NASA TECHNICAL MEMORANDUM

(NASA-TM-78279) USER'S GUIDE FOR SKYLAB DYNAMICS PROGRAM, SKYDYN (NASA) 73 p
HC A04/MF A01 CSCL 09B

N80-26056

Unclas
G3/61 23533

USERS GUIDE FOR SKYLAB DYNAMICS PROGRAM, SKYDYN

M. S. Hopkins
Systems Dynamics Laboratory

May 1980

NASA

George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO. NASA TM-78279	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
4. TITLE AND SUBTITLE Users Guide for Skylab Dynamics Program, SKYDYN		5. REPORT DATE May, 1980
		6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) M. S. Hopkins		8. PERFORMING ORGANIZATION REPORT #
9. PERFORMING ORGANIZATION NAME AND ADDRESS George C. Marshall Space Flight Center Marshall Space Flight Center, Alabama 35812		10. WORK UNIT NO.
		11. CONTRACT OR GRANT NO.
12. SPONSORING AGENCY NAME AND ADDRESS National Aeronautics and Space Administration Washington, D. C. 20546		13. TYPE OF REPORT & PERIOD COVERED Technical Memorandum
		14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

Prepared by Systems Dynamics Laboratory,
Science and Engineering

16. ABSTRACT

The Skylab Dynamics Program (SKYDYN) is an extensively modified version of the 6-degree-of-freedom digital program REENTR, developed by Northrop Services, Inc., Huntsville, AL. The program REENTR was modified for the Honeywell CP-V System and was tailored to the specific requirements for Skylab.

This user's manual provides a description of the capabilities of SKYDYN, the required input data and the resulting program output.

17. KEY WORDS	18. DISTRIBUTION STATEMENT Unclassified-Unlimited

19. SECURITY CLASSIF. (of this report) Unclassified	20. SECURITY CLASSIF. (of this page) Unclassified	21. NO. OF PAGES 73	22. PRICE NTIS

TABLE OF CONTENTS

PRECEDING PAGE BLANK NOT FILMED

LIST OF ILLUSTRATIONS

TECHNICAL MEMORANDUM

USERS GUIDE FOR SKYLAB DYNAMICS PROGRAM, SKYDYN

I. INTRODUCTION

The program SKYDYN was developed to simulate the orbital dynamics of an uncontrolled asymmetric vehicle subjected to perturbing torques due to gravity gradient and aerodynamic forces. The program utilizes an oblate rotating Earth model and a variable step size, five-pass Runge-Kutta integration scheme. Quaternions are used to represent the attitude of the vehicle; thus, there are no restrictions on attitude or small angle motion.

The versatility of the program output allows the user to specify the output parameters without reprogramming. Tape output can be used for plotting or as input to other programs for open-loop calculations that would otherwise increase the run time for the dynamics program.

The input data requirements and a sample data input listing are given in Section II. Program output specifications and a sample program printout are given in Section III.

Appendix A presents definitions of the program variables. The reference frames and corresponding transformation are given in Appendix B. The equations of motion used in the simulation are presented in Appendix C. A listing of the main routine, SKYDYN, and all subroutines used are given in Appendix D.

II. INPUT DATA REQUIREMENTS

A. Data Input

Table 1 defines the variables required as input to the program SKYDYN. A change in code number indicates the beginning of a new line of input. Each variable specified to be printed and/or saved on tape (see Code 2) begins a new line of input with a maximum of 90 lines per run.

All angular data are input in degrees and converted to radians at the start of each simulation. A more detailed description of the table input for density and aerodynamic coefficients is presented following the list of input data requirements. The transformation matrix from principal to body axes (Code 25) is used for open-loop calculations only, and if data are unavailable, dummy variables may be used as input.

All data listed in Table 1 are required for initialization of the program. If it is desired to restart the simulation with initial conditions saved from a previous run (IOPT1=1), card input data for Codes 23 and 24, the initial start time and integration step size are ignored. For multiple cases (NCASE > 1, Code 1), data cards Codes 17 through 25 are repeated. Card input data are assigned to Unit 5. Tape input (see IOPT1, Code 19) is assigned to Unit 14.

A typical data input listing is given in Section II. D.

TABLE 1. SKYDYN INPUT

Code	Program Symbol	Variable Definition	Units	Format
1	NCASE	Case number	Unitless	I5 (Col 1-5)
	NX	Number of integration variables	Unitless	I5 (Col 6-10)
2	IVAR	Location of variable in common to be printed and/or saved on tape	Unitless	I3 (Col 1-3)
	ISCAL	Scale factor designation ISCAL=0, Scale factor=1.; ISCAL=1, Scale factor= 57.29578; ISCAL≠0 or ≠1, Scale factor supplied by user.	Unitless	I1 (Col 5)

TABLE 1. (Continued)

Code	Program Symbol	Variable Definition	Units	Format
	SCAL	Scale factor	User Specified	E12.5 (Col 7-18)
	PNAME	Name to be assigned to printed variables	Unitless	A4 (Col 19-22)
	IPW	Print and/or save designation: =0, print and save; =1, print only; =2 save only	Unitless	I1 (Col 23)
3		Blank card to signify end of output variables		
4	ATNM(I) I=1,18	Atmosphere Title Card	Unitless	18A4
5	TLAT(I) I=1,11	Latitude table for density lookup	deg	7F8.3
6	TLNG(I) I=1,37	Longitude table for density lookup	deg	7F8.3
7	FRHO(I,J) I=1,11 J=1,37	Atmospheric density	kg/m^3	6E9.3
8	NALP	Number of total angle-of-attack values	Unitless	I5 (Col 1-5)
	NPHIA	Number of aerodynamic roll angle values	Unitless	I5 (Col 6-10)
9	TALP(I) I=1,NALP	Total angle-of-attack table for aerodynamic data lookup	deg	7F8.3
10	TPHIA(I) I=1,NPHIA	Aerodynamic roll angle table for aerodynamic data lookup	deg	7F8.3
11	FCA(I,J) I=1,NPHIA J=1,NALP	Axial force coefficient	Unitless	7F8.3
12	FCN(I,J) I=1,NPHIA J=1,NALP	Normal force coefficient	Unitless	7F8.3
13	FCY(I,J) I=1,NPHIA J=1,NALP	Side force coefficient	Unitless	7F8.3

TABLE 1. (Contin ,ed)

Code	Program Symbol	Variable Definition	Units	Format
14	FCM(I,J) I=1,NPHIA J=1,NALP	Pitching moment coefficient	Unitless	7F8.3
15	FCEN(I,J) I=1,NPHIA J=1,NALP	Yawing moment coefficient	Unitless	7F8.3
16	FCL(I,J) I=1,NPHIA J=1,NALP	Rolling moment coefficient	Unitless	7F8.3
17	CASE(I) I=1,18	Case title card	Unitless	18A4
18	PROPT	Multiplier for print interval (Print interval=PROPT *DTP)	Unitless	F10.1
19	DT	Initial integration time step	sec	F10.4
	DTP	Output frequency for save tape	sec	F10.4
	DTSAM	Specified time to save variables for restart	sec	F10.4
	TRUN	Total run time	sec	F10.4
	TIME	Initial start time	sec	F10.4
	IOPT1	Initialization option =0, use initial conditions from data pack =1, read in initial conditions from tape	Unitless	I5
20	WT	Vehicle weight	lb	F10.4
	XCG	Vehicle cg in x-direction	ft	F10.4
	YCG	Vehicle cg in y-direction	ft	F10.4
	ZCG	Vehicle cg in z-direction	ft	F10.4
	XMRP	Aerodynamic moment reference point in x-direction	ft	F10.4
	YMRP	Aerodynamic moment reference point in Y-direction	ft	F10.4
	ZMRP	Aerodynamic moment reference point in z-direction	ft	F10.4

TABLE 1. (Continued)

Code	Program Symbol	Variable Definition	Units	Format
21	DREF	Aerodynamic reference diameter	ft	F10.4
	SREF	Aerodynamic reference area	ft^2	F10.4
22	IXYZ(1,1)	Moment of inertia about x-axis	slugs-ft^2	F10.1
	IXYZ(2,2)	Moment of inertia about y-axis	slugs-ft^2	F10.1
	IXYZ(3,3)	Moment of inertia about z-axis	slugs-ft^2	F10.1
	IXYZ(1,2)	xy product of inertia	slugs-ft^2	F10.1
	IXYZ(1,3)	xz product of inertia	slugs-ft^2	F10.1
	IXYZ(2,3)	yz product of inertia	slugs-ft^2	F10.1
23	PSI	Geocentric latitude (positive north and negative south of equator)	deg	F10.6
	LAMDE	Earth fixed longitude (positive east and negative west of Greenwich)	deg	F10.5
	RMAG	Radius vector magnitude	ft	F10.2
	VIMAG	Inertial velocity	ft/sec	F10.3
	SIGI	Inertial heading (positive clockwise from north)	deg	F10.6
	GAMI	Inertial flight path angle (positive up from local geocentric horizontal)	deg	F10.8
24	PHIBI	Initial bank angle	deg	F10.5
	ALPHAI	Initial total angle-of-attack	deg	F10.5
	PHIAI	Initial aerodynamic roll angle	deg	F10.5

TABLE 1. (Continued)

Code	Program Symbol	Variable Definition	Units	Format
	PQR(I) I=1,3	Initial rates about x, y, and z body axes, respectively	deg/sec	3F10.5
25	ABP(I,J) J=1,3 I=1,3	Transformation matrix from principal axes to body axes	Unitless	610.5
26		Blank card to signify termination of input data		

B. Density Table Input

The density table input is a bivariate function of latitude (ψ) and longitude (λ). The density values were determined for a specified altitude using the Jacchia 1970-3 atmosphere model and predicted solar and geomagnetic data provided by Space Sciences Laboratory, Marshall Space Flight Center.

The dependent density values are input in the following manner:

$$\rho(\psi_1, \lambda_1)\ \rho(\psi_2, \lambda_1) \cdots\cdots \rho(\psi_6, \lambda_1)$$

$$\rho(\psi_7, \lambda_1) \cdots\cdots \rho(\psi_{11}, \lambda_1)$$

$$\vdots \qquad\qquad\qquad \vdots$$

$$\rho(\psi_1, \lambda_{37})\ \rho(\psi_2, \lambda_{37}) \cdots\cdots \rho(\psi_6, \lambda_{37})$$

$$\rho(\psi_7, \lambda_{37}) \cdots\cdots \rho(\psi_{11}, \lambda_{37})$$

C. Aerodynamic Coefficients Table Input

The aerodynamic coefficients are bivariate functions of aerodynamic roll angle (ϕ_α) and total angle-of-attack (α_T). The dimension of these tables is chosen by the user with the restriction that all six coefficients must have the same dimension.

For the sample case presented, the coefficients are dimensioned 21 × 11 and are input in the following manner:

$$
\begin{array}{lcl}
C(\phi_{\alpha 1}, \alpha_{T1}) & \cdots\cdots\cdots\cdots & C(\phi_{\alpha 7}, \alpha_{T1}) \\
C(\phi_{\alpha 8}, \alpha_{T1}) & \cdots\cdots\cdots\cdots & C(\phi_{\alpha 14}, \alpha_{T1}) \\
C(\phi_{\alpha 15}, \alpha_{T1}) & \cdots\cdots\cdots\cdots & C(\phi_{\alpha 21}, \alpha_{T1}) \\
\vdots & & \vdots \\
C(\phi_{\alpha 1}, \alpha_{T11}) & \cdots\cdots\cdots\cdots & C(\phi_{\alpha 7}, \alpha_{T11}) \\
\vdots & & \vdots \\
C(\phi_{\alpha 15}, \alpha_{T11}) & \cdots\cdots\cdots\cdots & C(\phi_{\alpha 21}, \alpha_{T11})
\end{array}
$$

D. Sample Data Input Listing

```
    1    13
067               THOR0
008               QSQR1
151 1             PHIB0
152 1             ALPT0
153 1             PHIA0
503               VREL1
002               DELT1
329 1             GAMR1
328 1             SIGR1
517               VELX1
518               VELY1
519               VELZ1
061 1             P   0
021 1             PDOT1
526               GGB10
431               TOAX0
532               TABX0
410               CAR10
062 1             Q   0
022 1             QDOT1
527               GGB20
432               TOAY0
533               TABY0
411               CAR21
063 1             R   0
023 1             RDOT1
528               GGB30
433               TOAZ0
534               TABZ0
412               CAR31
154               BR110
155               BR210
156               BR310
157               BR120
158               BR220
159               BR320
160               BR130
161               BR230
162               BR330
181               IB111
182               IB211
183               IB311
184               IB121
185               IB221
186               IB321
187               IB131
188               IB231
189               IB331
501               RHO 1
```

```
502                QBAR1
323 1              LAT 1
326 1              LONI1
331 1              GAMI1
330 1              SIGI1

 SSL DENSITIES,F(LAT,LONG),230NM,MEASURED
-50.      -40.      -30.      -20.      -10.       0.        10.
 20.       30.       40.       50.
 0.        10.       20.       30.       40.       50.       60.
 70.       80.       90.       100.      110.      120.      130.
 140.      150.      160.      170.      180.      190.      200.
 210.      220.      230.      240.      250.      260.      270.
 280.      290.      300.      310.      320.      330.      340.
 350.      360.
  .270E-11 .296E-11 .321E-11 .345E-11 .365E-11 .381E-11
  .392E-11 .397E-11 .396E-11 .388E-11 .376E-11
  .268E-11 .293E-11 .318E-11 .341E-11 .361E-11 .377E-11
  .388E-11 .393E-11 .392E-11 .385E-11 .373E-11
  .263E-11 .287E-11 .311E-11 .333E-11 .352E-11 .367E-11
  .378E-11 .383E-11 .383E-11 .377E-11 .366E-11
  .255E-11 .277E-11 .299E-11 .319E-11 .337E-11 .352E-11
  .362E-11 .367E-11 .368E-11 .364E-11 .355E-11
  .245E-11 .265E-11 .284E-11 .302E-11 .319E-11 .332E-11
  .342E-11 .348E-11 .349E-11 .347E-11 .341E-11
  .234E-11 .250E-11 .267E-11 .283E-11 .298E-11 .310E-11
  .319E-11 .325E-11 .328E-11 .328E-11 .325E-11
  .221E-11 .235E-11 .249E-11 .263E-11 .275E-11 .286E-11
  .295E-11 .301E-11 .306E-11 .308E-11 .307E-11
  .209E-11 .220E-11 .232E-11 .243E-11 .253E-11 .262E-11
  .271E-11 .278E-11 .283E-11 .287E-11 .290E-11
  .197E-11 .205E-11 .214E-11 .223E-11 .232E-11 .240E-11
  .248E-11 .255E-11 .262E-11 .268E-11 .273E-11
  .186E-11 .192E-11 .198E-11 .205E-11 .212E-11 .219E-11
  .226E-11 .234E-11 .241E-11 .249E-11 .257E-11
  .176E-11 .179E-11 .184E-11 .189E-11 .195E-11 .201E-11
  .207E-11 .215E-11 .223E-11 .233E-11 .243E-11
  .167E-11 .168E-11 .172E-11 .175E-11 .179E-11 .184E-11
  .191E-11 .198E-11 .208E-11 .218E-11 .230E-11
  .159E-11 .159E-11 .161E-11 .164E-11 .167E-11 .171E-11
  .177E-11 .185E-11 .194E-11 .206E-11 .220E-11
  .153E-11 .152E-11 .153E-11 .154E-11 .157E-11 .160E-11
  .166E-11 .174E-11 .184E-11 .196E-11 .211E-11
  .148E-11 .146E-11 .146E-11 .147E-11 .149E-11 .152E-11
  .157E-11 .165E-11 .175E-11 .188E-11 .204E-11
  .145E-11 .142E-11 .141E-11 .142E-11 .143E-11 .146E-11
  .151E-11 .159E-11 .169E-11 .183E-11 .199E-11
  .142E-11 .139E-11 .138E-11 .136E-11 .139E-11 .142E-11
  .147E-11 .155E-11 .165E-11 .179E-11 .196E-11
  .141E-11 .138E-11 .136E-11 .136E-11 .137E-11 .140E-11
  .145E-11 .153E-11 .163E-11 .177E-11 .194E-11
  .141E-11 .137E-11 .136E-11 .136E-11 .136E-11 .139E-11
```

ORIGINAL PAGE IS
OF POOR QUALITY

.144E-11	.152E-11	.162E-11	.176E-11	.193E-11	
.140E-11	.137E-11	.136E-11	.135E-11	.136E-11	.139E-11
.144E-11	.151E-11	.162E-11	.176E-11	.193E-11	
.140E-11	.137E-11	.136E-11	.136E-11	.136E-11	.139E-11
.144E-11	.152E-11	.162E-11	.176E-11	.193E-11	
.141E-11	.138E-11	.136E-11	.136E-11	.137E-11	.140E-11
.145E-11	.152E-11	.163E-11	.177E-11	.194E-11	
.142E-11	.139E-11	.138E-11	.138E-11	.139E-11	.142E-11
.147E-11	.154E-11	.165E-11	.179E-11	.195E-11	
.145E-11	.142E-11	.141E-11	.142E-11	.143E-11	.146E-11
.151E-11	.159E-11	.169E-11	.183E-11	.199E-11	
.149E-11	.147E-11	.147E-11	.148E-11	.150E-11	.153E-11
.158E-11	.166E-11	.176E-11	.189E-11	.205E-11	
.155E-11	.154E-11	.155E-11	.157E-11	.159E-11	.163E-11
.169E-11	.177E-11	.187E-11	.199E-11	.213E-11	
.163E-11	.164E-11	.166E-11	.169E-11	.173E-11	.178E-11
.184E-11	.191E-11	.201E-11	.212E-11	.225E-11	
.173E-11	.176E-11	.181E-11	.185E-11	.190E-11	.196E-11
.203E-11	.210E-11	.219E-11	.229E-11	.240E-11	
.185E-11	.191E-11	.198E-11	.205E-11	.212E-11	.218E-11
.226E-11	.233E-11	.241E-11	.249E-11	.257E-11	
.199E-11	.208E-11	.218E-11	.227E-11	.236E-11	.2[illegible]4E-11
.252E-11	.259E-11	.266E-11	.271E-11	.276E-11	
.214E-11	.226E-11	.239E-11	.251E-11	.262E-11	.272E-11
.280E-11	.287E-11	.292E-11	.296E-11	.297E-11	
.229E-11	.244E-11	.260E-11	.275E-11	.288E-11	.300E-11
.309E-11	.315E-11	.319E-11	.320E-11	.318E-11	
.242E-11	.261E-11	.280E-11	.298E-11	.313E-11	.326E-11
.336E-11	.342E-11	.344E-11	.342E-11	.337E-11	
.254E-11	.276E-11	.297E-11	.317E-11	.335E-11	.349E-11
.359E-11	.365E-11	.366E-11	.362E-11	.353E-11	
.263E-11	.287E-11	.310E-11	.332E-11	.351E-11	.367E-11
.377E-11	.383E-11	.382E-11	.376E-11	.365E-11	
.268E-11	.294E-11	.318E-11	.341E-11	.362E-11	.377E-11
.388E-11	.393E-11	.392E-11	.385E-11	.373E-11	
.270E-11	.296E-11	.321E-11	.345E-11	.365E-11	.381E-11
.392E-11	.397E-11	.396E-11	.388E-11	.376E-11	

11	21					
0.0	20.0	40.0	60.0	80.0	90.0	100.0
120.0	140.0	160.0	180.0			
0.0	20.0	40.0	50.0	80.0	90.0	100.0
120.0	140.0	160.0	180.0	200.0	220.0	240.0
260.0	270.0	280.0	300.0	320.0	340.0	360.0
2.401	2.401	2.401	2.401	2.401	2.401	2.401
2.401	2.401	2.401	2.401	2.401	2.401	2.401
2.401	2.401	2.401	2.401	2.401	2.401	2.401
3.166	3.129	2.964	2.787	2.600	2.707	2.517
2.952	3.340	3.630	3.860	3.554	3.212	2.798
2.370	2.625	2.576	2.842	2.945	3.099	3.166
4.124	3.919	3.330	3.036	2.669	2.687	2.710
3.522	4.202	4.713	4.873	4.672	3.909	3.063
2.480	2.628	2.706	3.286	3.699	4.145	4.124

3.366	3.251	2.989	2.560	2.110	1.988	2.154
2.854	-3.423	3.857	-4.017	-3.840	-3.187	-2.473
1.980	1.988	2.164	2.828	3.187	3.407	3.366
1.433	1.347	1.220	.997	.812	.741	.832
1.122	1.315	1.471	1.518	1.436	1.224	.976
.786	.763	.847	1.125	1.290	1.371	1.433
.001	-.004	-.001	.003	.004	.004	.004
.004	.004	.005	.005	.005	.004	.005
.005	.005	.005	.004	-.001	-.003	.001
-1.529	-1.445	-1.226	-.993	-.801	-.730	-.827
-1.128	-1.317	-1.399	-1.446	-1.359	-1.225	-.974
-.779	-.754	-.839	-1.130	-1.323	-1.492	-1.529
-4.142	-3.905	-3.332	-2.725	-2.111	-1.988	-2.186
-2.943	-3.298	-3.566	-3.515	-3.321	-2.767	-2.419
-1.999	-2.049	-2.204	-2.987	-3.535	-3.971	-4.142
-5.136	-4.899	-4.366	-3.591	-2.760	-2.742	-2.810
-3.513	-3.981	-4.308	-4.308	-3.882	-3.255	-2.962
-2.595	-2.829	-2.868	-3.769	-4.533	-4.956	-5.136
-4.106	-3.966	-3.491	-3.201	-2.700	-2.813	-2.630
-3.044	-3.209	-3.313	-3.344	-3.246	-3.029	-2.901
-2.556	-2.858	-2.760	-3.299	-3.646	-4.007	-4.106
-2.590	-2.590	-2.590	-2.590	-2.590	-2.590	-2.590
-2.590	-2.590	-2.590	-2.590	-2.590	-2.590	-2.590
-2.590	-2.590	-2.590	-2.590	-2.590	-2.590	-2.590
.002	.002	.002	.002	.002	.002	.002
.002	.002	.002	.002	.002	.002	.002
.002	.002	.002	.002	.002	.002	.002
1.314	1.225	.960	.609	.227	-.001	-.224
-.663	-1.096	-1.434	-1.618	-1.404	-1.058	-.632
-.212	.006	.235	.624	.951	1.208	1.314
3.770	3.372	2.366	1.419	.468	-.001	-.472
-1.655	-2.953	-4.022	-4.416	-3.989	-2.759	-1.456
-.440	.010	.493	1.548	2.625	3.566	3.770
6.235	5.657	4.279	2.415	.733	-.001	-.751
-2.691	-4.853	-6.658	-7.368	-6.629	-4.532	-2.357
-.702	.009	.780	2.678	4.558	5.923	6.235
8.578	7.626	5.647	3.026	.903	.000	-.933
-3.422	-6.029	-8.215	-8.985	-8.015	-5.615	-2.993
-.893	.004	.976	3.436	5.971	7.762	8.578
9.330	8.237	5.874	3.115	.922	.000	-.961
-3.578	-6.320	-8.339	-9.236	-8.203	-5.902	-3.149
-.935	.001	1.009	3.584	6.291	8.460	9.330
9.153	8.132	5.659	3.040	.900	.000	-.938
-3.465	-6.069	-7.853	-8.640	-7.625	-5.655	-3.020
-.900	.003	.981	3.479	6.114	8.393	9.153
7.605	6.757	4.746	2.578	.734	.000	-.768
-2.778	-4.661	-6.146	-6.438	-5.715	-3.928	-2.312
-.717	.007	.800	2.838	5.037	6.867	7.605
4.644	4.181	3.073	1.695	.491	.000	-.500
-1.643	-2.777	-3.668	-3.886	-3.295	-2.281	-1.406
-.468	.006	.526	1.785	3.189	4.224	4.644
1.703	1.552	1.143	.716	.248	-.001	-.246

-.678	-1.041	-1.298	-1.390	-1.271	-.982	-.649
-.239	.002	.257	.739	1.189	1.566	1.703
.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000
.000	.000	.000	.000	.000	.000	.000
-.002	-.002	-.002	-.002	-.002	-.002	-.002
-.002	-.002	-.002	-.002	-.002	-.002	-.002
-.002	-.002	-.002	-.002	-.002	-.002	-.002
.000	-.408	-.724	-.929	-.994	-1.048	-.966
-.985	-.021	-.473	-.001	.460	.779	.920
.898	1.005	.977	.943	.720	.405	.000
.000	-1.143	-1.840	-2.280	-2.295	-2.346	-2.330
-2.637	-2.322	-1.304	-.001	1.368	2.137	2.264
2.109	2.269	2.307	2.457	2.033	1.209	.000
.000	-1.947	-3.375	-3.931	-3.711	-3.551	-3.778
-4.367	-3.876	-2.320	-.001	2.307	3.585	3.748
3.445	3.516	3.775	4.321	3.589	2.037	.000
.000	-2.640	-4.490	-4.959	-4.615	-4.280	-4.721
-5.576	-4.840	-2.876	-.001	2.807	4.482	4.809
4.423	4.360	4.776	5.568	4.740	2.686	.000
.000	-2.857	-4.670	-5.103	-4.715	-4.348	-4.857
-5.826	-5.072	-2.923	-.001	2.875	4.714	5.065
4.626	4.515	4.945	5.806	4.990	2.933	.000
.000	-2.821	-4.495	-4.973	-4.609	-4.269	-4.734
-5.637	-4.871	-2.754	-.001	2.674	4.510	4.832
4.431	4.359	4.780	5.628	4.846	2.911	.000
.000	-2.335	-3.752	-4.169	-3.703	-3.545	-3.825
-4.494	-3.731	-2.148	-.001	1.999	3.114	3.659
3.464	3.615	3.834	4.548	3.973	2.374	.000
.000	-1.426	-2.388	-2.663	-2.348	-2.372	-2.392
-2.613	-2.194	-1.264	-.001	1.136	1.778	2.179
2.186	2.424	2.421	2.784	2.476	1.442	.000
.000	-.505	-.840	-1.043	-1.011	-1.066	-.988
-.998	-.783	-.429	-.002	.418	.733	.940
.951	1.074	1.027	1.070	.874	.510	.000
-.001	-.001	-.001	-.001	-.001	-.001	-.001
-.001	-.001	-.001	-.001	-.001	-.001	-.001
-.001	-.001	-.001	-.001	-.001	-.001	-.001
-.332	-.332	-.332	-.332	-.332	-.332	-.332
-.332	-.332	-.332	-.332	-.332	-.332	-.332
-.332	-.332	-.332	-.332	-.332	-.332	-.332
-1.578	-1.492	-1.249	-.937	-.497	-.317	-.150
-.030	.075	.169	.335	.116	.009	-.086
-.175	-.339	-.516	-.951	-1.235	-1.468	-1.578
-3.628	-3.294	-2.490	-1.632	-.695	-.269	.086
.606	1.117	1.598	1.784	1.568	.832	.346
.037	-.292	-.729	-1.774	-2.778	-3.533	-3.628
-5.041	-4.677	-3.778	-2.263	-.874	-.192	.355
1.525	2.804	3.719	4.023	3.678	2.383	1.082
.277	-.203	-.935	-2.577	-4.119	-5.011	-5.041
-6.132	-5.639	-4.170	-2.280	-.828	-.068	.652
2.423	4.102	5.471	5.782	5.210	3.532	1.828

.594	-.069	-.925	-2.845	-4.627	-5.830	-6.132
-6.423	-5.807	-4.045	-2.124	-.730	.000	-.785
2.755	4.593	5.990	6.378	5.757	4.025	2.166
.747	-.001	-.846	-2.772	-4.629	-6.051	-6.423
-5.767	-5.251	-3.564	-1.858	-.593	.068	.872
2.875	-4.740	-5.988	-6.346	-5.740	-4.192	-2.317
.833	.061	-.708	-2.469	-4.171	-5.535	-5.767
-4.302	-3.842	-2.578	-1.353	-.329	.193	.904
2.722	4.323	5.358	5.334	4.938	3.662	2.292
.884	.177	-.426	-1.721	-2.977	-4.008	-4.302
-2.161	-1.885	-1.326	-.693	-.155	.265	.735
1.923	-3.011	-3.712	-3.821	-3.383	-2.585	-1.727
.732	.246	-.201	-.824	-1.462	-1.933	-2.161
-.487	-.419	-.199	-.089	.091	.305	.518
1.035	1.376	1.648	1.731	1.630	1.318	1.013
.518	.297	.076	-.122	-.260	-.439	-.487
.304	.304	.304	.304	.304	.304	.304
.304	.304	.304	.304	.304	.304	.304
.304	.304	.304	.304	.304	.304	.304
-.298	-.298	-.298	-.298	-.298	-.298	-.298
-.298	-.298	-.298	-.298	-.298	-.298	-.298
-.298	-.296	-.298	-.298	-.298	-.298	-.298
-.421	-.143	.121	.527	.728	.706	.788
.663	.413	.038	-.536	-.665	-.824	-.908
-.836	-1.134	-1.050	-1.006	-.827	-.651	-.421
-.719	.115	.910	1.583	2.049	2.062	2.061
1.926	1.379	.449	-.686	-1.590	-1.830	-1.769
-1.836	-2.149	-2.227	-2.271	-1.954	-1.523	-.719
-.522	.879	1.971	2.744	3.243	3.221	3.280
3.279	2.552	1.121	-.674	-2.284	-3.015	-2.788
-2.874	-3.251	-3.410	-3.671	-3.120	-1.972	-.522
-.244	1.625	2.925	3.434	3.899	3.865	4.032
4.279	3.427	1.839	-.264	-2.234	-3.302	-3.403
-3.664	-4.011	-4.168	-4.481	-3.656	-2.132	-.244
-.001	2.041	3.273	3.606	3.984	3.948	4.181
4.572	-3.735	-2.121	.000	-2.043	-3.252	-3.531
-3.858	-4.164	-4.286	-4.579	-3.723	-2.125	-.001
.291	2.253	3.345	3.599	3.862	3.844	4.104
4.550	3.798	2.216	.248	-1.687	-2.918	-3.321
-3.698	-3.969	-4.078	-4.287	-3.398	-1.814	.291
.776	2.418	3.301	3.459	3.255	3.257	3.503
3.977	-3.388	-2.146	.556	-.970	-1.969	-2.657
-3.070	-3.281	-3.291	-3.314	-2.472	-1.034	.776
.943	1.944	2.537	2.638	2.373	2.378	2.422
2.634	2.316	1.662	.767	-.230	-.983	-1.575
-2.037	-2.117	-2.080	-1.846	-1.191	-.192	.943
.731	1.112	1.261	1.356	1.225	1.350	1.125
1.218	1.096	-.846	.519	-.136	-.138	-.501
-.745	-.659	-.690	-.519	-.137	.270	.731
.436	.436	.436	.436	.436	.436	.436
.436	.436	.436	.436	.436	.436	.436
.436	.436	.436	.436	.436	.436	.436

```
 .000     .000     .000     .000     .000     .000     .000
 .000     .000     .000     .000     .000    -.000    -.000
 .000     .000     .000     .000     .000     .000     .000
 .169     .249     .270     .199     .126     .117     .110
 .133     .095    -.021    -.231    -.237    -.260    -.222
-.126    -.121    -.106    -.106    -.045     .054     .169
 .650     .865     .750     .519     .256     .227     .248
 .397     .303    -.119    -.632    -.885    -.832    -.500
-.281    -.234    -.226    -.290    -.161     .268     .650
 .958    1.315    1.590     .991     .475     .332     .433
 .654     .352    -.365   -1.252   -1.704   -1.591   -1.014
-.500    -.333    -.428    -.536    -.231     .383     .958
1.458    2.002    2.057    1.296     .640     .386     .567
 .722     .363    -.582   -1.573   -2.171   -2.097   -1.344
-.667    -.370    -.553    -.659    -.295     .493    1.458
1.702    2.242    2.162    1.350     .664     .391     .575
 .668     .264    -.683   -1.671   -2.230   -2.244   -1.423
-.701    -.362    -.556    -.640    -.[illegible]86     .709    1.702
1.745    2.275    2.095    1.319     .639     .385     .552
 .626     .223    -.549   -1.488   -1.968   -2.148   -1.373
-.671    -.367    -.542    -.606    -.167     .755    1.745
1.430    1.836    1.711    1.096     .474     .333     .395
 .445     .129    -.453   -1.028   -1.352   -1.397   -1.022
-.507    -.317    -.378    -.420    -.073     .672    1.430
 .860    1.095    1.028     .688     .265     .223     .199
 .204     .048    -.324    -.706    -.826    -.771    -.587
-.290    -.212    -.166    -.202     .013     .433     .860
 .310     .381    -.357    -.265    -.141     .112     .084
 .063    -.023    -.122    -.225    -.280    -.302    -.246
-.143    -.110    -.070    -.049     .041     .178     .310
 .000     .000     .000     .000     .000     .000     .000
 .000     .000     .000     .000     .000     .000     .000
 .000     .000     .000     .000     .000     .000     .000
SAMPLE CASE
100.
14.0      50.0      71900.0   72000.0   0.        0         1
157710.   -27.28    -0.325    2.64      0.0       0.0       0.0
33.       855.3006
632989.   2870412.  2751954.  45813.    391287.   19334.
0.        0.        22323228. 25120.315 40.       0.
-92.      88.7774   270.      1.73623   -.05752   .12772
.9841605  .0162073  -.1765375 .0210063  .9781355  .2069051
.1760309  -.2073364 .9623018
```

III. PROGRAM OUTPUT

A. Output Specifications

The program allows the user to specify up to 90 variables, contained in the first 999 locations of the common block, for printing or to be saved on tape. Printed output data are assigned to Unit 6. Each block of printed data are preceded by the simulation time and case title. The block of printed data (6 variables per line of print) uses an E15.8 format with a four character identification name.

All tape output data are written in binary form. Variables saved for restart are assigned to Unit 13. Data to be plotted or used as input to other programs are assigned to Unit 12. The first variable of the data block saved is the simulation time in seconds. The remaining variables, and their order of output, are specified by the input data (See Section III, Code 2).

B. Sample Program Printout

```
SAMPLE CASE

CASE NUMBER -1

--------------------------------------------------------------------------------

***********VEHICLE DATA***********

WEIGHT =157710.000                        CG-LOCATION = ( -27.2800,   -.3250,    2.6400)
MOMENT REFERENCE POINT = (       .0000,      .0000,      .0000)
AERODYNAMIC REFERENCE LENGTH =   33.0000
AERODYNAMIC REFERENCE AREA =  855.3006

VEHICLE INERTIA MATRIX =  632989.0       -45813.0       -391287.0
                          -45813.0      2870412.0        -19334.0
                         -391287.0       -19334.0       2751954.0

--------------------------------------------------------------------------------

********INITIAL CONDITIONS********

RMAG     = 22323228.
GEOCENTRIC LATITUDE =    .000000
LONGITUDE =     .00000
INERTIAL VELOCITY = 25120.315
INERTIAL HEADING =  40.00000
INERTIAL FLIGHT PATH ANGLE =    .000000

VEHICLE ATTITUDE DATA - PHIBI  +1 = -92.00000
                        ALPHAI +2 =  88.77740
                        PHIAI  +1 = 270.00000

INITIAL BODY RATES      ROLL RATE =    1.73623
                        PITCH RATE =   -.05752
                        YAW RATE =     .12772

ABP          .9841605E 00     -.1620730E-01     -.1765375E 00
             .2100630E-01      .9781355E 00      .2069051E 00
             .1760309E 00     -.2073364E 00      .9623018E 00

--------------------------------------------------------------------------------

********OTHER INFORMATION********

THIS PROGRAM USES      SSL DENSITIES,F(LAT,LONG),230NM,MEASURED

DELT 14.000000     DTPR  50.00000    DTSM  71900.00    TRUN  72000.00    TIME      .00    IOPTI  0.
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
TIME    .00000000E 00   SAMPLE CASE

 THOR .00000000E 00  QSQR .10000000E 01  PHIB-.92000001E 02  ALPT .88777401E 02  PHIA .27000000E 03  VREL .24106224E 05
 DELT .14000000E 02  GAMR .50330791E-24  SIGR -.37034776E 02  VELX .51434911E 03  VELY-.24100736E 05  VELZ .61097815E-05
 P     .17362300E 01  PDOT-.29179373E-03  GGB1 .67537424E-01  TOAX-.19997458E-03  TABX .67337449E-01  CAR1-.44971102E 01
 Q    -.57520000E-01  QDOT-.42798320E-02  GGB2-.12011496E 01  TOAY-.22367822E-04  TABY-.12011719E 01  CAR2 .17662010E-02
 R     .12772000E 00  RDOT .22653979E-02  GGB3 .14629498E-02  TOAZ-.25852463E-01  TABZ-.24389513E-01  CAR3 .13059337E-02
 BR11 .21336776E-01  BR21-.99916331E 00  BR31 .34891552E-01  BR12-.99977235E 00  BR22-.21323778E-01  BR32 .74464249E-03
 BR13 .25345245E-09  BR23-.34899497E-01  BR33-.99939083E 00  IB11-.34891552E-01  IB21-.74464243E-03  IB31 .99939083E 00
 IB12-.78475105E 00  IB22-.61918466E 00  IB32-.27859224E-01  IB13 .61882822E 00  IB23-.78524506E 00  IB33 .21019955E-01
 RHO  .73723390E-14  QBAR .21420701E-05  LAT  .00000000E 00  LONI .00000000E 00  GAMI .48298970E-24  SIGI .40000000E 02

TIME    .50400000E 04   SAMPLE CASE

 THOR .14000000E 01  QSQR .10000039E 01  PHIB-.14847984E 03  ALPT .10656796E 03  PHIA .31927640E 03  VREL .24091804E 05
 DELT .14000000E 02  GAMR-.83026439E-01  SIGR .43189544E 02  VELX-.68698884E 04  VELY-.15065303E 05  VELZ .17500441E 05
 P     .16769681E 01  PDOT .73576451E-03  GGB1 .59098450E 00  TOAX-.79386784E-02  TABX .59892318E 00  CAR1-.75672781E 01
 Q     .20990150E 00  QDOT .31259377E-02  GGB2 .19439304E 01  TOAY .39763889E-01  TABY .19836943E 01  CAR2 .25659871E 00
 R     .43336474E 00  RDOT-.42785581E-02  GGB3-.37751404E 01  TOAZ-.31983303E-01  TABZ-.37431571E 01  CAR3 .10902362E-01
 BR11-.28515458E 00  BR21-.50109348E 00  BR31 .81706463E 00  BR12-.62532898E 00  BR22-.54878170E 00  BR32-.55480011E 00
 BR13 .72640643E 00  BR23-.66914806E 00  BR33-.15685382E 00  IB11-.84633506E 00  IB21 .81173253E-01  IB31 .52643165E 00
 IB12-.23265300E 00  IB22-.94539474E 00  IB32-.22825910E 00  IB13 .47915756E 00  IB23-.31565889E 00  IB33 .81902140E 00
 RHO  .60555831E-14  QBAR .17573757E-05  LAT -.25904655E 02  LONI .33561530E 03  GAMI .19934758E-01  SIGI .45625349E 02

TIME    .10080000E 05   SAMPLE CASE

 THOR .28000000E 01  QSQR .10000079E 01  PHIB-.55653710E 02  ALPT .93810592E 02  PHIA .31534739E 02  VREL .24067797E 05
 DELT .14000000E 02  GAMR-.53289897E-01  SIGR-.66409990E 02  VELX-.15995385E 04  VELY-.12560254E 05  VELZ .20468605E 05
 P     .17174721E 01  PDOT-.83596535E-03  GGB1-.54336834E 00  TOAX .62465101E-01  TABX-.48090324E 00  CAR1-.78386424E 01
 Q    -.16706133E 00  QDOT-.10221143E-02  GGB2 .21335679E 01  TOAY .34285919E-01  TABY .21678538E 01  CAR2 .16832387E 00
 R     .25983901E 00  RDOT .50209526E-02  GGB3 .30950126E 01  TOAZ-.13556832E-01  TABZ .30814558E 01  CAR3-.21595551E 00
 BR11-.66459696E-01  BR21-.82383186E 00  BR31-.56295585E 00  BR12 .52186970E 00  BR22 .45218628E 00  BR32-.72333376E 00
 BR13 .85045610E 00  BR23-.34187357E 00  BR33 .39987746E 00  IB11 .11030652E 00  IB21 .74867493E 00  IB31 .65372275E 00
 IB12-.98880791E 00  IB22 .16072869E-01  IB32 .14843419E 00  IB13 .10063051E 00  IB23-.66276666E 00  IB33 .74207072E 00
 RHO  .42199782E-14  QBAR .12222299E-05  LAT -.45850138E 02  LONI .29945043E 03  GAMI .18323906E-01  SIGI .67444995E 02

TIME    .15120000E 05   SAMPLE CASE

 THOR .42000000E 01  QSQR .10000119E 01  PHIB-.64085556E 02  ALPT .12164011E 03  PHIA .94233077E 02  VREL .24067260E 05
 DELT .14000000E 02  GAMR .53421578E-01  SIGR .10721627E 03  VELX-.12625363E 05  VELY .20434181E 05  VELZ-.15124531E 04
 P     .17108091E 01  PDOT .86864621E-03  GGB1-.24130842E-01  TOAX .15912015E-02  TABX-.22539641E-01  CAR1-.41579238E 01
 Q     .22888160E 00  QDOT-.11841670E-02  GGB2 .17522241E 01  TOAY .14564135E-02  TABY .17536605E 01  CAR2 .32695276E-01
 R     .25570274E 00  RDOT-.46911601E-02  GGB3 .30446860E 00  TOAZ .77407063E-02  TABZ .31220931E 00  CAR3 .73747913E-01
 BR11-.52458663E 00  BR21-.76576108E 00  BR31-.37207220E 00  BR12 .84904478E 00  BR22-.50280950E 00  BR32-.16224456E 00
 BR13-.62842761E-01  BR23-.40104102E 00  BR33 .91395269E 00  IB11-.60295642E 00  IB21 .77832291E 00  IB31 .17514531E 00
 IB12-.73231543E 00  IB22-.62706959E 00  IB32 .26554769E 00  IB13 .31650678E 00  IB23 .31857627E-01  IB33 .94810399E 00
 RHO  .29635062E-14  QBAR .85828028E-06  LAT -.47867339E 02  LONI .24737058E 03  GAMI .10900868E-03  SIGI .10647919E 03
```

```
TIME    .20160000E 05    SAMPLE CASE

THCR .56000000E 01  QSQR .10000159E 01  PHIB-.92602805E 02  ALPT .34425274E 02  PHIA .15911975E 03  VREL .24093032E 05
DELT .14000000E 02  GAMR .98312027E-01  SIGR .13428372E 03  VELX .19874588E 05  VELY .48548482E 04  VELZ-.12726749E 05
P    .16834588E 01  PDOT-.72622674E-03  GGB1-.24478415E 00  TOAX-.29721171E-02  TABX-.24775627E 00  CAR1-.55804310E 01
Q   -.13251078E 00  QDOT-.34428498E-02  GGB2-.36302376E 00  TOAY-.25061130E-01  TABY-.38808489E 00  CAR2-.37039723E 00
R    .44714938E 00  RDOT .34141259E-02  GGB3 .87371921E 00  TOAZ-.11868536E-01  TABZ .86185068E 00  CAR3 .92113528E-01
BR11 .82491620E 00  BR21-.56476793E 00  BR31-.25673663E-01  BR12 .20150425E 00  BR22 .33614470E 00  BR32 .92005468E 00
BR13-.52823358E 00  BR23-.75376000E 00  BR33 .39107676E 00  IB11 .55511788E 00  IB21 .83111308E 00  IB31-.34553815E-01
IB12-.82060342E 00  IB22 .54035081E 00  IB32-.18635687E 00  IB13-.13620922E 00  IB23 .13179490E 00  IB33 .98192841E 00
RHO  .26316836E-14  QBAR .76381219E-06  LAT -.30025320E 02  LONI .20786087E 03  GAMI-.10829239E-01  SIGI .13203963E 03

TIME    .25200000E 05    SAMPLE CASE

THOR .70000000E 01  QSQR .10000199E 01  PHIB-.89926058E 02  ALPT .91196217E 02  PHIA .22676562E 03  VREL .24113241E 05
DELT .14000000E 02  GAMR .20195882E-01  SIGR .14279188E 03  VELX-.50343715E 03  VELY-.17565441E 05  VELZ-.16514895E 05
P    .17391894E 01  PDOT .29290790E-03  GGB1 .23098630E 00  TOAX-.33288075E-01  TABX .19769823E 00  CAR1-.68919946E 01
Q    .83603655E-01  QDOT-.47033452E-02  GGB2-.71178876E 00  TOAY-.14941819E-01  TABY-.72673058E 00  CAR2 .11788410E-01
R    .11138874E 00  RDOT-.11908389E-02  GGB3-.66969949E 00  TOAZ .14447640E-01  TABZ-.65525185E 00  CAR3-.31058728E 00
BR11-.20878037E-01  BR21-.99982021E 00  BR31-.12903085E-02  BR12-.72845624E 00  BR22 .14334069E-01  BR32 .68502812E 00
BR13-.68488906E 00  BR23 .15248600E-01  BR33-.72856832E 00  IB11-.18770412E-01  IB21 .61775713E 00  IB31-.78619451E 00
IB12-.78512956E 00  IB22 .47780779E 00  IB32 .39419383E 00  IB13 .61911005E 00  IB23 .62465457E 00  IB33 .47605332E 00
RHO  .26629954E-14  QBAR .77419721E-06  LAT -.45909697E 01  LONI .18237691E 03  GAMI-.27294854E-02  SIGI .13984465E 03

TIME    .30240000E 05    SAMPLE CASE

THOR .84000000E 01  QSQR .10000238E 01  PHIB-.15057682E 03  ALPT .10550746E 03  PHIA .28242493E 03  VREL .24102279E 05
DELT .14000000E 02  GAMR-.83478081E-01  SIGR .13883130E 03  VELX-.64445305E 04  VELY-.22682484E 05  VELZ .49974140E 04
P    .16685238E 01  PDOT .27485078E-03  GGB1 .29431738E 00  TOAX-.36509205E-02  TABX .29066646E 00  CAR1-.49371637E 01
Q    .11994295E 00  QDOT .48753305E-02  GGB2 .36294657E 01  TOAY .19398469E-02  TABY .36314056E 01  CAR2 .19840531E 00
R    .50414431E 00  RDOT-.20554326E-02  GGB3-.22428856E 01  TOAZ-.18816029E-02  TABZ-.22447672E 01  CAR3 .48341630E-01
BR11-.26738262E 00  BR21-.47341446E 00  BR31 .83938097E 00  BR12-.94109294E 00  BR22-.59140360E-01  BR32-.33315752E 00
BR13 .20734197E 00  BR23-.87893180E 00  BR33-.42969288E 00  IB11 .83849382E 00  IB21 .17726502E 00  IB31-.51544192E 00
IB12-.51920151E 00  IB22 .54766614E 00  IB32-.65623092E 00  IB13 .16593085E 00  IB23 .81779353E 00  IB33 .55120629E 00
RHO  .30410736E-14  QBAR .88331002E-06  LAT  .21651254E 02  LONI .15872397E 03  GAMI .96768489E-02  SIGI .13623389E 03

TIME    .35280000E 05    SAMPLE CASE

THOR .98000000E 01  QSQR .10000278E 01  PHIB-.57168979E 02  ALPT .94970126E 02  PHIA .15246627E 03  VREL .24073978E 05
DELT .14000000E 02  GAMR-.81904632E-01  SIGR .11920856E 03  VELX-.20857588E 04  VELY .63072732E 04  VELZ .23140179E 05
P    .17314916E 01  PDOT-.67096724E-03  GGB1-.65834859E 00  TOAX-.60909402E-01  TABX-.59743919E 00  CAR1-.87391084E 01
Q   -.13844598E 00  QDOT-.27343138E-02  GGB2 .12049614E 01  TOAY .40612918E-01  TABY .12455743E 01  CAR2 .93774492E-01
R    .19042582E 00  RDOT .43338830E-02  GGB3 .36304254E-01  TOAZ-.40890499E-02  TABZ .36263363E-01  CAR3-.94384181E-01
BR11-.86639555E-01  BR21-.83715151E 00  BR31-.54014906E 00  BR12 .26199547E 00  BR22 .50403256E 00  BR32-.82311071E 00
BR13 .96121126E 00  BR23-.21282291E 00  BR33 .17565346E 00  IB11-.19022277E 00  IB21-.56121353E 00  IB31-.80557106E 00
IB12-.30754901E 00  IB22 .81338905E 00  IB32-.49397728E 00  IB13 .93237276E 00  IB23 .15375084E 00  IB33-.32728662E 00
RHO  .39717499E-14  QBAR .11509266E-05  LAT  .43301751E 02  LONI .12554690E 03  GAMI .60318506E-02  SIGI .11789020E 03
```

Appendix A

PROGRAM VARIABLE DEFINITIONS

The following program variables, located in the COMMON BLOCK, are alphabetized according to their Fortran Mnemonics.

Program Symbol	Variable Definition	Units
ABI(3,3)	Transformation matrix from I to B-frame	Unitless
ACCOM(582)	Dummy common	Unitless
ADG(3,3)	Transformation matrix from G to D-frame	Unitless
ADI(3,3)	Transformation matrix from I to D-frame	Unitless
ADR(3,3)	Transformation matrix from R to D-frame	Unitless
AGR(3,3)	Transformation matrix from R to G-frame	Unitless
AIB(3,3)	Transformation matrix from B to I-frame	Unitless
AID(3,3)	Transformation matrix from D to I-frame	Unitless
AIG(3,3)	Transformation matrix from G to I-frame	Unitless
ALPHT	Total angle-of-attack	rad
ARB(3,3)	Transformation matrix from B to R-frame	Unitless
ARD(3,3)	Transformation matrix from D to R-frame	Unitless
ARI(3,3)	Transformation matrix from I to R-frame	Unitless
ARP(3,3)	Transformation matrix from P to R-frame	Unitless
ATCOM(45)	Dummy common	Unitless
CA	Axial force coefficient	Unitless
CCCOM(13)	Dummy common	Unitless

Program Symbol	Variable Definition	Units
CH(3)	Coefficients for geodetic latitude computations	Unitless
CEN	Yawing moment coefficient	Unitless
CL	Rolling moment coefficient	Unitless
CLMNT(3)	Aerodynamic moment coefficients about vehicle center of mass	Unitless
CM	Pitching moment coefficient	Unitless
CN	Normal force coefficient	Unitless
CNV	57.29577951	deg/rad
CR	Oblate earth radial coefficient	Unitless
CW	Oblate earth spin axis coefficient	Unitless
CPLMN(3)	Aerodynamic moment coefficient vector about moment reference point	Unitless
CXYZ(3)	Aerodynamic force coefficient vector	Unitless
CY	Side force coefficient	Unitless
DIW(3)	Dummy variable used in gravity gradient torque calculation	slugs-ft^2
DPSI	Difference between geodetic and geocentric latitudes	rad
DREF	Aerodynamic reference diameter	ft
DT	Integration time step	sec
DTP	Output frequency for tape save	sec
DTSAM	Specified time to save variables for restart	sec
EPCOM(66)	Dummy common	Unitless
FAB(3)	B-frame aerodynamic forces	lb
FAI(3)	I-frame aerodynamic forces	lb
FAR(3)	R-frame aerodynamic forces	lb
FARC(3)	R-frame aerodynamic force coefficients	Unitless
FCA(21,11)	Table of axial force coefficients	Unitless
FCEN(21,11)	Table of yawing moment coefficients about moment reference point	Unitless

Program Symbol	Variable Definition	Units
FCL(21,11)	Table of rolling moment coefficients about moment reference point	Unitless
FCM(21,11)	Table of pitching moment coefficients about moment reference point	Unitless
FCN(21,11)	Table of normal force coefficients	Unitless
FCY(21,11)	Table of side force coefficients	Unitless
FOMI(3)	Aerodynamic term in translational acceleration calculation	ft/sec^2
FRHO(11,37)	Table of atmospheric densities	kg/m^3
G(3)	Gravitational acceleration	ft/sec^2
GAMI	Inertial flight path angle (positive up from local geocentric horizontal)	rad
GAMR	Relative flight path angle (positive up from local geodetic horizontal)	rad
GCR	Temporary variable in acceleration of gravity calculation	ft/sec^2
GGB(3)	Gravity gradient torques in B-frame	ft-lb
GMAG	Temporary variable in acceleration of gravity calculation	ft/sec^2
GMASI	Reciprocal of vehicle mass	1/slugs
GMASS	Vehicle mass	slugs
H	Geodetic altitude	ft
HB(3)	Angular momentum in B-frame	ft-lb-sec
IXYZ(3,3)	Moment of inertia tensor	$slugs\text{-}ft^2$
IXYZI(3,3)	Inverse of moment of inertia tensor	$1/slugs\text{-}ft^2$
LAMDA	Inertial longitude measured in I-frame	rad
LAMDE	Earth fixed longitude	rad
LMRP(3)	Temporary variable in aerodynamic moment calculation	Unitless
OMGE	Earth's rotation rate	rad/sec
PCDUM(12)	Dummy common	Unitless
PHIA	Aerodynamic roll angle	rad

Program Symbol	Variable Definition	Units
PHIBK	Bank angle	rad
PI	π	rad
PQR(3)	Angular rates in B-frame	rad/sec
PQRD(3)	Angular accelerations in B-frame	rad/sec^2
PSI	Geocentric latitude (positive north, negative south of equator)	rad
PSID	Geodetic latitude (positive north, negative south of equator)	rad
Q(4)	Quaternion parameters	Unitless
QBAR	Dynamic pressure	lb/ft^2
QD(4)	Time derivities of quaternions	1/sec
QSQR	$\sqrt{Q(1)^2 + Q(2)^2 + Q(3)^2 + Q(4)^2}$	Unitless
R(3)	Vehicle I-frame position	ft
R1(3)	Unit vector along vehicle I-frame position	Unitless
RB1(3)	Unit vector along vehicle B-frame position	Unitless
RDOT(3)	Vehicle I-frame translational velocity	ft/sec
RHO	Atmospheric density	slugs/ft^3
RMAG	Magnitude of vehicle position vector	ft
RPSI	Radius of the earth	ft
RR,RR2	Temporary variables used in gravitational acceleration calculations	Unitless
SD,SJ,SH	Earth's gravitational constants	Unitless
SIGI	Inertial heading (positive clockwise from north)	rad
SIGR	Relative heading (positive clockwise from north)	rad
SPSI,SPSI2	Temporary variables used in gravitational acceleration calculation	Unitless
SREF	Aerodynamic reference area	ft^2
SWD,SWH	Temporary variables used in gravitational acceleration calculation	Unitless

Program Symbol	Variable Definition	Units
TAB(3)	Total external torques in B-frame	ft-lb
TALP(11)	Table of angle-of-attack values	rad
TEND	Simulation stop time	sec
THOR	Simulation time	hr
TIME	Simulation time	sec
TLAT(11)	Table of geocentric latitude values	rad
TLNG(37)	Table of longitude values	rad
TOA(3)	Aerodynamic torques in B-frame	ft-lb
TPHIA(21)	Table of aerodynamic roll angle values	rad
TRUN	Total run time	sec
TSUM(3)	Time rate of change of vehicle angular momentum	ft lb
TWOPI	2π	rad
V(3)	Inertial velocity in I-frame	ft/sec
VACOM(67)	Dummy common	Unitless
VATM(3)	Atmospheric velocity in I-frame	ft/sec
VCCOM(18)	Dummy common	Unitless
VDCOM(107)	Dummy common	Unitless
VDOT(3)	Inertial translation acceleration in I-frame	ft/sec^2
VID(3)	Inertial velocity in D-frame	ft/sec
VIG(3)	Inertial velocity in G-frame	ft/sec
VIMAG	Magnitude of inertial velocity	ft/sec
VPCOM(39)	Dummy common	Unitless
VRELB(3)	Relative velocity in B-frame	ft/sec
VRELD(3)	Relative velocity in D-frame	ft/sec
VRELG(3)	Relative velocity in G-frame	ft/sec
VRELI(3)	Relative velocity in I-frame	ft/sec
VRMAG	Magnitude of relative velocity	ft/sec
WT	Vehicle weight	lb
WXH(3)	Angular momentum direction change	ft-lb
XCG,YCG,ZCG	Cg location in N-frame along x,y, and z-axes, respectively	ft

Program Symbol	Variable Definition	Units
XDUM(27)	Dummy common	Unitless
XDDUM(27)	Dummy common	Unitless
XMRP, YMRP, ZMRP	Moment reference point in N-frame along x, y, and z-axes, respectively	ft

Appendix B

REFERENCE FRAMES AND TRANSFORMATIONS

A. Reference Frames

All reference frames are right handed systems (Figs. B-1, B-2, and B-3).

1. I — Inertial Frame. The I-frame has its origin at the center of the earth with the X_I axis through the Greenwich meridian at time zero. The Z_I axis points through the North Pole and the Y_I axis completes the right handed system. It is in this frame the accelerations are integrated.

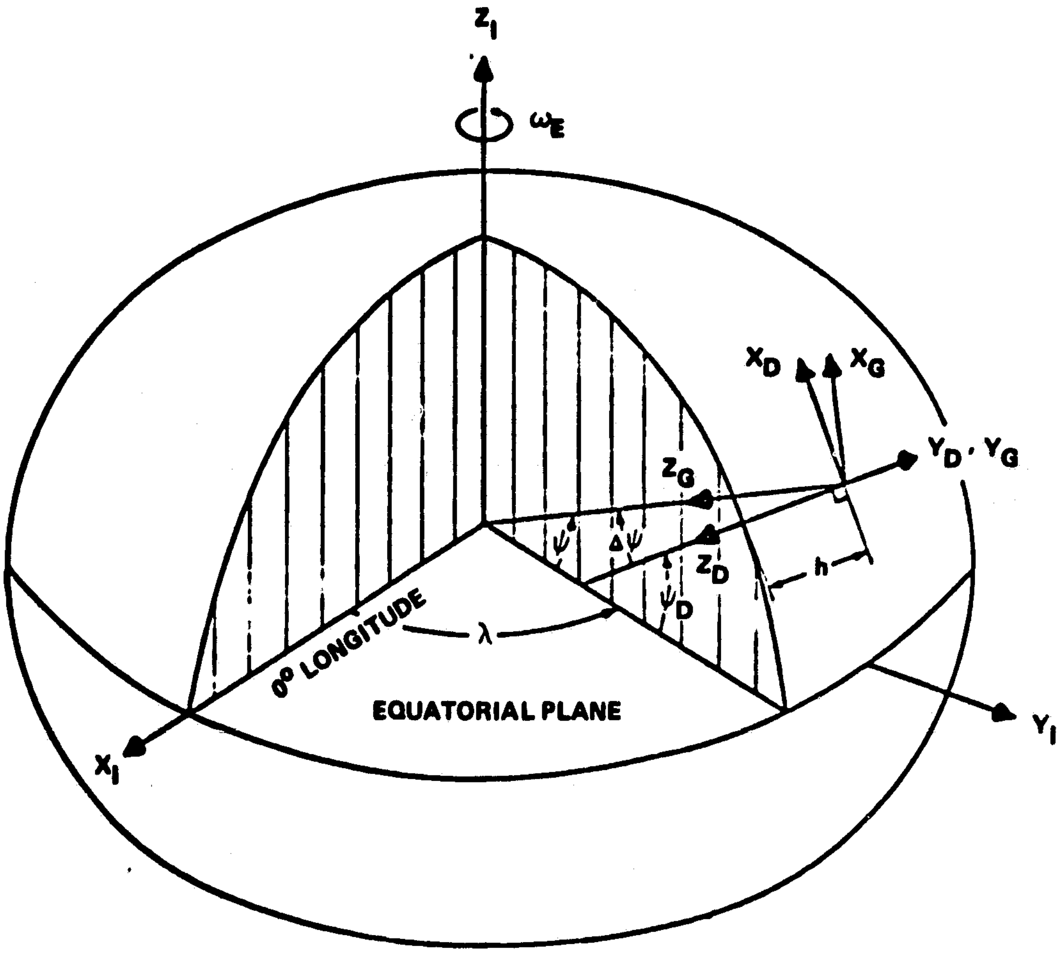
2. G — Geocentric Frame. The G-frame has its origin at the vehicle's center of mass. The X_G axis points north, the Y_G axis points east, and the Z_G axis is directed downward along the radius vector to the earth's center.

3. D — Geodetic Frame. The D-frame has its origin at the vehicle's center of mass (Fig. B-2). The X_D axis points north, the Y_D axis points east, and the Z_D axis is directed downward along the local geodetic.

4. R — Relative Velocity Frame. The R-frame has its origin at the vehicle's center of mass (Fig. B-2). The X_R axis is directed along the relative velocity vector. The Z_R axis is directed downward in a plane containing the velocity vector and the local geodetic. The Y_R axis completes the right handed system.

5. B — Body Frame. The body fixed B-frame has its origin at the vehicle's center of mass (Fig. B-3). The direction of the axes are chosen so as to be consistent with the definition of aerodynamic parameters. It is in this frame that the external forces and moments are computed.

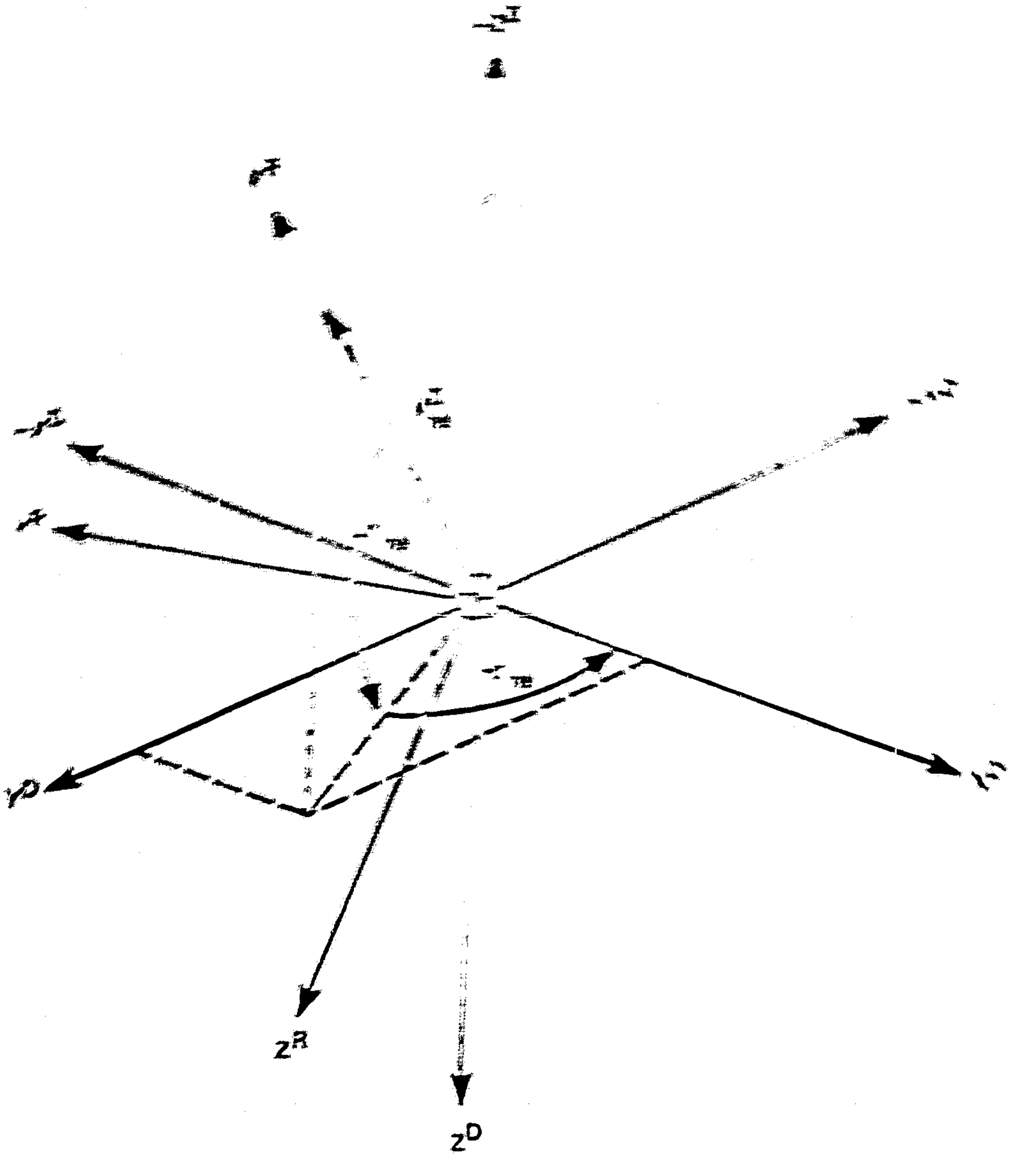
6. N — Input Data Reference Frame. This body fixed reference frame is parallel to the B-frame coordinate system with its origin chosen by the user. It is in this frame that the aerodynamic data and mass properties data are read into the program. Usually, it is convenient to choose the aerodynamic moment reference point as the origin of this frame.



$$A^{GI} = [-(90 + \psi)]_2 \; [\lambda]_3$$

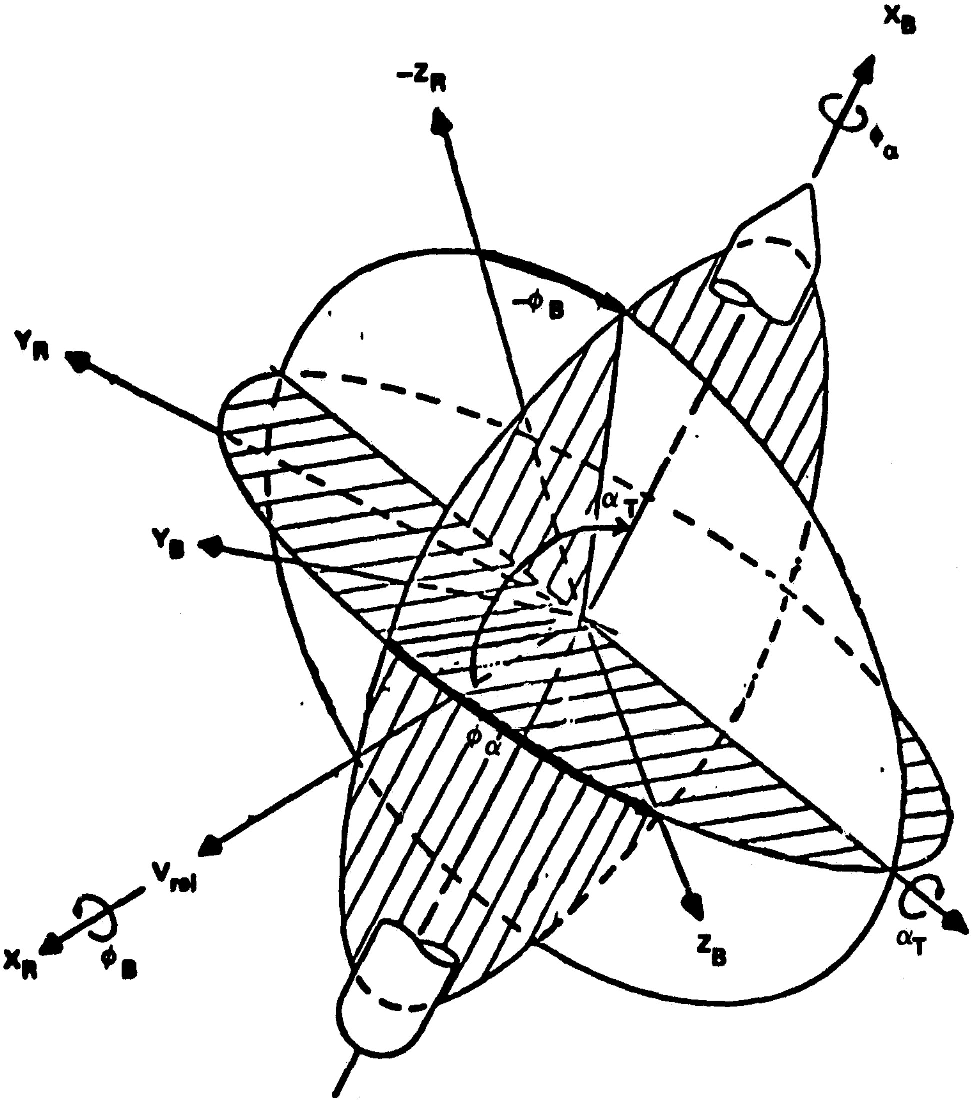
$$A^{DI} = [-(90 + \psi_D)]_2 \; [\lambda]_3$$

Figure B-1. Inertial (I), geocentric (G) and geodetic (D) reference frames.



$$A^{RD} = [\gamma_R]_2 \quad [\sigma_R]_3$$

Figure B-2. Geodetic (D) and relative velocity (R) reference frames.



$$A^{BR} = [\varphi_\alpha]_1 \, [\alpha_T]_2 \, [\varphi_B]_1$$

Figure B-3. Relative velocity (R) and Body (B) reference frames.

B. Transformations

The transformation matrix A^{ML} transforms vectors from the L-frame to the M-frame. The matrices are formed by successive rotations through the indicated Euler angles. The rotation $[\theta]_i$ is used to indicate the direction cosine matrix for a positive rotation about the i^{th} axis through the angle θ. The sequence of rotations is read from right to left. Since all transformations shown are orthogonal, the inverse transformation matrices are merely the transpose of those given.

1. Transformation from I to G-frame A^{GI}

$$A^{GI} = [-(90 + \psi)]_2 \, [\lambda]_3$$

where λ = inertial longitude

ψ = geocentric latitude

2. Transformation from I to D-frame A^{DI}

$$A^{DI} = [-(90 + \psi_D)]_2 \, [\lambda]_3$$

where λ = inertial longitude

ψ_D = geodetic latitude

3. Transformation from D to R-frame A^{RD}

$$A^{RD} = [\gamma_R]_2 \, [\sigma_R]_3$$

where σ_R = relative heading

γ_R = relative flight path angle

4. Transformation from R to B-frame

$$A^{BR} = [\phi_\alpha]_1 \, [\alpha_T]_2 \, [\phi_\beta]_1$$

where ϕ_β = bank angle

α_T = total angle-of-attack

ϕ_α = aerodynamic roll angle

5. **Transformation from I to B-frame**

Initially, the transformation from the I to B-frame is computed by $A^{BI} = A^{BR} A^{RD} A^{DI}$

From the inverse of this transformation, the initial quaternion parameters are computed. The four quaternion parameters are defined as follows:

$$Q_1 = \alpha \sin (\phi/2)$$

$$Q_2 = \beta \sin (\phi/2)$$

$$Q_3 = \gamma \sin (\phi/2)$$

$$Q_4 = \cos (\phi/2)$$

where α, β, and γ are eigenaxis direction cosines, and ϕ is the eigenaxis rotation angle. The quaternions are initialized as follows:

$$A^{IB} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

$$Q_4 = \frac{1}{2} \sqrt{1 + a_{11} + a_{22} + a_{33}}$$

If $Q_4 \neq 0$,

$$Q_1 = \frac{1}{4\,Q_4} (a_{32} - a_{23})$$

$$Q_2 = \frac{1}{4\,Q_4}\,(a_{13} - a_{31})$$

$$Q_3 = \frac{1}{4\,Q_4}\,(a_{21} - a_{12})$$

If $Q_4 = 0$,

$$Q_1 = \sqrt{-\frac{1}{2}\,(a_{22} + a_{33})}$$

If $Q_1 \neq 0$,

$$Q_2 = \frac{1}{4\,Q_1}\,(a_{12} + a_{21})$$

$$Q_3 = \frac{1}{4\,Q_1}\,(a_{13} + a_{31})$$

If $Q_1 = 0$,

$$Q_2 = \sqrt{\frac{1}{2}\,(a_{22} + 1)}$$

If $Q_2 \neq 0$,

$$Q_3 = \frac{1}{4\,Q_2}\,(a_{23} + a_{32})$$

If $Q_2 = 0$,

$$Q_3 = 1$$

From the integrated values of the quaternion parameter time derivaties, the A^{IB} transformation matrix is updated as shown in Appendix C.

Appendix C

EQUATIONS OF MOTION

This appendix presents briefly the equations of motion used to describe the dynamics of a vehicle in 6-degrees-of-freedom. The equations were derived for a vehicle of constant mass distribution and the external forces and torques considered were those resulting only from gravity and aerodynamics. The mathematical models for the Earth and gravity can be found in Reference 1. The atmospheric density is an input to the program as a function of position in orbit.

The aerodynamic force and moment coefficients and the atmospheric density are determined by table lookup at each integration step. The aerodynamic coefficients (C_A, C_Y, C_N, C_ℓ, C_m, C_n) are a function of total angle-of-attack, α_T, and aerodynamic roll angle, ϕ_α. The density (ρ) is a function of geocentric latitude, ψ, and inertial longitude, λ. These angles are defined as

$$\alpha_T = \tan^{-1} \frac{\sqrt{V_{RELB}(2)^2 + V_{RELB}(3)^2}}{V_{RELB}(1)} \qquad 0 \leq \alpha_T \leq 180°$$

$$\phi_\alpha = \tan^{-1} \frac{V_{RELB}(2)}{V_{RELB}(3)} \qquad 0 \leq \phi_\alpha \leq 360°$$

$$\psi = \tan^{-1} \frac{R(3)}{\sqrt{R(1)^2 + R(2)^2}} \qquad -90° \leq \psi \leq 90°$$

$$\lambda = \tan^{-1} \frac{R(2)}{R(1)} \qquad 0 \leq \lambda \leq 360°$$

where

$\bar{V}_{RELB}$ = relative velocity vector in B-frame

$\bar{R}$ = inertial I-frame position vector.

A. Translational Motion

The translational equation of motion for a body in the inertial I-frame is

$$\ddot{\bar{R}} = \bar{g} + \frac{1}{m} \bar{F}_{AI}$$

where

$\ddot{\bar{R}} = \dot{\bar{V}}$ = inertial acceleration vector

$\bar{g}$ = gravitational acceleration vector

m = mass of the body

$\bar{F}_{AI}$ = I-frame aerodynamic forces

The I-frame velocity equation is

$$\dot{\bar{R}} = \bar{V} = \dot{\bar{R}}_o + \int \ddot{\bar{R}}\, dt$$

where

$\dot{\bar{R}}_o = \bar{V}_o$ = initial I-frame velocity vector.

Integrating the velocity vector and adding the initial position, the I-frame position vector is

$$\bar{R} = \bar{R}_o + \int \dot{\bar{R}}\, dt$$

where

$\bar{R}_o$ = initial I-frame position vector.

The aerodynamic forces are computed in the body B-frame and transformed into the I-frame for the translational acceleration computation. The B-frame aerodynamic forces are

$$\bar{F}_{AB} = q_{BAR} \, S_{Ref} \, \bar{C}_{XYZ}$$

where

$$\bar{C}_{XYZ} = \begin{bmatrix} -C_A \\ C_Y \\ -C_N \end{bmatrix} = \text{B-frame aerodynamic force coefficient vector}$$

$$q_{BAR} = \frac{1}{2} \rho \left| V_{RELB} \right|^2 = \text{dynamic pressure}$$

$$S_{Ref} = \text{aerodynamic reference area.}$$

Then the I-frame aerodynamic forces are

$$\bar{F}_{AI} = A^{IB} \, \bar{F}_{AB} \quad .$$

B. Rotational Motion

The equation describing the rotational motion about the center of mass of a rigid body in the rotating B-frame is given by

$$[I] \, \dot{\bar{\omega}} = \bar{T}_{AB} - \bar{\omega} \times ([I] \cdot \bar{\omega})$$

where

$$[I] = \begin{bmatrix} I_{XX} & -I_{XY} & -I_{XZ} \\ -I_{XY} & I_{YY} & -I_{YZ} \\ -I_{XZ} & -I_{YZ} & I_{ZZ} \end{bmatrix} = \text{inertia tensor}$$

$\dot{\bar{\omega}}$ = $\overline{PQRD}$ = B-frame angular aceleration vector

$\bar{\omega}$ = $\overline{PQR}$ = B-frame angular velocity vector

$\bar{T}_{AB}$ = B-frame external torques.

Then

$$\bar{\omega} = \bar{\omega}_o + \int [I]^{-1} [\bar{T}_{AB} - \bar{\omega} \times ([I] \cdot \bar{\omega}) \, dt$$

where

$\bar{\omega}_o$ = initial B-frame angular velocity vector.

The aerodynamic torques about the vehicles center of mass are

$$\bar{T}_{OA} = q_{BAR} \, S_{Ref} \, D_{Ref} \left[\bar{C}_{PLMN} + \frac{(\bar{X}_{MRP} - \bar{X}_{CG})}{D_{Ref}} \times \bar{C}_{XYZ} \right]$$

where

D_{Ref} = aerodynamic reference diameter

$$\bar{C}_{PLMN} = \begin{bmatrix} C_\ell \\ C_m \\ C_n \end{bmatrix}$$ = moment coefficients about the moment reference point

$\bar{X}_{MRP}$ = aerodynamic moment reference point

$\bar{X}_{CG}$ = center of mass location.

The gravitational torques are given by

$$\overline{GGB} = \frac{3\ GM}{|R|^3} [\overline{RB1} \times [I]\ \overline{RB1}]$$

where

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

$\overline{RB1}$ = unit vector along body B-frame position.

The B-frame external torques are

$$\bar{T}_{AB} = \bar{T}_{OA} + \overline{GGB} \ .$$

The quaternion parameters are updated in a simular manner:

$$\bar{Q} = \bar{Q}_o + \int \dot{\bar{Q}}\, dt$$

where

$\bar{Q}_o$ = initial quaternion parameters defined in Appendix B.

The time derivity $\dot{\bar{Q}}$ is defined as

$$\dot{\bar{Q}} = \begin{bmatrix} \dot{Q}_1 \\ \dot{Q}_2 \\ \dot{Q}_3 \\ \dot{Q}_4 \end{bmatrix} = \begin{bmatrix} Q_4 & -Q_3 & Q_2 \\ Q_3 & Q_4 & -Q_1 \\ -Q_2 & Q_1 & -Q_4 \\ -Q_1 & -Q_2 & -Q_3 \end{bmatrix} \bar{\omega} \ .$$

From these integrated values of $\bar{Q}$, the A^{IB} transformation matrix is calculated as follows:

$$A^{IB} = \begin{bmatrix} (2\,Q_1^2 + 2\,Q_4^2 - 1) & 2(Q_1\,Q_2 - Q_3\,Q_4) & 2(Q_1\,Q_3 + Q_2\,Q_4) \\ 2(Q_1\,Q_2 + Q_3\,Q_4) & (2\,Q_2^2 + 2\,Q_4^2 - 1) & 2(Q_2\,Q_3 - Q_1\,Q_4) \\ 2(Q_1\,Q_3 - Q_2\,Q_4) & 2(Q_2\,Q_3 + Q_1\,Q_4) & (2\,Q_3^2 + 2\,Q_4^2 - 1) \end{bmatrix}$$

Appendix D

PROGRAM LISTING

```
C***********************************************************************
C
C     SKYLAB DYNAMICS PROGRAM(SKYDYN)
C
C
C***********************************************************************
C         THIS PROGRAM IS AN ALTERED VERSION OF NORTHROP SERVICES PROGRAM
C     REENTR - SPACE SHUTTLE EXTERNAL TANK REENTRY SIMULATION. THE
C     THEORY USED IN THIS PROGRAM IS DOCUMENTED IN -
C        M-250-1393
C     MANY OF THE OPTIONS IN PROGRAM REENTR ARE DELETED IN
C     ORDER TO GIVE MAXIMUM SPEED TO THIS VERSION.
C
C                          ASSUMPTIONS
C
C          1.  FISCHER ELLIPSOID EARTH MODEL
C          2.  DENSITY INPUT AS FUNCTION OF ORBITAL POSITION ONLY
C          3.  AERO COEFFICIENTS DEPENDENT ONLY OF ATTITUDE
C          4.  CONSTANT MASS DISTRIBUTION
C          5.  AERO AND GRAVITY TORQUES
C
C
C
C***********************************************************************
      REAL IXYZ(3,3),IXYZI(3,3),LMRP(3),LANDA,LAMDE
      REAL IXYZX(3,3),LMRPX(3),LXC(3)
      DIMENSION IVAR(90),SCALE(90),PVALU(90),PNAME(90),CASE(18),
     $A1(3,3),A2(3,3),A(3),A3(3,3),IPW(90),PLT(90)
      DIMENSION ATMM(13),LI(3),MI(3),ABP(3,3)
C
C******************** C O M M O N    B L O C K***************************
C
C     PROGRAM CONTROL AND INTEGRATION (100)
C
      COMMON TIME,DT,DTP,DTSAM,TRUN,TEND,THOR,QSQR,PCDUM(12)
      COMMON PQRD(3),QD(4),VDOT(3),RDOT(3),XDDUM(27)
      COMMON PQR(3),G(4),V(3),R(3),XDUM(27)
C
C     VEHICLE CHARACTERISTICS COMMON (50)
C
      COMMON WT,XCG,YCG,ZCG,IXYZ,IXYZI,DREF,SREF,LMRP,GMASS,GMASI,
     $ XMRP,YMRP,ZMRP,VCCOM(18)
C
C     VEHICLE POSITION COMMON (150)
C
      COMMON PHIBK,ALPHT,PHIA,ARB(3,3),AIB(3,3),ARP(3,3),ABI(3,3),
     $ AID(3,3),AIG(3,3),AGR(3,3),ADR(3,3),ADI(3,3),ADG(3,3),
     $ AND(3,3),ARI(3,3),VPCOM(39)
C
C     EARTH POSITION COMMON (100)
C
      COMMON RMAG,RPSI,RI(3),G(3),CR,CW,GMAG,RR,RR2,SPSI,SPSI2,SJ,
```

```
     $ SD,SH,SWH,SWD,GCR,H,PSI,DPSI,PSID,LAMDA,LAMDE,SIGR,GAMR,
     $ SIGI,GAMI,RBI(3),EPCOM(66)
C
C     VEHICLE AERODYNAMICS COMMON (100)
C
      COMMON CA,CN,CL,CM,CY,CEN,FAR(3),FARC(3),CPLMN(3),CXYZ(3),
     $ CLMNT(3),FAB(3),FAI(3),FOMI(3),TOA(3),VACOM(67)
C
C
C     VEHICLE DYNAMICS COMMON (150)
C
      COMMON RHO,QBAR,VRMAG,VRELD(3),WIMAG,WID(3),VRELC(3),
     $ VRELI(3),VRELB(3),VATM(3),DIW(3),GGB(3),VIG(3),TAB(3),
     $ HB(3),WXH(3),TSUM(3),VDCOM(107)
C
C     THE FOLLOWING VARIABLES ARE NOT CHANGED FROM CASE TO CASE AND
C      ARE CONSIDERED AS CONSTANTS
C
C
C     ATMOSPHERE TABLES COMMON (500)
C
      COMMON TLAT(11),TLNG(37),FRHO(11,37),ATCOM(45)
C
C     TABLES OF AERODYNAMIC COEFFICIENTS COMMON (2000)
C
      COMMON TALP(11),TPHIA(21),FCN(21,11),FCA(21,11),
     $ FCY(21,11),FCL(21,11),FCM(21,11),FCEN(21,11),ACCOM(582)
C
C
C
C     CHANGELESS CONSTANTS COMMON(20)
C
      COMMON CH(3),TWOPI,OMGE,CNV,PI,CCCOM(13)
C
C
C     TOTAL POINTS IN COMMON BLOCK  =  3170
C     TOTAL POINTS IN VARYING COMMON = 650
C
C
      DATA BLANK/4H    /
C
C     NCASE = NO. OF CASES TO BE RUN PER CASE DECK
C
C     NX = NO. OF INTEGRATION VARIABLES
C
C
      CALL EBOMB(945)
      READ(5,1)NCASE,NX
    1 FORMAT(3I5)
      NR = 5
      NW = 6
      NP = 12
```

```
      NCOM = 650
      NCOM1 = 3170
C*****ZERO  COMPLETE COMMON BLOCK
      CALL ZCOM(NCOM1)
C*****READ  MAIN PRINT SPECIFICATIONS
      CALL PSPEC(5,6,12,NPV,IVAR,PNAME,SCALE,IPW)
C*****READ  IN ATMOSPHERIC DATA
    4 FORMAT(2F10.5,I5)
      READ(5,5)(ATNM(I),I=1,18)
    5 FORMAT(18A4)
      READ(5,300)(TLAT(I),I=1,11)
      READ(5,300)(TLNG(I),I=1,37)
      DO 337 J=1,37
  337 READ(5,6)(FRHO(I,J),I=1,11)
    6 FORMAT(6E9.3)
      CNV = 57.29577951
C*****READ  IN VEHICLE AERODYNAMIC DATA
      READ(NR,301)NALP,NPHIA
  301 FORMAT(5I5)
C
C***READ IN ALPHAT AND PHIA INDEPENDENT VARIABLE TABLES
C
  302 FORMAT(9F8.3)
      READ(NR,300)(TALP(I),I=1,NALP)
      READ(NR,300)(TPHIA(I),I=1,NPHIA)
  303 FORMAT(6E12.1)
C
C***AXIAL  FORCE COEFFICIENT TABLE
C
  300 FORMAT(7F8.3)
      READ(NR,300)((FCA(I,J),I=1,NPHIA),J=1,NALP)
C
C***NORMAL  FORCE COEFFICIENT TABLE
C
      READ(NR,300)((FCN(I,J),I=1,NPHIA),J=1,NALP)
C
C***SIDE  FORCE COEFFICIENT TABLE
C
      READ(NR,300)((FCY(I,J),I=1,NPHIA),J=1,NALP)
C
C***PITCHING  MOMENT COEFFICIENT TABLE
C
      READ(NR,300)((FCM(I,J),I=1,NPHIA),J=1,NALP)
C
C***YAWING MOMENT COEFFICIENT TABLE
C
      READ(NR,300)((FCEN(I,J),I=1,NPHIA),J=1,NALP)
C
C***ROLLING MOMENT COEFFICIENT TABLE
C
      READ(NR,300)((FCL(I,J),I=1,NPHIA),J=1,NALP)
C
```

```
      C
            DO 334 I=1,NALP
        334 TALP(I)=TALP(I)/CNV
            DO 335 I=1,NPHIA
        335 TPHIA(I)=TPHIA(I)/CNV
            DO 338 I=1,11
        338 TLAT(I)=TLAT(I)/CNV
            DO 339 I=1,37
        339 TLNG(I)=TLNG(I)/CNV
            PI=180./CNV
            TWOPI=2.*PI
            CM=1.40765391E16
            RE=20925721.785
            RS=20909734.843
            GF=0.6738525415E-02
            GJ=.162405E-02
            GH=-.640E-05
            GD=.69125E-05
            OMGE=7.29221158E-05
            CH0=3.3718120539E-03
            CH(1)=-1.57258544E-10
            CH(2)=5.907450825E-18
            CH(3)=-1.08416187E-25
          7 FORMAT(8E10.1)
            ICASE=0
      C*****BEGIN CASE LOOP
          8 CALL ZCOM(NCOM)
            ICASE=ICASE+1
            IF(ICASE.GT.1)GO TO 11
      C
      C*****READ CASE DEFINITION DATA
      C
      C
      C
            READ(5,5)(CASE(I),I=1,18)
            IF(CASE(1).EQ.BLANK)GO TO 94
            READ(5,2)PROPT
            READ(5, 2)DT,DTP,DTSAM,TRUN,TIME,IOPT1
          2 FORMAT(5F10.4,I5)
            DTX=DT
            DTPX=DTP
            DTSX=DTSAM
            TRUNX=TRUN
            TIMEX=TIME
            GO TO 12
         11 IF(ICASE.GT.NCASE)GO TO 90
            DT=DTX
            DTP=DTPX
            DTSAM=DTSX
            TRUN=TRUNX
            TIME=TIMEX
      C
```

```
C*****INITIALIZE VARIABLES
C
   12 TEND=TRUN+TIME
      TP=TIME
      TPR=TIME
      IF(ICASE.GT.1) GO TO 17
C*****READ VEHICLE DATA
      READ(5,13)WT,XCG,YCG,ZCG,XMRP,YMRP,ZMRP
   13 FORMAT(8F10.4)
      READ(5,13)DREF,SREF
      READ(5,7)IXYZ(1,1),IXYZ(2,2),IXYZ(3,3),IXYZ(1,2),IXYZ(1,3),
     $IXYZ(2,3)
      IXYZ(1,2)=-IXYZ(1,2)
      IXYZ(1,3)=-IXYZ(1,3)
      IXYZ(2,3)=-IXYZ(2,3)
      IXYZ(2,1)=IXYZ(1,2)
      IXYZ(3,1)=IXYZ(1,3)
      IXYZ(3,2)=IXYZ(2,3)
      LMRP(1)=(XMRP-XCG)/DREF
      LMRP(2)=(YMRP-YCG)/DREF
      LMRP(3)=(ZMRP-ZCG)/DREF
      WTX=WT
      DREFX=DRFF
      SREFX=SREF
      DO 14 I=1,3
      DO 14 J=1,3
      IXYZX(I,J)=IXYZ(I,J)
      LMRPX(I)=LMRP(I)
   14 CONTINUE
C
C*****POSITION INITIALIZATION
C
      READ(5,15)PSI,LAMDE,RMAG,VIMAG,SIGI,GAMI
      LAMDA=LAMDE+OMGE*TIMEX*CNV
   15 FORMAT(F10.6,F10.5,F10.2,F10.3,F10.6,F10.8)
      READ(5,16)PHIBI,ALPHAI,PHIAI,(PQR(I),I=1,3)
      READ(5,210)((ABP(I,J),J=1,3),I=1,3)
   16 FORMAT(6F10.5)
      PSIX=PSI
      XLAND=LAMDE
      RMAGX=RMAG
      VRELX=VIMAG
      SIGIX=SIGI
      GAMIX=GAMI
      PHIAIX=PHIAI
      ALPAIX=ALPHAI
      PHIBIX=PHIBI
      PP=PQR(1)
      PQ=PQR(2)
      PR=PQR(3)
      GO TO 19
   17 CONTINUE
```

```
      WT=WTX
      DREF=DREFX
      SREF=SREFX
      DO 18 I=1,3
      DO 18 J=1,3
      IXYZ(I,J)=IXYZX(I,J)
      LMRP(I)=LMRPX(I)
   18 CONTINUE
      PSI=PSIX
      LAMDA=XLAMD+OMGE*TIMEX*CNV
      RMAG=RMAGX
      VIMAG=VRELX
      SIGI=SIGIX
      GAMI=GAMIX
      PHIAI=PHIAIX
      ALPHAI=ALPAIX
      PHIBI=PHIBIX
      PQR(1)=PP
      PQR(2)=PQ
      PQR(3)=PR
   19 CONTINUE
      LAMDA=LAMDA/CNV
      LAMDE=LAMDE/CNV
      PSI=PSI/CNV
      SIGI=SIGI/CNV
      GAMI=GAMI/CNV
      PHIAI=PHIAI/CNV
      ALPHAI=ALPHAI/CNV
      PHIBI=PHIBI/CNV
      DO 20 I=1,3
   20 PQR(I)=PQR(I)/CNV
      GMASS=WT*RS*RS/GM
      GMASI=1./GMASS
      DO 21 I=1,3
      DO 21 J=1,3
   21 IXYZI(I,J)=IXYZ(I,J)
      CALL MINV(IXYZI,3,DI,LI,MI)
C
C     INERTIAL POSITION VECTOR COMPUTATION
C
      RPSI=RE/SQRT(1.+GF*(SIN(PSI)**2))
      H=RMAG-RPSI
      R(1)=RMAG*COS(PSI)*COS(LAMDA)
      R(2)=RMAG*COS(PSI)*SIN(LAMDA)
      R(3)=RMAG*SIN(PSI)
      DPSI=CHO
      HW=1.
      DO 22 I=1,3
      HW=HW*H
   22 DPSI=DPSI+CH(I)*HW
      DPSI=DPSI*SIN(2.0*PSI)
      PSID=PSI+DPSI
```

```
      C
      C*****GEODETIC COMPUTATIONS
      C
            CALL TWIST(LAMDA,A1,3)
            PSI90=PSID+90./CNV
            CALL TWIST(-PSI90,A2,2)
            DO 23 I=1,3
            DO 23 J=1,3
         23 AID(I,J)=A1(I,1)*A2(1,J)+A1(I,2)*A2(2,J)+A1(I,3)*A2(3,J)
      C
      C*****GEOCENTRIC COMPUTATIONS
      C
            PS90=PSI+90./CNV
            CALL TWIST(-PS90,A2,2)
            DO 24 I=1,3
            DO 24 J=1,3
         24 AIG(I,J)=A1(I,1)*A2(1,J)+A1(I,2)*A2(2,J)+A1(I,3)*A2(3,J)
            DO 350 I=1,3
            DO 350 J=1,3
        350 ADI(I,J)=AID(J,I)
      C
      C     D-FRAME INERTIAL VELOCITY COMPONENTS
      C
            VIG(1)=VINAG*COS(GAMI)*COS(SIGI)
            VIG(2)=VINAG*COS(GAMI)*SIN(SIGI)
            VIG(3)=-VINAG*SIN(GAMI)
            CALL TWIST(DPSI,ADG,2)
            DO 25 I=1,3
         25 VID(I)=ADG(I,1)*VIG(1)+ADG(I,2)*VIG(2)+ADG(I,3)*VIG(3)
      C
      C     COMPUTE INITIAL ATMOSPHERIC VELOCITY COMPONENTS(INERTIAL FRAME)
      C
            VATM(1)=-OMGE*R(2)
            VATM(2)=OMGE*R(1)
            VATM(3)=0.0
      C
      C     RESOLVE VECTORS INTO I-FRAME
      C
            DO 29 I=1,3
            V(I)=AID(I,1)*VID(1)+AID(I,2)*VID(2)+AID(I,3)*VID(3)
         29 A(I)=V(I)-VATM(I)
      C     D-FRAME RELATIVE VELOCITY COMPONENTS
            DO 351 I=1,3
        351 VRELD(I)=ADI(I,1)*A(1)+ADI(I,2)*A(2)+ADI(I,3)*A(3)
      C
      C     DETERMINE EULER MATRIX FROM RELATIVE VELOCITY FRAM TO BODY FRAME
      C
            CALL TWIST(-ALPHAI,A1,2)
            CALL TWIST(-PHIBI,A2,1)
            DO 30 I=1,3
            DO 30 J=1,3
         30 A3(I,J)=A1(I,1)*A2(1,J)+A1(I,2)*A2(2,J)+A1(I,3)*A2(3,J)
```

```
      CALL TWIST(-PHIAI,A1,1)
      DO 750 I=1,3
      DO 750 J=1,3
  750 A2(I,J)=A1(I,1)*A3(1,J)+A1(I,2)*A3(2,J)+A1(I,3)*A3(3,J)
      DO 31 I=1,3
      DO 31 J=1,3
   31 ARB(I,J)=A2(J,I)
C
C     DETERMINE EULER MATRIX FROM LOCAL FRAME TO RELATIVE VELOCITY FRAME
C
      GAMR=ATN2(-VRELD(3),SQRT(VRELD(1)*VRELD(1)+VRELD(2)*VRELD(2)))
      SIGR=ATN2(VRELD(2),VRELD(1))
      CALL TWIST(SIGR,A1,3)
      CALL TWIST(GAMR,A2,2)
   34 DO 35 I=1,3
      DO 35 J=1,3
   35 ADR(I,J)=A1(I,1)*A2(1,J)+A1(I,2)*A2(2,J)+A1(I,3)*A2(3,J)
      DO 37 I=1,3
      DO 37 J=1,3
   37 A1(I,J)=AID(I,1)*ADR(1,J)+AID(I,2)*ADR(2,J)+AID(I,3)*ADR(3,J)
      DO 38 I=1,3
      DO 38 J=1,3
   38 AIB(I,J)=A1(I,1)*ARB(1,J)+A1(I,2)*ARB(2,J)+A1(I,3)*ARB(3,J)
      DO 709 I=1,3
      DO 709 J=1,3
  709 ABI(I,J)=AIB(J,I)
      CALL EULR6(AIB,Q)
      WRITE(6,39)(CASE(I),I=1,18)
   39 FORMAT(1H1,18A4//)
      WRITE(6,40)ICASE
   40 FORMAT(1X,11HCASE NUMBER,I3/)
      WRITE(6,41)
   41 FORMAT(1X,119H-----------------------------------------------------------
     $-----------------------------------------------------------------------
     $-/)
      WRITE(6,42)
   42 FORMAT(1X,32H**********VEHICLE DATA**********/)
      WRITE(6,43)WT,XCG,YCG,ZCG,XMRP,YMRP,ZMRP,DREF,SREF
     $,((IXYZ(K,J),J=1,3),K=1,3)
   43 FORMAT(1X,8HWEIGHT =,F10.3,21X,15HCG LOCATION = (,F10.4,1H,,F10.4,
     $1H,,F10.4,1H)/1X,26HMOMENT REFERENCE POINT = (,F10.4,1H,,F10.4,1H,
     $,F10.4,1H)/1X,30HAERODYNAMIC REFERENCE LENGTH =,F10.4/1X,28HAERODY
     $NAMIC REFERENCE AREA =,F10.4
     $//1X,24HVEHICLE INERTIA MATRIX =,F10.1,5X,F10.1,5X,F10.1/25X,F10.1
     $,5X,F10.1,5X,F10.1/25X,F10.1,5X,F10.1,5X,F10.1//)
      WRITE(6,41)
      WRITE(6,44)
   44 FORMAT(1X,32H*******INITIAL CONDITIONS*******/)
      WRITE(6,45)RMAG,PSIX,XLAMD,VIMAG,SIGIX,GAMIX,PHIBIX,ALPAIX,PHIAIX,
     $PP,PQ,PR
      WRITE(6,211)((ABP(I,J),J=1,3),I=1,3)
   45 FORMAT(1X,10HRMAG      =,F10.0/1X,21HGEOCENTRIC LATITUDE =,F10.6/1X
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
     $,11HLONGITUDE =,F10.5/1X,19HINERTIAL VELOCITY =,F10.3/1X,18HINERTI
     $AL HEADING =,F10.5/1X,28HINERTIAL FLIGHT PATH ANGLE =,F10.6//1X,23
     $HVEHICLE ATTITUDE DATA -,1X,11HPHIBI  +1 =,F10.5/
     $ 25X,11H LPHAI +2 =,F10.5/25X,11HPHIAI +1  =,F10.5//1X,
     $18HINITIAL BODY RATES,
     $6X,11HROLL RATE =,F10.5/25X,12HPITCH RATE =,F10.5/25X,10HYAW RATE
     $=,F10.5//)
      WRITE(6,41)
      WRITE(6,46)
   46 FORMAT(1X,32H********OTHER INFORMATION*******/)
      WRITE(6,47)(ATNM(I),I=1,18)
   47 FORMAT(1X,20HTHIS PROGRAM USES   ,18A4/)
      WRITE(6,98)DT,DTP,DTSAM,TRUN,TIME,IOPT1
   98 FORMAT(1X,4HDELT,F10.6,5X,4HDTPR,F10.5,5X,4HDTSM,F10.2,5X,
     14HTRUN,F10.2,5X,4HTIME,F10.2,4X,5HIOPT1,I3)
  210 FORMAT(6E12.6/3E12.6)
  211 FORMAT(1X,4HABP ,5X,3E18.7/10X,3E18.7/10X,3E18.7/)
C
      IF(IOPT1.EQ.0) GO TO 600
      READ(14)(TIME,DT,(PQRD(I),PQR(I),VDOT(I),V(I),
     $RDOT(I),R(I),I=1,3),(QD(J),Q(J),J=1,4))
  600 CONTINUE
C*****BEGIN INTEGRATION LOOP
   51 CONTINUE
      DO 86  KUTTA=1,5
C
C*****EVALUATE DERIVATIVES
C
      RMAG=SQRT(R(1)*R(1)+R(2)*R(2)+R(3)*R(3))
      R1(1)=R(1)/RMAG
      R1(2)=R(2)/RMAG
      R1(3)=R(3)/RMAG
      GMAG=GM/(RMAG*RMAG)
      RR=RE/RMAG
      RR2=RR*RR
      SPSI=R1(3)
      SPSI2=SPSI*SPSI
      RPSI=RE/SQRT(1.+GF*SPSI2)
      SJ=1.-5.*SPSI2
      SD=1./7.-2.*SPSI2+3.*SPSI2*SPSI2
      SH=3.-7.*SPSI2
      SWH=-SJ/5.
      SWD=SH/7.
      CR=RR2*(GH*RR*SPSI*SH+3.*GD*RR2*SD+GJ*SJ)+1.
      CW=RR2*(3.*GH*RR*SWH+4.*GD*RR2*SPSI*SWD+2.*GJ*SPSI)
      GCR=-GMAG*CR
      G(1)=GCR*R1(1)
      G(2)=GCR*R1(2)
      G(3)=-GMAG*CW+GCR*R1(3)
      H=RMAG-RPSI
      PSI=ATN2(R(3),SQRT(R(1)*R(1)+R(2)*R(2)))
      N=3
```

```
      DPSI=CH0
      HW=1.
      DO 53 I=1,N
      HW=HW*H
   53 DPSI=DPSI+CH(I)*HW
      DPSI=DPSI*SIN(2.0*PSI)
      PSID=PSI+DPSI
      PSI90=PSID+90./CNV
      LAMDA=ATN2(R(2),R(1))
      CALL TWIST(LAMDA,A3,3)
      CALL TWIST(-PSI90,A2,2)
      DO 54 I=1,3
      DO 54 J=1,3
   54 AID(I,J)=A3(I,1)*A2(1,J)+A3(I,2)*A2(2,J)+A3(I,3)*A2(3,J)
      PSG90=PSI+90./CNV
      CALL TWIST(-PSG90,A2,2)
      DO 58 I=1,3
      DO 58 J=1,3
   58 AIG(I,J)=A3(I,1)*A2(1,J)+A3(I,2)*A2(2,J)+A3(I,3)*A2(3,J)
      VATM(1)=-OMGE*R(2)
      VATM(2)=OMGE*R(1)
      VATM(3)=0.0
      DO 55 I=1,3
   55 VRELI(I)=V(I)-VATM(I)
      VRMAG=SQRT(VRELI(1)*VRELI(1)+VRELI(2)*VRELI(2)+VRELI(3)*VRELI(3))
      CALL EULR5(Q,AIB)
      DO 56 I=1,3
      DO 56 J=1,3
   56 ABI(I,J)=AIB(J,I)
      DO 57 I=1,3
      RB1(I)=-ABI(I,1)*R1(1)-ABI(I,2)*R1(2)-ABI(I,3)*R1(3)
   57 VRELB(I)=ABI(I,1)*VRELI(1)+ABI(I,2)*VRELI(2)+ABI(I,3)*VRELI(3)
C***********************************************************************
C
C    AERODYNAMIC FORCES AND MOMENTS
C
C***********************************************************************
C
      IF(LAMDA.LT.0.)  LAMDA=LAMDA+TWOPI
      CALL LOOK2(RHO,FRHO,11,PSI,TLAT,11,LAMDA,TLNG,37,IPSI,ILNG,
     $ 1)
      RHO=RHO*1.9349971E-03
      QBAR=0.5*RHO*VRMAG*VRMAG
      ALPHT=ATN2(SQRT(VRELB(2)*VRELB(2)+VRELB(3)*VRELB(3)),VRELB(1))
      PHIA=ATN2(VRELB(2),VRELB(3))
      IF(PHIA.LT.0.0)PHIA=PHIA+TWOPI
  501 CONTINUE
C
C
C*****MAIN BODY COEFFICIENTS
C
      CALL LOOK2(CN,FCN,21,PHIA,TPHIA,NPHIA,ALPHT,TALP,NALP,IPHIA,IALP,
```

```
      $ 1)
       CALL LOOK2 CA,FCA,21,PHIA,TPHIA,NPHIA,ALPHT,TALP,NALP,IPHIA,IALP,
      $ 0)
       CALL LOOK2(CM,FCM,21,PHIA,TPHIA,NPHIA,ALPHT,TALP,NALP,IPHIA,IALP,
      $ 0)
       CALL LOOK2(CY,FCY,21,PHIA,TPHIA,NPHIA,ALPHT,TALP,NALP,IPHIA,IALP,
      $ 0)
       CALL LOOK2(CL,FCL,21,PHIA,TPHIA,NPHIA,ALPHT,TALP,NALP,IPHIA,IALP,
      $ 0)
       CALL LOOK2(CEN,FCEN,21,PHIA,TPHIA,NPHIA,ALPHT,TALP,NALP,IPHIA,IALP,
      $0)
       CPLMN(1)=CL
       CPLMN(2)=CM
       CPLMN(3)=CEN
       CXYZ(1)=-CA
       CXYZ(2)=CY
       CXYZ(3)=-CN
       LXC(1)=LMRP(2)*CXYZ(3)-LMRP(3)*CXYZ(2)
       LXC(2)=LMRP(3)*CXYZ(1)-LMRP(1)*CXYZ(3)
       LXC(3)=LMRP(1)*CXYZ(2)-LMRP(2)*CXYZ(1)
       DO 63 I=1,3
       DIW(I)=IXYZ(I,1)*RB1(1)+IXYZ(I,2)*RB1(2)+IXYZ(I,3)*RB1(3)
    63 CLMNT(I)=CPLMN(I)+LXC(I)
       GGB(1)=RB1(2)*DIW(3)-RB1(3)*DIW(2)
       GGB(2)=RB1(3)*DIW(1)-RB1(1)*DIW(3)
       GGB(3)=RB1(1)*DIW(2)-RB1(2)*DIW(1)
       GGB(1)=GGB(1)*(GM*3.0/RMAG**2)/RMAG
       GGB(2)=GGB(2)*(GM*3.0/RMAG**2)/RMAG
       GGB(3)=GGB(3)*(GM*3.0/RMAG**2)/RMAG
       QS=QBAR*SREF
       QSD=QS*DREF
       DO 64 I=1,3
       FAB(I)=CXYZ(I)*QS
       TOA(I)=CLMNT(I)*QSD
    64 TAB(I)=TOA(I)+GGB(I)
C****************************************************************************
C
C     RIGID BODY DYNAMICS
C
C****************************************************************************
       DO 65 I=1,3
    65 HB(I)=IXYZ(I,1)*PQR(1)+IXYZ(I,2)*PQR(2)+IXYZ(I,3)*PQR(3)
       WXH(1)=PQR(2)*HB(3)-PQR(3)*HB(2)
       WXH(2)=PQR(3)*HB(1)-PQR(1)*HB(3)
       WXH(3)=PQR(1)*HB(2)-PQR(2)*HB(1)
       DO 66 I=1,3
    66 TSUM(I)=TAB(I)-WXH(I)
       DO 67 I=1,3
    67 PQRD(I)=IXYZI(I,1)*TSUM(1)+IXYZI(I,2)*TSUM(2)+IXYZI(I,3)*TSUM(3)
       QD(1)= .5*(Q(4)*PQR(1)-Q(3)*PQR(2)+Q(2)*PQR(3))
       QD(2)= .5*(Q(3)*PQR(1)+Q(4)*PQR(2)-Q(1)*PQR(3))
       QD(3)= .5*(-Q(2)*PQR(1)+Q(1)*PQR(2)+Q(4)*PQR(3))
```

```
      QD(4)=-.5*(Q(1)*PQR(1)+Q(2)*PQR(2)+Q(3)*PQR(3))
      QSQR=SQRT(Q(1)**2+Q(2)**2+Q(3)**2+Q(4)**2)
      DO 68 I=1,3
      FAI(I)=AIB(I,1)*FAB(1)+AIB(I,2)*FAB(2)+AIB(I,3)*FAB(3)
      FOMI(I)=FAI(I)*GMASI
      VDOT(I)=FOMI(I)+G(I)
   68 RDOT(I)=V(I)
      IF(KUTTA-1) 86,69,86
C
C*****PERFORM END-OF-DTSTEP COMPUTATIONS
   69 CONTINUE
C
      LANDE=ATN2(R(2),R(1))-OMGE*TIME
  497 IF(LANDE)498,499,499
  498 LANDE=LANDE+TWOPI
      GO TO 497
  499 CONTINUE
      DO 101 I=1,3
      DO 101 J=1,3
  101 ADI(I,J)=AID(J,I)
      DO 103 I=1,3
      VIG(I)=AIG(1,I)*V(1)+AIG(2,I)*V(2)+AIG(3,I)*V(3)
  103 VID(I)=ADI(I,1)*V(1)+ADI(I,2)*V(2)+ADI(I,3)*V(3)
      VIMAG=SQRT(V(1)*V(1)+V(2)*V(2)+V(3)*V(3))
      DO 102 I=1,3
  102 VRELD(I)=ADI(I,1)*VRELI(1)+ADI(I,2)*VRELI(2)+ADI(I,3)*VRELI(3)
      GAMR=ATN2(-VRELD(3),SQRT(VRELD(1)*VRELD(1)+VRELD(2)*VRELD(2)))      OUT 0450
      SIGR=ATN2(VRELD(2),VRELD(1))                                        OUT 0460
      GAMI=ATN2(-VIG(3),SQRT(VIG(1)*VIG(1)+VIG(2)*VIG(2)))                OUT 0480
      SIGI=ATN2(VIG(2),VIG(1))                                            OUT 0490
      CALL TWIST(SIGR,A1,3)
      CALL TWIST(GAMR,A2,2)
      DO 104 I=1,3
      DO 104 J=1,3
  104 ADR(I,J)=A1(I,1)*A2(1,J)+A1(I,2)*A2(2,J)+A1(I,3)*A2(3,J)
      DO 105 I=1,3
      DO 105 J=1,3
  105 ARD(I,J)=ADR(J,I)
      DO 106 I=1,3
      DO 106 J=1,3
  106 ARI(I,J)=ARD(I,1)*ADI(1,J)+ARD(I,2)*ADI(2,J)+ARD(I,3)*ADI(3,J)
      DO 107 I=1,3
      DO 107 J=1,3
  107 ARB(I,J)=ARI(I,1)*AIB(1,J)+ARI(I,2)*AIB(2,J)+ARI(I,3)*AIB(3,J)
      DO 212 I=1,3
      DO 212 J=1,3
  212 ARP(I,J)=ARB(I,1)*ABP(1,J)+ARB(I,2)*ABP(2,J)+ARB(I,3)*ABP(3,J)
      PHIBK=ATN2(ARB(2,1),-ARB(3,1))
      DO 110 I=1,3
      FAR(I)=ARB(I,1)*FAB(1)+ARB(I,2)*FAB(2)+ARB(I,3)*FAB(3)
  110 FARC(I)=FAR(I)/GS
C
```

```
C*****PRINT IF APPROPRIATE
C
      IF(TIME-TP)74,73,73
C
   73 CONTINUE
      THOR=TIME/3600.0
      NXPR=1
      IF(TIME-TPR)120,121,121
  120 NXPR=-1
  121 CONTINUE
      IF(TIME.LT.DTSAM) GO TO 601
      WRITE(13)(TIME,DT,(PQRD(I),PQR(I),VDOT(I),V(I),
     $RDOT(I),P(I),I=1,3),(QD(J),Q(J),J=1,4))
      DTSAM=DTSAM+TEND
      NXPR=1
  601 CONTINUE
      IF(NXPR.GT.0)TPR=TIME+PROPT*DTP
      IF(TIME.GE.TEND)NXPR=1
      CALL BLOCK( 6,NP,CASE,NPV,IVAR,PNAME,PVALU,SCALE,IPW,NXPR)
      TP=TIME+DTP
C
C*****CHECK WHETHER THIS CASE SHOULD BE TERMINATED
C
   74 CONTINUE
      IF(TIME-TEND)86,87,87
C
C*****INTEGRATE DYNAMIC VARIABLES
C
   86 CALL RUNK2(KUTTA,NX)
      GO TO 51
C
C*****TERMINATE THIS CASE
C
   87 CONTINUE
   90 IF(NCASE-20)91,91,94
   91 ICASE=0
      GO TO 8
   94 STOP
      END
```

```
      SUBROUTINE PSPEC(NR,NW,NP,NPV,IVAR,PNAME,SCALE,IPW)               PSP 001
C***********************************************************************PSP 002
C                                                                       PSP 003
C     PRINT SPECIFICATIONS SUBROUTINE                                   PSP 004
C     DEFINES THE VARIABLES TO BE BLOCK-PRINTED OR PLOTTED              PSP 005
C                                                                       PSP 006
C            NR    = CARD READER LOGICAL UNIT NUMBER, =5 FOR IBM-7044   PSP 007
C            NW    = LINE PRINTER LOGICAL UNIT NUMBER, =6 FOR IBM-7044  PSP 008
C            NP    = PLOT FILE LOGICAL UNIT NUMBER, = 12
C            NPV   = NUMBER OF VARIABLES TO BE PRINTED  OR PLOTTED      PSP 009
C            IVAR  = LOCATION ARRAY IN COMMON BLOCK OF PRINT VARIABLES  PSP 010
C            PNAME = ALPHAMERIC NAMES TO BE ASSIGNED TO PRINTED VARIABLES PSP 011
C                    IN INPUT DATA
C            SCALE = SCALE FACTOR ARRAY TO BE APPLIED TO PRINTED VARIABLES PSP 012
C            ISCAL = SCALE FACTOR DESIGNATION                           PSP 013
C                  IF ISCAL = 0, SCALE = 1.                             PSP 014
                            = 1,        = 57.29578                      PSP 015
C                           = 2,        = SCAL (READ FROM DATA CARDS)
C            IPW = PRINT-PLOT FLAG
C                  IF IPW = 0, PRINT AND PLOT
C                         = 1, PRINT ONLY
C                         = 2, PLOT ONLY
C                                                                       PSP 017
C***********************************************************************PSP 018
      DIMENSION IVAR(90),PNAME(90),SCALE(90),IPW(90)                    PSP 019
      CNV=57.29578                                                      PSP 020
      I=0                                                               PSP 021
   10 I=I+1                                                             PSP 022
      READ(NR,114) IVAR(I),ISCAL,SCAL,PNAME(I),IPW(I)                   PSP 023
  114 FORMAT(I3,1X,I1,1X,E12.5,A4,I1)                                   PSP 024
      N=IVAR(I)                                                         PSP 025
      IF(N) 70,70,20                                                    PSP 026
   20 IF(ISCAL)30,30,40                                                 PSP 027
   30 SCALE(I)=1.                                                       PSP 028
      GO TO 10                                                          PSP 029
   40 IF(ISCAL-1)60,50,60                                               PSP 030
   50 SCALE(I)=CNV                                                      PSP 031
      GO TO 10                                                          PSP 032
   60 SCALE(I)=SCAL                                                     PSP 033
      GO TO 10                                                          PSP 034
   70 NPV=I-1                                                           PSP 035
      RETURN                                                            PSP 036
      END                                                               PSP 037
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE PCOM(NW,NCOM)
C***********************************************************************PCO 002
C                                                                       PCO 003
C     PCOM-PRINTS ALL LOCATIONS IN THE COMMON BLOCK                     PCO 004
C                                                                       PCO 005
C         NW    = LINE PRINTER LOGICAL UNIT NUMBER                      PCO 006
C         NCOM = NUMBER OF LOCATIONS IN COMMON BLOCK                    PCO 007
C                                                                       PCO 008
C***********************************************************************PCO 009
      COMMON VALU(1)                                                    PCO 010
      WRITE(NW,2)                                                       PCO 011
    2 FORMAT(/)                                                         PCO 012
      WRITE(NW,1) (VALU(I),I=1,100)                                     PCO 013
    1 FORMAT( 2(10E11.3/)/,4(10E11.3/)/,4(10E11.3/))                    PCO 014
      WRITE(NW,3) (VALU(I),I=101,NCOM)                                  PCO 015
    3 FORMAT( 5(10E11.3/))                                              PCO 016
      RETURN                                                            PCO 018
      END                                                               PCO 019
```

```
      SUBROUTINE EULR6(A,Q)                                       ER6 0010
C     COMPUTES EULER PARAMETERS FROM ROTATION MATRIX              ER6 0020
C     SENSE IS A=TA'                                              ER6 0030
      DIMENSION A(9),Q(4)                                         ER6 0040
      X=A(1)+A(5)+A(9)+1.                                         ER6 0050
      IF(X.LT.1E-8.AND.X.GT.-1E-8) X=0.0
      Z4=SQRT(X)
      IF(X)10,10,60                                               ER6 0060
   10 D=0.                                                        ER6 0070
      X=1.+A(1)-X                                                 ER6 0080
      IF(X.LT.1E-8.AND.X.GT.-1E-8) X=0.0
      IF(X)20,20,50                                               ER6 0090
   20 Z1=0.                                                       ER6 0100
      X=A(5)+1.                                                   ER6 0110
      IF(X.LT.1E-8.AND.X.GT.-1E-8) X=0.0
      IF(X)30,30,40                                               ER6 0120
   30 Z2=0.                                                       ER6 0130
      Z3=2.                                                       ER6 0140
      GO TO 70                                                    ER6 0150
   40 Z2=SQRT(2.*X)                                               ER6 0160
      Z3=(A(6)+A(8))/Z2                                           ER6 0170
      GO TO 70                                                    ER6 0180
   50 Z1=SQRT(2.*X)                                               ER6 0190
      Z2=(A(2)+A(4))/Z1                                           ER6 0200
      Z3=(A(3)+A(7))/Z1                                           ER6 0210
      GO TO 70                                                    ER6 0220
   60 Z4=SQRT(X)                                                  ER6 0230
      Z1=(A(6)-A(8))/Z4                                           ER6 0240
      Z2=(A(7)-A(3))/Z4                                           ER6 0250
      Z3=(A(2)-A(4))/Z4                                           ER6 0260
   70 Q(1)=.5*Z1                                                  ER6 0270
      Q(2)=.5*Z2                                                  ER6 0280
      Q(3)=.5*Z3                                                  ER6 0290
      Q(4)=.5*Z4                                                  ER6 0300
      RETURN                                                      ER6 0310
      END                                                         ER6 0320
```

```
      SUBROUTINE EULR5(Q,A)                                             ER5 0010
C   - COMPUTES ROTATION MATRIX FROM EULER PARAMETERS                    ER5 0020
C     SENSE IS  R=AR'                                                   ER5 0030
      DIMENSION Q(4),A(3,3)                                             ER5 0040
      P=Q(4)*Q(4)-0.5                                                   ER5 0050
      DO 50 I=1,3                                                       ER5 0060
      DO 50 J=1,3                                                       ER5 0070
      X=Q(I)*Q(J)                                                       ER5 0080
      KGN=2*(J-I)                                                       ER5 0090
      IF(KGN)10,40,20                                                   ER5 0100
   10 SGN=KGN+3                                                         ER5 0110
      GO TO 30                                                          ER5 0120
   20 SGN=KGN-3                                                         ER5 0130
   30 KGN=6-I-J                                                         ER5 0140
      X=X+SGN*Q(KGN)*Q(4)                                               ER5 0150
      GO TO 50                                                          ER5 0160
   40 X=X+P                                                             ER5 0170
   50 A(I,J)=2.0*X                                                      ER5 0180
      RETURN                                                            ER5 0190
      END                                                               ER5 0200
```

```
      SUBROUTINE LOOK1(F,FT,X,XT,NX,IX,MULT)                             LK1 0010
C************************************************************************LK1 0020
C     ONE-DIMENSIONAL TABLE LOOKUP ROUTINE  (* = RETURNED VALUES)        LK1 0030
C      *F          =FT(X)       (            )                           LK1 0040
C       FT         FUNCTION TABLE  FT(NR)                                LK1 0050
C       NR         NO. OF ROWS IN FT-TABLE                               LK1 0060
C       NC         NO. OF COLUMNS IN FT-TABLE                            LK1 0070
C       X          WORKING VALUE  OF INDEPENDENT VARIABLE                LK1 0080
C       XT         INDEPENDENT VARIABLE TABLE                            LK1 0090
C       NX         DIMENSION OF XT,TABLE                                 LK1 0100
C      *IX         X      INDEX   ON PREVIOUS LOOKUP (UPDATED ON EACH CALL) LK1 0110
C       MULT       =0, USE PREVIOUS X                                    LK1 0120
C       MULT       =1, LOOK UP NEW X,                                    LK1 0130
C************************************************************************LK1 0140
      DIMENSION FT(1),XT(1)                                              LK1 0150
      IF(MULT-1)2,1,2                                                    LK1 0160
    1 CALL INDEX(X,XT,NX,IX,KEEP,RX)                                     LK1 0170
    2 F=FT(IX)                                                           LK1 0180
      IF(KEEP-1)3,4,3                                                    LK1 0190
    3 F=(FT(IX+1)-F)*RX+F                                                LK1 0200
    4 RETURN                                                             LK1 0210
      END                                                                LK1 0220
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      SUBROUTINE LOOK2(F,FT,NR,X,XT,NX,Y,YT,NY,IX,IY,MULT)             LK2 0010
C***********************************************************************LK2 0020
C    TWO-DIMENSIONAL TABLE LOOKUP ROUTINE  (* = RETURNED VALUES)        LK2 0030
C     F          =FT(X,Y)     (          )                               LK2 0040
C     FT         FUNCTION TABLE  FT(NR,NC)                               LK2 0050
C     NR         NO. OF ROWS IN FT-TABLE                                 LK2 0060
C     NC         NO. OF COLUMNS IN FT-TABLE                              LK2 0070
C     X,Y        WORKING VALUES OF INDEPENDENT VARIABLES                 LK2 0080
C     XT,YT      INDEPENDENT VARIABLE TABLES                             LK2 0090
C     NX,NY      DIMENSION OF XT,YT,TABLES                               LK2 0100
C    *IX,IY      X,Y   INDEXES ON PREVIOUS LOOKUP (UPDATED ON EACH CALL) LK2 0110
C     MULT       =0, USE PREVIOUS X,Y                                    LK2 0120
C     MULT       =1, LOOK UP NEW X,Y,                                    LK2 0130
C***********************************************************************LK2 0140
      DIMENSION FT(1),XT(1),YT(1)                                        LK2 0150
      IF(MULT-1)2,1,2                                                    LK2 0160
    1 CALL INDEX(Y,YT,NY,IY,KEEP,RY)                                     LK2 0170
      I=1+(IY-1)*NR                                                      LK2 0180
      II=I+NR                                                            LK2 0190
    2 CALL LOOK1(F,FT(I),X,XT,NX,IX,MULT)                                LK2 0200
      IF(KEEP-1)3,4,3                                                    LK2 0210
    3 CALL LOOK1(F2,FT(II),X,XT,NX,IX,1)                                 LK2 0220
      F=(F2-F)*RY+F                                                      LK2 0230
    4 RETURN                                                             LK2 0240
      END                                                                LK2 0250
```

```
      SUBROUTINE INDEX(X,XT,NX,IX,KEEP,RATIO)                           IND 0010
C*********************************************************************IND 0020
C  TABLE INDEX LOOKUP ROUTINE   (* = RETURNED VALUES)                   IND 0030
C     X           WORKING VALUE OF INDEPENDENT VARIABLE                 IND 0040
C     XT          INDEPENDENT VARIABLE TABLE  XT(NX)                    IND 0050
C     NX          DIMENSION OF X-TABLE                                  IND 0060
C     IX          INDEX OF PREVIOUS LOOKUP (UPDATED ON EACH CALL)       IND 0070
C    *KEEP        =0 - X.NE.ANY XT. INTERPOLATION WAS REQUIRED.         IND 0080
C                 =1 - X.EQ.XT(IX). INTERPOLATION NOT REQUIRED.         IND 0090
C    *RATIO       INTERPOLATION RATIO                                   IND 0100
C*********************************************************************IND 0110
      DIMENSION XT(1)                                                   IND 0120
      IX=0
      RATIO=0.                                                          IND 0130
      KEEP=0                                                            IND 0140
      IF(NX-IX)1,1,2                                                    IND 0150
    1 IX=NX-1                                                           IND 0160
    2 IF(IX-1)3,3,4                                                     IND 0170
    3 IX=1                                                              IND 0180
    4 IF(NX-1)10,10,5                                                   IND 0190
    5 IF(XT(IX)-X)6,10,7                                                IND 0200
    6 IF(XT(IX+1)-X)8,9,11                                              IND 0210
    7 IX=IX-1                                                           IND 0220
      IF(IX-1)9,5,5                                                     IND 0230
    8 IX=IX+1                                                           IND 0240
      IF(IX-NX)5,10,10                                                  IND 0250
    9 IX=IX+1                                                           IND 0260
   10 KEEP=1                                                            IND 0270
      RETURN                                                            IND 0280
   11 RATIO=(X-XT(IX))/(XT(IX+1)-XT(IX))                                IND 0290
      RETURN                                                            IND 0300
      END                                                               IND 0310
```

```
      FUNCTION ATN2(S,C)                                                ATN 0010
C***********************************************************************ATN 0020
C                                                                      *ATN 0030
C     ATN2 IS A SINGLE PRECISION ROUTINE                               *ATN 0040
C     ARCTAN IS IN RADIANS -PI TO +PI                                  *ATN 0050
C     S=NUMERATOR                                                      *ATN 0060
C     C=DENOMINATOR                                                    *ATN 0070
C     SYSTEM ROUTINES USED ARE ATAN,SIGN                               *ATN 0080
C     ROUTINE IS VALID FOR ALL VALUES OF S AND C                       *ATN 0090
C     IF S AND C ARE BOTH 0.0,THE VALUE RETURNED IS ZERO.              *ATN 0100
C                                                                      *ATN 0110
C***********************************************************************ATN 0120
      ATN2=0.                                                           ATN 0130
      IF(C)4,2,5                                                        ATN 0140
    2 IF(S)3,7,3                                                        ATN 0150
    3 ATN2=SIGN(1.5707963268,S)                                         ATN 0160
      RETURN                                                            ATN 0170
    4 ATN2=SIGN(3.1415926536,S)                                         ATN 0180
    5 ATN2=ATN2+ATAN(S/C)                                               ATN 0190
    7 RETURN                                                            ATN 0200
      END                                                               ATN 0210
```

```
      SUBROUTINE ZCON (NCOM)                                            ZCO 001
C***********************************************************************ZCO 002
C                                                                       ZCO 003
C     SUBROUTINE TO ZERO ALL LOCATIONS IN COMMON BLOCK                  ZCO 004
C                                                                       ZCO 005
C         NCOM = NUMBER OF LOCATIONS OF COMMON BLOCK                    ZCO 006
C                                                                       ZCG 007
C***********************************************************************ZCO 008
      COMMON VALU(1)                                                    ZCO 009
      DO 10 I=1,NCOM                                                    ZCO 010
   10 VALU(I)=0.                                                        ZCO 011
      RETURN                                                            ZCO 012
      END                                                               ZCO 013
```

```
      SUBROUTINE TWIST(THETA,T,N)                                       TWI 001
C***********************************************************************TWI 002
C                                                                       TWI 003
C     TWIST COMPUTES THE TRANSFORMATION MATRIX T FOR A ROTATION ABOUT   TWI 004
C     ONE OF THE 3 AXES IN THE SENSE                                    TWI 005
C                                                                       TWI 006
C         R = T R'                                                      TWI 007
C                                                                       TWI 008
C         THETA = ANGLE OF ROTATION, IN RADIANS                         TWI 009
C         T     = 3X3 ROTATION MATRIX OUTPUT                            TWI 010
C         N     = AXIS NO. FOR WHICH ROTATION IS CALLED (1,2 OR 3)      TWI 011
C                                                                       TWI 012
C***********************************************************************TWI 013
      DIMENSION T(3,3)                                                  TWI 014
      ZERO=0.                                                           TWI 015
      ONE=1.                                                            TWI 016
      CTHET= COS(THETA)                                                 TWI 017
      DO 90 I=1,3                                                       TWI 018
      DO 90 J=I,3                                                       TWI 019
      IF(I-J)10,60,10                                                   TWI 020
   10 IF((I-N)*(J-N))30,20,30                                           TWI 021
   20 X=ZERO                                                            TWI 022
      GO TO 50                                                          TWI 023
   30 X= SIN(THETA)                                                     TWI 024
      IF(N-2)50,40,50                                                   TWI 025
   40 X=-X                                                              TWI 026
   50 T(I,J)=-X                                                         TWI 027
      GO TO 90                                                          TWI 028
   60 IF(I-N)70,80,70                                                   TWI 029
   70 X=CTHET                                                           TWI 030
      GO TO 90                                                          TWI 031
   80 X=ONE                                                             TWI 032
   90 T(J,I)=X                                                          TWI 033
      RETURN                                                            TWI 034
      END                                                               TWI 035
```

```
      SUBROUTINE AUTO(X,E,KONTRL)
      DIMENSION X(40),E(40)
      ERW=SQRT(E(1)*E(1)+E(2)*E(2)+E(3)*E(3))
      ERQ=SQRT(E(4)*E(4)+E(5)*E(5)+E(6)*E(6)+E(7)*E(7))
      ERV=SQRT(E(8)*E(8)+E(9)*E(9)+E(10)*E(10))
      ERR=SQRT(E(11)*E(11)+E(12)*E(12)+E(13)*E(13))
      ER=SQRT(ERW*ERW+ERQ*ERQ)
      KONTRL=1
      IF(ER.GT.32.E-5)KONTRL=2
      IF(ER.LE..5E-5)KONTRL=3
      RETURN
      END
```

```
      SUBROUTINE BLOCK(NW,NP,CASE,NPV,IVAR,PNAME,PVALU,SCALE,IPW,NXPR)  BLO 001
C***********************************************************************BLO 002
C                                                                       BLO 003
C     STANDARD BLOCK PRINTOUT ROUTINE                                   BLO 004
C                                                                       BLO 005
C            NW    = LINE PRINTER LOGICAL UNIT NUMBER, =6 FOR IBM-7044  BLO 006
C            NP    = PLOT FILE LOGICAL UNIT NUMBER, = 12
C            CASE  = ALPHANERIC CASE TITLE                              BLO 007
C            NPV   = NUMBER OF PRINT OR PLOT VARIABLES                  BLO 008
C            IVAR  = LOCATION ARRAY OF PRINT VARIABLES IN COMMON BLOCK  BLO 009
C            PNAME = ALPHAMERIC NAMES OF PRINT OR PLOT VARIABLES        BLO 010
C                    IN INPUT DATA
C            PRNM  = ALPHAMERIC NAMES OF PRINT VARIABLES
C            PVALU = SCALED PRINT OR PLOT VALUES                        BLO 011
C            SCALE = SCALE FACTORS TO BE APPLIED TO PRINT VARIABLES     BLO 012
C            IPW = PRINT-PLOT FLAG
C                        IF IPW = 0, PRINT AND PLOT
C                               = 1, PRINT ONLY
C                               = 2, PLOT ONLY
C                                                                       BLO 013
C***********************************************************************BLO 014
      DIMENSION CASE(18),IVAR(90),PVALU(90),SCALE(90),PNAME(90)         BLO 015
     1,IPW(90),PLT(90),PRNM(90)
      COMMON VALU(400)                                                  BLO 016
C                                                                       BLO 017
C*****SET UP PRINT ARRAY                                                BLO 018
      I=0                                                               BLO 019
      K=1
      L=1
   10 I=I+1                                                             BLO 020
      J=IVAR(I)                                                         BLO 021
      IF(IPW(I)-1) 5,6,7
    5 PVALU(L)=SCALE(I)*VALU(J)                                         BLO 022
      PRNM(L)=PNAME(I)
      PLT(K)=PVALU(L)
      K=K+1
      L=L+1
      IF(I-NPV) 10,20,20                                                BLO 023
    6 PVALU(L)=SCALE(I)*VALU(J)                                         BLO 022
      PRNM(L)=PNAME(I)
      L=L+1
      IF(I-NPV) 10,20,20                                                BLO 023
    7 PLT(K) = SCALE(I)*VALU(J)
      K=K+1
      IF(I-NPV) 10,20,20                                                BLO 023
C*****PRINT OUTPUT BLOCK                                                BLO 024
   20 IF(NXPR.LT.0) GO TO 30
      WRITE(NW,120) VALU(1),( CASE(J),J=1,18)                           BLO 025
      WRITE(NW,110) ((PRNM(I)),PVALU(I),I=1,L-1)                        BLO 026
   30 CONTINUE
      WRITE(NP) VALU(1),(PLT(I),I=1,K-1)
  120 FORMAT(//1X,6HTIME   ,1 E15.8,2X,18A4,/)                          BLO 027
```

```
  110 FORMAT(6(2X,A4,E14.8))                                   BLO 028
      RETURN                                                   BLO 029
      END                                                      BLO 030
```

```
      SUBROUTINE MINV(A,N,D,L,M)                                        MIN 0001
C***********************************************************************MIN 0002
C                                                                       MIN 0003
C     MATRIX INVERSE, OF AN NXN MATRIX-A INTO THE SAME MATRIX-A        MIN 0004
C                                                                       MIN 0005
C            -1                          D = OUTPUT DETERMINANT OF ORIGINAL A MIN 0006
C         A    = A                       L, M ARE TWO WORKING INTEGER VECTORS MIN 0007
C                                           DIMENSIONED IN MAIN, OF LENGTH NMIN 0008
C                                                                       MIN 0009
C     NOTE THAT THE ORIGINAL A-MATRIX IS DESTROYED                      MIN 0010
C     ALSO NOTE THAT A CHECK ON D SHOULD BE MADE IN CASE MATRIX-A EVER  MIN 0011
C                                                     BECOMES SINGULAR  MIN 0012
C                                                                       MIN 0013
C***********************************************************************MIN 0014
      DIMENSION A(1),L(1),M(1)                                          MIN 0015
C        SEARCH FOR LARGEST ELEMENT                                     MIN 0016
      D=1.0                                                             MIN 0017
      NK=-N                                                             MIN 0018
      DO 80 K=1,N                                                       MIN 0019
      NK=NK+N                                                           MIN 0020
      L(K)=K                                                            MIN 0021
      M(K)=K                                                            MIN 0022
      KK=NK+K                                                           MIN 0023
      BIGA=A(KK)                                                        MIN 0024
      DO 20 J=K,N                                                       MIN 0025
      IZ=N*(J-1)                                                        MIN 0026
      DO 20 I=K,N                                                       MIN 0027
      IJ=IZ+I                                                           MIN 0028
      IF( ABS(BIGA)- ABS(A(IJ))) 15,20,20                               MIN 0029
   15 BIGA=A(IJ)                                                        MIN 0030
      L(K)=I                                                            MIN 0031
      M(K)=J                                                            MIN 0032
   20 CONTINUE                                                          MIN 0033
C        INTERCHANGE ROWS                                               MIN 0034
      J=L(K)                                                            MIN 0035
      IF(J-K) 35,35,25                                                  MIN 0036
   25 KI=K-N                                                            MIN 0037
      DO 30 I=1,N                                                       MIN 0038
      KI=KI+N                                                           MIN 0039
      HOLD=-A(KI)                                                       MIN 0040
      JI=KI-K+J                                                         MIN 0041
      A(KI)=A(JI)                                                       MIN 0042
   30 A(JI) =HOLD                                                       MIN 0043
C        INTERCHANGE COLUMNS                                            MIN 0044
   35 I=M(K)                                                            MIN 0045
      IF(I-K) 45,45,38                                                  MIN 0046
   38 JP=N*(I-1)                                                        MIN 0047
      DO 40 J=1,N                                                       MIN 0048
      JK=NK+J                                                           MIN 0049
      JI=JP+J                                                           MIN 0050
      HOLD=-A(JK)                                                       MIN 0051
      A(JK)=A(JI)                                                       MIN 0052
```

```
      40 A(JI) =HOLD                                                MIN 0053
C        DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS    MIN 0054
C          CONTAINED IN BIGA)                                       MIN 0055
      45 IF(ABS(BIGA)-1.E-20)46,46,48                               MIN 0056
      46 D=0.0                                                      MIN 0057
         RETURN                                                     MIN 0058
      48 DO 55 I=1,N                                                MIN 0059
         IF(I-K) 50,55,50                                           MIN 0060
      50 IK=NK+I                                                    MIN 0061
         A(IK)=A(IK)/(-BIGA)                                        MIN 0062
      55 CONTINUE                                                   MIN 0063
C          REDUCE MATRIX                                            MIN 0064
         DO 65 I=1,N                                                MIN 0065
         IK=NK+I                                                    MIN 0066
         HOLD=A(IK)                                                 MIN 0067
         IJ=I-N                                                     MIN 0068
         DO 65 J=1,N                                                MIN 0069
         IJ=IJ+N                                                    MIN 0070
         IF(I-K) 60,65,60                                           MIN 0071
      60 IF(J-K) 62,65,62                                           MIN 0072
      62 KJ=IJ-I+K                                                  MIN 0073
         A(IJ)=HOLD*A(KJ)+A(IJ)                                     MIN 0074
      65 CONTINUE                                                   MIN 0075
C          DIVIDE ROW BY PIVOT                                      MIN 0076
         KJ=K-N                                                     MIN 0077
         DO 75 J=1,N                                                MIN 0078
         KJ=KJ+N                                                    MIN 0079
         IF(J-K) 70,75,70                                           MIN 0080
      70 A(KJ)=A(KJ)/BIGA                                           MIN 0081
      75 CONTINUE                                                   MIN 0082
C          PRODUCT OF PIVOTS                                        MIN 0083
         D=D*BIGA                                                   MIN 0084
C          REPLACE PIVOT BY RECIPROCAL                              MIN 0085
         A(KK)=1.0/BIGA                                             MIN 0086
      80 CONTINUE                                                   MIN 0087
C          FINAL ROW AND COLUMN INTERCHANGE                         MIN 0088
         K=N                                                        MIN 0089
     100 K=(K-1)                                                    MIN 0090
         IF(K) 150,150,105                                          MIN 0091
     105 I=L(K)                                                     MIN 0092
         IF(I-K) 120,120,108                                        MIN 0093
     108 JQ=N*(K-1)                                                 MIN 0094
         JR=N*(I-1)                                                 MIN 0095
         DO 110 J=1,N                                               MIN 0096
         JK=JQ+J                                                    MIN 0097
         HOLD=A(JK)                                                 MIN 0098
         JI=JR+J                                                    MIN 0099
         A(JK)=-A(JI)                                               MIN 0100
     110 A(JI) =HOLD                                                MIN 0101
     120 J=M(K)                                                     MIN 0102
         IF(J-K) 100,100,125                                        MIN 0103
     125 KI=K-N                                                     MIN 0104
```

```
      DO 130 I=1,N                                              MIN 0105
      KI=KI+N                                                   MIN 0106
      HOLD=A(KI)                                                MIN 0107
      JI=KI-K+J                                                 MIN 0108
      A(KI)=-A(JI)                                              MIN 0109
  130 A(JI) =HOLD                                               MIN 0110
      GO TO 100                                                 MIN 0111
  150 RETURN                                                    MIN 0112
      END                                                       MIN 0113
```

```
      SUBROUTINE RUNK2(KUTTA,N)
      REAL NINE
      DIMENSION C1(40),C2(40),C3(40),C4(40),C5(40),SX(40),E(40)
      COMMON TIME,DT,DTP,DTSAM,TRUN,TEND,PCDUM(14)
      COMMON XD(40)
      COMMON X(40)
      DATA HALF,THIRD,SIXTH,EIGTH/.5,.33333333,.16666667,.125/
      DATA TWO,THREE,FOUR,EIGHT,NINE,THIRTY/2.,3.,4.,8.,9.,30./
      GO TO (1,2,3,4,5),KUTTA
    1 DO 11 I=1,N
      SX(I)=X(I)
      C1(I)=DT*XD(I)
   11 X(I)=SX(I)+THIRD*C1(I)
      RETURN
    2 DO 12 I=1,N
      C2(I)=DT*XD(I)
   12 X(I)=SX(I)+SIXTH*(C1(I)+C2(I))
      RETURN
    3 DO 13 I=1,N
      C3(I)=DT*XD(I)
   13 X(I)=SX(I)+EIGTH*(C1(I)+THREE*C3(I))
      RETURN
    4 DO 14 I=1,N
      C4(I)=DT*XD(I)
   14 X(I)=SX(I)+HALF*(C1(I)-THREE*C3(I)+FOUR*C4(I))
      RETURN
    5 DO 15 I=1,N
      C5(I)=DT*XD(I)
      X(I)=SX(I)+SIXTH*(C1(I)+FOUR*C4(I)+C5(I))
   15 E(I)=(TWO*C1(I)-NINE*C3(I)+EIGHT*C4(I)-C5(I))/THIRTY*(DT/.25)
      CALL AUTO(X,E,KONTRL)
      IF(KONTRL-2)10,20,30
   20 DT=DT*HALF
      DO 21 I=1,N
   21 X(I)=SX(I)
      KUTTA=0
      RETURN
   10 TIME=TIME+DT
      RETURN
   30 TIME=TIME+DT
      DT=DT*TWO
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

REFERENCES

1. Shuford, D. W.: "Users Guide for Program REENTR; Space Shuttle External Tank Reentry Simulation Program." Northrop Services, Inc., Huntsville, Alabama, M-250-1303, August 1974.

BIBLIOGRAPHY

Crenshaw, J.: "QUAD2 Integration Routine." Northrop Services, Inc., Huntsville, Alabama, Informal Memorandum 7960-69-1, January 1969.

Grafton, E. A. and Javinen, W. A.: "Users Guide: SRB Reentry Program 'BDBI'." Northrop Services, Inc., Huntsville, Alabama, TN-250-1329, September 1974.

APPROVAL

USERS GUIDE FOR SKYLAB DYNAMICS PROGRAM, SKYDYN

by M. S. Hopkins

The information in this report has been reviewed for technical content. Review of any information concerning Department of Defense or nuclear energy activities or programs has been made by the MSFC Security Classification Officer. This report, in its entirety, has been determined to be unclassified.

G. D. Hopson
Director, System Dynamics Laboratory

Harold N. Scofield
H. N. Scofield
Control Systems Division

U.S. GOVERNMENT PRINTING OFFICE: 1980—640-247/586 REGION NO. 4