td>-0.00</td>
<td>0.19</td>
<td>-0.06</td>
<td>-0.01</td>
<td>0.00</td>
<td>-0.48</td>
</tr>
<tr>
<td>3.00</td>
<td>0.10</td>
<td>-0.02</td>
<td>0.01</td>
<td>0.02</td>
<td>0.10</td>
<td>-2.73</td>
</tr>
<tr>
<td>4.00</td>
<td>-0.81</td>
<td>0.06</td>
<td>-0.03</td>
<td>0.02</td>
<td>-0.81</td>
<td>-3.90</td>
</tr>
<tr>
<td>5.00</td>
<td>-0.70</td>
<td>0.02</td>
<td>0.01</td>
<td>0.04</td>
<td>-0.70</td>
<td>-3.39</td>
</tr>
<tr>
<td>6.00</td>
<td>-0.03</td>
<td>0.03</td>
<td>0.01</td>
<td>0.03</td>
<td>-0.03</td>
<td>-2.04</td>
</tr>
<tr>
<td>7.00</td>
<td>0.20</td>
<td>-0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>0.20</td>
<td>-1.20</td>
</tr>
<tr>
<td>8.00</td>
<td>-0.03</td>
<td>-0.01</td>
<td>-0.04</td>
<td>-0.02</td>
<td>-0.03</td>
<td>-0.57</td>
</tr>
<tr>
<td>9.00</td>
<td>0.21</td>
<td>0.06</td>
<td>-1.00</td>
<td>0.04</td>
<td>0.21</td>
<td>-2.77</td>
</tr>
<tr>
<td>10.00</td>
<td>-0.16</td>
<td>0.03</td>
<td>-0.09</td>
<td>0.00</td>
<td>-0.16</td>
<td>0.19</td>
</tr>
<tr>
<td>11.00</td>
<td>-0.01</td>
<td>0.00</td>
<td>0.01</td>
<td>0.01</td>
<td>-0.01</td>
<td>-0.18</td>
</tr>
<tr>
<td>12.00</td>
<td>-0.01</td>
<td>-0.04</td>
<td>-0.04</td>
<td>-0.01</td>
<td>-0.01</td>
<td>0.02</td>
</tr>
<tr>
<td>13.00</td>
<td>-0.01</td>
<td>-0.01</td>
<td>-0.02</td>
<td>0.01</td>
<td>-0.01</td>
<td>-0.13</td>
</tr>
<tr>
<td>14.00</td>
<td>-0.01</td>
<td>-0.07</td>
<td>-0.04</td>
<td>0.03</td>
<td>-0.01</td>
<td>-0.20</td>
</tr>
<tr>
<td>15.00</td>
<td>1.60</td>
<td>0.00</td>
<td>-0.01</td>
<td>-0.03</td>
<td>1.60</td>
<td>1.22</td>
</tr>
<tr>
<td>16.00</td>
<td>-0.08</td>
<td>-0.02</td>
<td>-0.03</td>
<td>0.05</td>
<td>-0.08</td>
<td>1.20</td>
</tr>
<tr>
<td>17.00</td>
<td>-0.03</td>
<td>-0.00</td>
<td>-0.04</td>
<td>-0.02</td>
<td>-0.03</td>
<td>-0.19</td>
</tr>
<tr>
<td>18.00</td>
<td>-0.07</td>
<td>0.02</td>
<td>-0.06</td>
<td>0.01</td>
<td>-0.07</td>
<td>-1.01</td>
</tr>
<tr>
<td>19.00</td>
<td>-0.54</td>
<td>0.05</td>
<td>-0.03</td>
<td>0.05</td>
<td>-0.54</td>
<td>-1.08</td>
</tr>
<tr>
<td>20.00</td>
<td>-0.14</td>
<td>-0.07</td>
<td>-0.03</td>
<td>0.05</td>
<td>-0.14</td>
<td>-0.05</td>
</tr>
</tbody>
</table>

**JET STATES.** 1=ON, 0=OFF

<table>
<thead>
<tr>
<th>TIME</th>
<th>JET STATES</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>1.00</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>2.00</td>
<td>0 0 1 1</td>
</tr>
<tr>
<td>3.00</td>
<td>0 1 1 0</td>
</tr>
<tr>
<td>4.00</td>
<td>0 0 1 1</td>
</tr>
<tr>
<td>5.00</td>
<td>1 0 0 1</td>
</tr>
<tr>
<td>6.00</td>
<td>0 1 1 0</td>
</tr>
<tr>
<td>7.00</td>
<td>0 1 1 0</td>
</tr>
<tr>
<td>8.00</td>
<td>0 0 1 1</td>
</tr>
<tr>
<td>9.00</td>
<td>0 1 1 0</td>
</tr>
<tr>
<td>10.00</td>
<td>0 0 1 1</td>
</tr>
<tr>
<td>11.00</td>
<td>0 1 1 0</td>
</tr>
<tr>
<td>12.00</td>
<td>0 1 1 0</td>
</tr>
<tr>
<td>13.00</td>
<td>0 0 1 0</td>
</tr>
<tr>
<td>14.00</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>15.00</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>16.00</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>17.00</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>18.00</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>19.00</td>
<td>0 0 0 0</td>
</tr>
<tr>
<td>20.00</td>
<td>0 0 0 0</td>
</tr>
</tbody>
</table>

**STEP SIZE= .0005**  **IRATIO=10**  **STOP.**
APPENDIX B
PROGRAM VARIABLES

The following is an alphabetical listing of the computer program variables except for the variables in subroutine SEEKER.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALPHA</td>
<td>Missile angle of attack</td>
</tr>
<tr>
<td>ALPHAE</td>
<td>Angle of attack entry argument for coefficient table lookup</td>
</tr>
<tr>
<td>AUX1</td>
<td>Storage locations for integration routine used for HS.</td>
</tr>
<tr>
<td>:</td>
<td>$Y_i, Y_{i-1}, Y_{i-2}, Y_{i-3}, DY_i, DY_{i-1}, DY_{i-2}$, truncation corrector respectively</td>
</tr>
<tr>
<td>AUX8</td>
<td></td>
</tr>
<tr>
<td>BUX1</td>
<td>Storage locations for integration routine used for H.</td>
</tr>
<tr>
<td>:</td>
<td>$Z_i, Z_{i-1}, Z_{i-2}, Z_{i-3}, DZ_i, DZ_{i-1}, DZ_{i-2}$, truncation corrector respectively</td>
</tr>
<tr>
<td>BUX8</td>
<td></td>
</tr>
<tr>
<td>CL, C2</td>
<td>Intermediate answers for 2-D coefficient lookup</td>
</tr>
<tr>
<td>COO, CLO</td>
<td>Entry points for 2-D coefficient lookup</td>
</tr>
<tr>
<td>COL, C1</td>
<td></td>
</tr>
<tr>
<td>CB</td>
<td>Coefficient value return by 2-D coefficient lookup</td>
</tr>
<tr>
<td>CL</td>
<td>Aerodynamic total rolling moment coefficient in body fixed axes</td>
</tr>
<tr>
<td>CM</td>
<td>Aerodynamic total pitching moment coefficient in body fixed axes</td>
</tr>
<tr>
<td>CMP</td>
<td>Aerodynamic total pitching moment coefficient in pitch axes</td>
</tr>
<tr>
<td>CMPB</td>
<td>Array of pitching moment coefficient for baseline missile (no JRC) vs $\alpha$ and $\phi_w$</td>
</tr>
<tr>
<td>CN</td>
<td>Aerodynamic total yawing moment coefficient in body fixed axes</td>
</tr>
<tr>
<td>CNP</td>
<td>Aerodynamic total yawing moment coefficient in pitch axes</td>
</tr>
<tr>
<td>CNPB</td>
<td>Array of yawing moment coefficient for baseline missile (no JRC) vs $\alpha$ and $\phi_w$</td>
</tr>
<tr>
<td>CON1</td>
<td>Program constants</td>
</tr>
<tr>
<td>:</td>
<td></td>
</tr>
<tr>
<td>CON4</td>
<td></td>
</tr>
<tr>
<td>CP</td>
<td>Dummy variable</td>
</tr>
<tr>
<td>CPE</td>
<td>Vector of cosines of $\phi_E$ for each engine</td>
</tr>
<tr>
<td>CPHIL</td>
<td>$\cos(\phi_1)$</td>
</tr>
<tr>
<td>CPHIW</td>
<td>$\cos(\phi_w)$</td>
</tr>
<tr>
<td>CX</td>
<td>Aerodynamic total X force coefficient in body fixed axes</td>
</tr>
<tr>
<td>CXB</td>
<td>Array of X force coefficient for baseline missile (no JRC) vs $\alpha$ and $\phi_w$</td>
</tr>
<tr>
<td>CY</td>
<td>Aerodynamic total Y force coefficient in pitch axes</td>
</tr>
</tbody>
</table>
CYP  Aerodynamic total Y force coefficient in body fixed axes
CYPB  Array of Y force coefficients for baseline missile (no JRC) vs $\alpha$ and $\phi_w$
CZ  Aerodynamic total Z force coefficient in body fixed axes
CZP  Aerodynamic total Z force coefficient in pitch
CZPB  Array of Z force coefficients for baseline missile (no JRC) vs $\alpha$ and $\phi_w$
DALPHA  The residue of $\alpha$ required for 2-D coefficient lookup
DELT  Intermediate variable in integration routine
DIA  Missile diameter
DPHI  The residue of $\phi_w$ required for 2-D coefficient lookup
DY  Vector of derivatives of Y
DZ  Vector of derivatives of Z
EO  Quaterion variables
:  
E3  
EOD  Derivatives of E0-E3
:  
E3D  
EPB  Angular error between gyro spin axis and target line of sight in radians
EPBC  Angular error between coil housing axis and target line of sight in radians
EPBCO  EPBC in degrees for outputting
EPS  Error due to non-orthogonality of E0-E3
GL, G3  Internal constants for integration routine
GRAV  Gravity in ft/sec^2
HL, H3  Internal constants for integration routine
H  Step size for fast integration routine
HS  Step size for slow integration routine
HT  Altitude of missile
HTD  Derivative of HT
HTMT  Altitude difference between missile and target
HTTO  Initial height of target
HTTD  Target climb rate
HTTI  Altitude of target
IA  Entry location on $\alpha$ for 2-D coefficient lookup
ICOUNT
Number of times outputting is performed
IENTR
1--Calculate required starting values, 2--Predict Y(I) for integration, 3--Correct Y(I) for integration
IENTRF
2--Predict Z(I) for integration, 3--Correct Z(I) for integration
IJET
Array for each jet; 0--off, 1--on
IOUT
Array for outputting integer variables
IP
Entry location on $\xi_w$ for 2-D coefficient lookup
IPRINT
0--No outputting permitted, 1--Outputting permitted
IRATE
Controls integration routine; 0--Integrate both Y and Z, 1--Integrate only Z
IRATIO
HS/H
IRT2
2* IRATIO
NDIM
Dimension of Y(I)
NDIMF
Dimension of Z(I)
NPRINT
Outputting interval = NPRINT*H
NSTEP
Counter for number of H integration steps
OUT
Array for outputting real variables
P
Missile roll rate in body axes in rad/sec
PCAN
Pitch canard deflection
PD
Derivative of P
PHI
$\phi$ in degrees
PHI1
$\phi_1$ in radians
PHI0
PHI1 in degrees for outputting
PHID
Derivative of PHI1
PHIE1
$\phi_E$ for thruster number 1
PHIJ
$\phi_J$ for thruster in degrees
PHIW
$\xi_w$ in degrees
PHIWT
$\xi_w$ reduced to a range of 0 - 90 degrees
PHIWTE
$\xi_w$ entry argument for coefficient table lookup
PRMT
Dummy variable
PRMTY
Control function for JRC's aligned with missile Y axis
PRMTZ
Control function for JRC's aligned with missile Z axis
PSI
$\psi$ in degrees
PSI2
$\psi_2$ in radians
PSI20
$\psi_2$ in degrees for outputting
PSI2D
Derivative of PSI2
PSI3
$\psi_3$ in radians
PSI30 \( \psi_3 \) in degrees for outputting
PSI3D Derivative of PSI3
Q Missile pitch rate in body axes in rad/sec
QD Derivative of Q
QUE Dynamic pressure \( \frac{1}{2} \rho V^2 \) in lbs/ft²
R Missile yaw rate in body axes in rad/sec
RD Derivative of R
RHO \( \rho \) in slugs/ft³
RKM Total moment amplification factor along missile Y axis
RKMB Array of moment amplification factor vs \( \alpha \) and \( \delta_J \)
RKN Total moment amplification factor along missile Z axis
RKNB Array of moment amplification factor vs \( \alpha \) and \( \delta_J \)
RKY Total force amplification factor along missile Y axis
RKYB Array of forces amplification factor vs \( \alpha \) and \( \delta_J \)
RKZ Total force amplification factor along missile Z axis
RKZB Array of forces amplification factor vs \( \alpha \) and \( \delta_J \)
RMACH Mach number of missile
RMASS Instantaneous missile mass
RMASSO Initial missile mass
RMASSD Derivative or RMASS
S Missile reference area = \( \frac{\pi d^2}{4} \)
SIGA \( \sigma_A \) in degrees
SIGAD Derivative of SIGA
SIGB \( \sigma_B \) in degrees
SIGBD Derivative of SIGB
SP Dummy variable
SPS Vector of sines of \( \xi_E \) for each engine
SPHIL \( \sin(\xi_L) \)
SPHILW \( \sin(\xi_W) \)
T11 Elements of matrix transformation from body to inertial axes
T33
TBJRC Burn time of JRC's
TBURN Burn time of missile main thruster
THT \( \theta \) in degrees
THT4 \( \theta_4 \) in radians
THR: Main thrustur thrust
THT40: $\theta_4$ in degrees for outputting
THT4D: Derivative of THT4
TJRRC: Thrust of one JRC
TSJRC: Enable time for JRC's
U: U velocity of missile
UD: Derivative of U
UT: UT
V: V velocity of missile
VD: Derivative of V
VEL: Total missile velocity
VEL1: Dummy variable
VEL2: Dummy variable
VSND: Velocity of sound
VT: VT
W: W velocity of missile
WCZ: $\Omega$ of seeker cage along Z axis
WGY: $\Omega$ of seeker gyro along Y axis
WGZ: $\Omega$ of seeker gyro along Z axis
WINEXB: Components of wind along missile body axes
WINEXI: Components of wind along inertial axes
WINEXZI: Components of wind along inertial axes
WT: WT
X: Time
XCG: Position of missile C.G. relative to aerodynamic reference point
XCGO: Initial XCG
XCGD: Derivative of XCG
XF: Total force along X axis
XFA: Total aerodynamic force along X axis
XFB: Total aerodynamic force along X axis for baseline missile (no JRC)
XI: Inertial X position of missile
XID: Derivative of XI
XIMT: Inertial X position of target relative to missile
XJET: Position of JRC's from aerodynamic reference point
XM: Total moment along X axis
XMAX: Total run time
XMB: Total moment along X axis for baseline missile (no JRC)
XMT: X position of target relative to missile in body axes
XMTD: Derivative of XMT
XT0: Initial XTI
XTD: Derivative of XTI
XTI: Inertial X position of target
XXI: \( L_{XX} \)
XXIO: Initial XXI
XXID: Derivative of XXI
Y: Vector of slow integration state variables
YCAN: Deflection of yaw canard
YF: Total along Y axis
YFA: Total aerodynamic force along Y axis
YFB: Total aerodynamic force along Y axis for baseline missile (no JRC)
YFJ: Total force along Y axis due to JRC's
YI: Inertial Y position of missile
YID: Derivative of YI
YIMT: Inertial Y position of target relative to missile
YM: Total moment along Y axis
YMB: Total moment along Y axis for baseline missile (no JRC)
YMJ: Total moment along Y axis due to JRC's
YMT: Y position of target relative to missile in body axes
YMTD: Derivative of YMT
YTO: Initial YTI
YTD: Derivative of YTI
YTI: Inertial Y position of target
YYI: \( L_{YY} \)
YYIO: Initial YYI
YYID: Derivative of YYI
Z: Vector of fast integration state variables
ZF: Total along Z axis
ZFA: Total aerodynamic force along Z axis
ZFB  Total aerodynamic force along Z axis for baseline missile (no JRC)
ZFJ  Total force along Z axis due to JRC's
ZM   Total moment along Z axis
ZMB  Total moment along Z axis for baseline missile (no JRC)
ZMJ  Total moment along Z axis due to JRC's
ZMT  Z position of target relative to missile in body axes
ZMTD Derivative of ZMT
<table>
<thead>
<tr>
<th>No.</th>
<th>Distribution List</th>
<th>No. of Copies</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Defense Documentation Center</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>Cameron Station</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Alexandria, Virginia 22314</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Library</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Naval Postgraduate School</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Monterey, California 93940</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Dean of Research</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Naval Postgraduate School</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Monterey, California 93940</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>Chairman, Department of Aeronautics</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Naval Postgraduate School</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Monterey, California 93940</td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>Dr. W. H. Clark</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>Code 30102</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Naval Weapons Center</td>
<td></td>
</tr>
<tr>
<td></td>
<td>China Lake, California 9355</td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td>Assistant Professor Michael H. Redlin</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Department of Aeronautics</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Naval Postgraduate School</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Monterey, California 93940</td>
<td></td>
</tr>
</tbody>
</table>
A Point Defense Missile Simulation has been developed. This report describes the concept of such a missile, the basic features of the simulation program including the integration routine and the jet reaction controllers, and provides a FORTRAN coded source program.
<table>
<thead>
<tr>
<th>Key Words</th>
<th>Link A</th>
<th>Link B</th>
<th>Link C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Point Defense Missile</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Missile Simulation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Missile Guidance</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Advanced Seeker Simulation</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>