

9675

REPORT NO. GDC-DBB70-002
CONTRACT NAS9-11193

CR 114985

SPACE SHUTTLE SYNTHESIS PROGRAM (SSSP)

VOLUME I, PART 3 + PROGRAM OUTPUT
FINAL REPORT

GENERAL DYNAMICS
Convair Aerospace Division

FACILITY FORM 604
N71-24825
(ACCESSION NUMBER)
353
(PAGES)
CR-114985
(NASA CR OR TMX OR AD NUMBER)

(THRU)
C-3
(CODE)
31
(CATEGORY)

REPORT NO. GDC-DBB7C-002

SPACE SHUTTLE SYNTHESIS PROGRAM (SSSP)

VOLUME I, PART 3 + PROGRAM OUTPUT
FINAL REPORT

December 1970

Prepared Under
Contract NAS9-11193

Submitted to
National Aeronautics and Space Administration
MANNED SPACECRAFT CENTER
Houston, Texas

Prepared by
CONVAIR AEROSPACE DIVISION OF GENERAL DYNAMICS
San Diego, California

FOREWORD

Volume I, Part 3: Program Output

The SSSP documentation is presented in two volumes. Volume I contains the basic user's manual text and all of the simulation input and output description as well as a complete listing of the computer program FORTRAN V source deck. Volume II contains a compilation of statistical data on previous aircraft, missiles and space systems to serve as background information and program inputs to the weight/volume portion of the program.

This report is the third of three documents for Volume I. Part 1 contains the engineering and programming discussion and Part 2 contains the program operating instructions. This document also contains all of the Volume I appendices.

SUMMARY

The Space Shuttle Synthesis Program (SSSP) automates the trajectory, weights and performance computations essential to predesign of the Space Shuttle system for earth-to-orbit operations. The two-stage Space Shuttle system is a completely reusable space transportation system consisting of a booster and an orbiter element. The SSSP's major parts are a detailed weight/volume routine, a precision three-dimensional trajectory simulation, and the iteration and synthesis logic necessary to satisfy the hardware and trajectory constraints.

The SSSP is a highly useful tool in conceptual design studies where the effects of various trajectory configuration and shuttle subsystem parameters must be evaluated relatively rapidly and economically. The program furnishes sensitivity and tradeoff data for proper selection of configuration and trajectory predesign parameters. Emphasis is placed upon predesign simplicity and minimum input preparation. Characteristic equations for describing aerodynamic and propulsion models and for computing weights and volumes are kept relatively simple. The synthesis program is designed for a relatively large number of two-stage Space Shuttle configurations and mission types, but avoids the complexity of a completely generalized computer program that would be unwieldy to use and/or modify.

CONTENTS

FOREWORD	i
SUMMARY	ii
Table of Contents	iii
List of Figures	ix
List of Tables	xii
1 INTRODUCTION	1-1
2 ENGINEERING DISCUSSION	2-1
2.1 MISSION AND VEHICLE MODEL	2-1
2.1.1 THE SPACE SHUTTLE CONCEPT	2-1
2.1.2 TRAJECTORY PROFILE	2-3
2.1.3 VEHICLE CHARACTERISTICS AND CONSTRAINTS	2-5
2.2 SYNTHESIS METHODOLOGY	2-7
2.2.1 WEIGHTS AND GEOMETRY	2-9
2.2.2 TRAJECTORY COMPUTATION	2-11
2.2.2.1 Coordinate Systems	2-17
2.2.2.2 Coordinate System Transformations	2-23
2.2.2.3 Three Degree-of-Freedom Scalar Differential Equations of Motion	2-29
2.2.2.4 Supplementary Trajectory Parameter Differential Equations	2-39
2.2.2.5 Trajectory Solution Singularities	2-44
2.2.2.6 Specified Time Rate of Change of Relative Flight Path Angle	2-47
2.2.2.7 Solution by Numerical Integration	2-50
2.2.2.8 Trajectory Parameters	2-55
2.2.2.9 After-The-Fact Parameters	2-75
2.2.2.10 Optional After-The-Fact Parameters	2-81
2.3 SYNTHESIS TECHNIQUES	2-106
2.3.1 BASIC SYNTHESIS ITERATION	2-109
2.3.1.1 Weight Sizing Driver	2-112
2.3.1.2 Trajectory Simulation Driver (Ascent)	2-113
2.3.2 BASIC SYNTHESIS OPTIONS	2-119
2.3.2.1 Fixed Booster Thrust (Common Stage Engines)	2-119
2.3.2.2 Fixed Liftoff Thrust-to-Weight	2-121

2.3.2.3 Fixed Booster Gross Weight (Liftoff Weight)	2-124
2.3.2.4 Fixed Orbiter Gross Weight	2-126
2.3.2.5 Fixed Orbiter Propellant Weight	2-128
2.3.3 SECONDARY SYNTHESIS OPTIONS	2-130
2.3.3.1 Sequential Stage Burns	2-130
2.3.3.2 Simultaneous Stage Burns	2-131
2.3.3.3 Fixed Solid Rocket Motor Augmentation	2-133
2.3.4 BOOSTER SUBSONIC CRUISE OPTIONS	2-134
2.3.4.1 Booster Range Calculation	2-136
2.3.4.2 Booster Cruise Performance Calculation	2-154
3 PROGRAMMING DISCUSSION	3-1
3.1 PROGRAM ORGANIZATION	3-1
3.2 OVERLAYS AND SUBROUTINES	3-8
3.2.1 RESIDENT OVERLAY	3-8
3.2.1.1 MAIN Program	3-9
3.2.1.2 CHECK Subroutine	3-9
3.2.1.3 STORE Subroutine	3-9
3.2.2 SYNTHESIS OVERLAY	3-9
3.2.2.1 SYNTH Program	3-9
3.2.2.2 FRENCH Subroutine	3-13
3.2.2.3 READY Subroutine	3-13
3.2.2.4 SETO Subroutine	3-13
3.2.2.5 VEHDF Subroutine	3-13
3.2.2.6 TRAJA Subroutine	3-13
3.2.2.7 TRAJC Subroutine	3-20
3.2.2.8 ITER8 Subroutine	3-21
3.2.2.9 RANGE Subroutine	3-21
3.2.2.10 TBL2D Subroutine	3-24
3.2.2.11 SUMOUT Subroutine	3-24
3.2.3 GTSM OVERLAY	3-28
3.2.3.1 GTSM Program	3-29
3.2.3.2 CNTVR Subroutine	3-30
3.2.3.3 XNEW1 Subroutine	3-30
3.2.3.4 XNEW4 Subroutine	3-34
3.2.3.5 TRAJB Subroutine	3-36
3.2.3.6 ENDCN Subroutine	3-38
3.2.3.7 INTGRT Subroutine	3-38
3.2.3.8 PAROUT Subroutine	3-42

3.2.3.9	EQTNL Subroutine	3-42
3.2.3.10	ANGLE Subroutine	3-44
3.2.3.11	ATMOS Subroutine	3-46
3.2.3.12	AERO1, AERO2, AERO3 Subroutines	3-46
3.2.3.13	PRPSN Subroutine	3-49
3.2.3.14	TRAN1, TRAN2, TRAN3 Subroutines	3-51
3.2.3.15	QUAD Subroutine	3-51
3.2.3.16	QARG Subroutine	3-51
3.2.4	WTVOL OVERLAY	3-51
3.2.4.1	WTVOL Program	3-52
3.2.4.2	WTSCH Subroutine	3-52
3.2.4.3	SOLVE Subroutine	3-52
3.2.4.4	TAMPER Subroutine	3-52
3.2.4.5	PRINTW Subroutine	3-52
3.2.4.6	PRINTV Subroutine	3-52
3.2.4.7	PROTHR Subroutine	3-62
3.2.4.8	PRWTSM Subroutine	3-62
3.2.4.9	PRITEQ Subroutine	3-62
3.2.4.10	PRITVA Subroutine	3-62
4	PROGRAM OPERATION INSTRUCTIONS	4-1
4.1	DECK SETUP	4-2
4.1.1	CONTROL CARDS	4-2
4.1.2	IQS CARD	4-4
4.1.2.1	JQ and JB	4-4
4.1.2.2	MPRNTØ and MPRNTB	4-4
4.1.2.3	MPNCHØ and MPNCHB	4-5
4.1.3	TITLES CARD	4-5
4.1.4	DATA DECKS	4-5
4.2	INPUT PARAMETERS	4-7
4.2.1	\$DATA3	4-8
4.2.2	\$DATA2	4-19
4.2.3	\$DATA1	4-25
4.2.3.1	Initial Conditions	4-26
4.2.3.2	General Iteration	4-35
4.2.3.3	Targeting	4-41
4.2.3.4	Integration Control	4-46
4.2.3.5	Program Control	4-50
4.2.3.6	Modeling Control	4-54
4.2.3.7	Input Parameter Index	4-85

4.3	BASIC SYNTHESIS OPERATION	4-92
4.3.1	PROPULSION OPTION	4-92
4.3.1.1	Fixed Booster Thrust	4-92
4.3.1.2	Fixed Liftoff Thrust-to-Weight	4-93
4.3.2	STAGE BURN SEQUENCE	4-94
4.3.2.1	Sequential Stage Burns	4-94
4.3.2.2	Simultaneous Stage Burns	4-94
4.3.3	BOOSTER CRUISE RANGE CALCULATION	4-95
4.3.3.1	Parametric Flyback Range Data	4-95
4.3.3.2	Staging Q Function Range	4-96
4.3.3.3	Constant Range	4-96
4.3.3.4	Ballistic Impact Range	4-96
4.3.3.5	Entry Trajectory Simulation Range	4-97
4.3.4	BOOSTER CRUISE PERFORMANCE CALCULATION	4-97
4.3.4.1	Simplified Single Segment Cruise	4-97
4.3.4.2	Four Segment Reference Range Option 1	4-97
4.3.4.3	Four Segment Reference Range Option 2	4-98
4.3.5	STAGE WEIGHT CONSTRAINTS AND SOLID ROCKET THRUST AUGMENTATION	4-98
4.3.5.1	Fixed Booster Gross Weight (Liftoff Weight)	4-98
4.3.5.2	Fixed Orbiter Gross Weight	4-99
4.3.5.3	Fixed Orbiter Propellant Weight	4-99
4.3.5.4	Fixed Solid Rocket Motor Augmentation	4-100
5	PROGRAM OUTPUT	5-1
5.1	OUTPUT PARAMETERS	5-5
5.1.1	WEIGHTS AND GEOMETRY	5-5
5.1.1.1	Weight Breakdown	5-5
5.1.1.2	Volume Breakdown and Design Data Summary	5-9
5.1.1.3	Sizing Basis and Performance Weight Summary	5-9
5.1.1.4	Scaling Coefficients	5-13
5.1.1.5	Space Shuttle Synthesis Summary	5-13
5.1.2	TRAJECTORY SIMULATION	5-21
5.1.2.1	Regular Output Parameters	5-22
5.1.2.2	Optional Output Parameters	5-33
5.1.2.3	Output Parameter Index	5-40
5.2	ERROR DIAGNOSTICS	5-45
5.2.1	SYNTHESIS DRIVER	5-46
5.2.2	TRAJECTORY SIMULATION	5-46
5.2.3	WEIGHT/VOLUME SIZING	5-49

6	SAMPLE CASE DISCUSSION	6-1
6.1	VEHICLE SIZING DATA BASE	6-1
6.2	SYNTHESIS ITERATION DATA BASE	6-2
6.3	TRAJECTORY SIMULATION DATA BASE	6-2
7	REFERENCES	7-1

APPENDICES

I	ELEMENTAL COORDINATE TRANSFORMATIONS	I-1
II	GTSM SUBPROGRAM PRINCIPAL PARAMETERS AND SUBSCRIPTED ARRAYS	II-1
II-1	The V Array	(II-1)-2
II-2	The VQ Array	(II-2)-5
II-3	The Q Array	(II-3)-7
II-4	The Z Array	(II-4)-9
II-5	The W Array	(II-5)-12
II-6	The WQ Array	(II-6)-16
III	COMPILED IN ATMOSPHERE TABLE	III-1
IV	COMPUTER PROGRAM FORTRAN LISTING	
IV-1	Program MAIN	(IV-1)-3
IV-2	Subroutine CHECK	(IV-2)-10
IV-3	Subroutine STORE	(IV-3)-11
IV-4	Program SYNTH	(IV-4)-19
IV-5	Subroutine FRENCH	(IV-5)-21
IV-6	Subroutine READY	(IV-6)-25
IV-7	Subroutine SETO	(IV-7)-27
IV-8	Subroutine VEHDF	(IV-8)-29
IV-9	Subroutine TRAJA	(IV-9)-33
IV-10	Subroutine TRAJC	(IV-10)-47
IV-11	Subroutine ITER8	(IV-11)-51
IV-12	Subroutine RANGE	(IV-12)-53
IV-13	Subroutine TBL2D	(IV-13)-57
IV-14	Subroutine SUMOUT	(IV-14)-59
IV-15	Program GTSM	(IV-15)-67
IV-16	Subroutine CNTVR	(IV-16)-71
IV-17	Subroutine XNEW1	(IV-17)-73
IV-18	Subroutine XNEW4	(IV-18)-75

IV-19	Subroutine TRAJB	(IV-19)-79
IV-20	Subroutine ENDCN	(IV-20)-87
IV-21	Subroutine INTGRT	(IV-21)-89
IV-22	Subroutine PAROUT	(IV-22)-95
IV-23	Subroutine EQTNL	(IV-23)-97
IV-24	Subroutine ANGLE	(IV-24)-113
IV-25	Subroutine ATMOS	(IV-25)-119
IV-26	Subroutine AERO1	(IV-26)-120
IV-27	Subroutine AERO2	(IV-27)-121
IV-28	Subroutine AFLOW	(IV-28)-123
IV-29	Subroutine PRPSN	(IV-29)-125
IV-30	Subroutines TRAN1, TRAN2, TRAN3	(IV-30)-129
IV-31	Subroutine QUAD	(IV-31)-135
IV-32	Subroutine QARG	(IV-32)-137
IV-33	Program WTVOL	(IV-33)-139
IV-34	Subroutine WTSCH	(IV-34)-145
IV-35	Subroutine SOLVE	(IV-35)-163
IV-36	Subroutine TAMPER	(IV-36)-165
IV-37	Subroutine PRINTW	(IV-37)-171
IV-38	Subroutine PRINTV	(IV-38)-179
IV-39	Subroutine PROTHR	(IV-39)-181
IV-40	Subroutine PRWTSM	(IV-40)-185
IV-41	Subroutine PRITEQ	(IV-41)-189
IV-42	Subroutine PRITVA	(IV-42)-191
IV-43	BLOCK DATA	(IV-43)-195
V	SPECIFIED \$DATA1 PARAMETERS	V-1
V-1	Specified Value Parameters	V-2
V-2	Parameter Defined During Synthesis	V-4
VI	SYNTHESIS DRIVER PRINCIPAL SUBSCRIPTED ARRAYS	VI-1
VII	SAMPLE CASE LISTING	VII-1

LIST OF FIGURES

Figure No.		Page No.
2-1	Space Shuttle System	2-2
2-2	Synthesis Design Process	2-8
2-3	Weight/Volume Calculations	2-10
2-4	Weight/Sizing Input Development	2-12
2-5	The Primary Inertial Coordinate System	2-18
2-6	The Earth-Fixed Coordinate System	2-19
2-7	The Local Coordinate System	2-20
2-8	The Relative Velocity Coordinate System	2-21
2-9	The Vehicle Coordinate System	2-22
2-10	Axes Alignment Initially and During the Vehicle Axes to Relative Velocity Axes Transformation	2-24
2-11	Pitch Angle of Attack Defined from the Pitch Angle	2-61
2-12	Geometry for the Specified Roll Axis Mode	2-62
2-13	Pitch Angle Defined from the Pitch Angle of Attack	2-63
2-14	Central Body Surface Geometry	2-64
2-15	Inertial Velocity - Relative Velocity Vector Relationship	2-80
2-16	Orbital Inclination and Nodal Longitude	2-81
2-17	The $\hat{J}_1 - \hat{J}_2 - \hat{J}_3$ Inertial Coordinate System	2-94
2-18	Transformation from Earth Fixed Axis to $\hat{J}_1 - \hat{J}_2 - \hat{J}_3$ Inertial Axes	2-97
2-19	Geodetic Tip Angle	2-99
2-20	SSSP Organization	2-107
2-21	Linear Altitude Angles for a Booster Entry/Return Trajectory	2-146
3-1	MAIN Overlay - Subroutine Interrelationship	3-2
3-2	SYNTH Overlay - Subroutine Interrelationship	3-3
3-3	GTSM Overlay - Subroutine Interrelationship	3-4
3-4	WTVOL Overlay - Subroutine Interrelationship	3-5
3-5	MAIN Flowchart	3-10
3-6	STORE Flowchart	3-12
3-7	SYNTH Flowchart	3-14
3-8	FRENCH Flowchart	3-15
3-9	READY Flowchart	3-16
3-10	VEHDF Flowchart	3-17
3-11	TRAJA Flowchart	3-18
3-12	TRAJC Flowchart	3-22
3-13	ITER8 Flowchart	3-23
3-14	RANGE Flowchart	3-25
3-15	TBL2D Flowchart	3-26

List of Figures (continued)

Figure No.		Page No.
3-16	SUMOUT Flowchart	3-27
3-17	GTSM Flowchart	3-31
3-18	CNTVR Flowchart	3-32
3-19	XNEW1 Flowchart	3-33
3-20	XNEW4 Flowchart	3-35
3-21	TRAJB Flowchart	3-37
3-22	ENDCN Flowchart	3-39
3-23	INTGRT Flowchart	3-40
3-24	EQTNL Flowchart	3-43
3-25	ANGLE Flowchart	3-45
3-26	ATMOS Flowchart	3-47
3-27	AERO1, AERO2, AERO3 Flowchart	3-48
3-28	PRPSN Flowchart	3-50
3-29	WTVOL Flowchart	3-53
3-30	WTSCH Flowchart	3-58
3-31	TAMPER Flowchart	3-61
4-1	Data Deck Setup	4-3
4-2	Flight Profile with General Iteration Blocks (GIS Blocks)	4-35
4-3	Downrange, Crossrange, and Targeting	4-42
4-4	Targeting - Iteration Control Relationships	4-45
4-5	Atmosphere Table	4-56
4-6	Three-Dimensional Aerodynamic Tables	4-58
4-7	Relative Pitch Rate Table	4-66
4-8	Propulsion Tables	4-81
5-1	Abbreviated Synthesis Iteration Output	5-2
5-2	Sample Weight Breakdown	5-6
5-3	Sample Volume and Design Data Summary	5-10
5-4	Sample Sizing and Weight Summary - Orbiter	5-11
5-5	Sample Sizing and Weight Summary - Booster	5-12
5-6	Sample Output of Scaling Coefficients	5-14
5-7	Sample Output of Scaling for the Booster	5-17
5-8	Sample Space Shuttle Synthesis Summary Sheet	5-18
5-9	Sample Summary Data	5-20

List of Figures (continued)

Figure No.		Page No.
5-10	Standard $\hat{\xi} - \hat{\eta} - \hat{\zeta}$ Vehicle Coordinate System	5-24
5-11	Standard $\hat{i}_a - \hat{i}_v - \hat{i}_y$ Velocity Coordinate System	5-26
5-12	Vehicle Attitude Definition	5-28
5-13	The $\hat{e}_1 - \hat{e}_2 - \hat{e}_3$ Earth Fixed Coordinate System	5-37
5-14	The $\hat{j}_1 - \hat{j}_2 - \hat{j}_3$ Inertial Coordinate System	5-37

LIST OF TABLES

Table No.		Page No.
2-1	Summary of Error Conditions	2-54
2-2	State Variables	2-56
2-3	Nodal Longitude Definition	2-82
2-4	Latitude of Instantaneous Impact Point for Polar Orbits	2-89
2-5	Latitude of the Instantaneous Impact Point for Non-Polar Orbits	2-90
2-6	Longitude of the Instantaneous Impact Point for Polar Orbits and a Non-Rotating Central Body	2-92
2-7	Longitude of the Instantaneous Impact Point for Non-Polar Orbits and a Non-Rotating Central Body	2-92
2-8	Vector Component Designation	2-101
2-9	Parametric Range Increments from Staging to Flyback Initiation	2-138
2-10	Sample Booster Numerically Integrated Return Trajectory Input	2-152
4-1	Definition of Great Circle	4-43
4-1a	Input Table Printing Controlled by "YOUT"	4-32a
4-2	Variable Stepsize Procedure	4-47
4-3	Definition of Roll (Bank) Angle (σ)	4-62
4-4	Linear Cotangent Steering Options	4-64
4-5	Corresponding Atmosphere and Pitch Angle of Attack Table Arguments	4-67
4-6	Pitch Angle (ψ) and Pitch Angle of Attack (α) Definition	4-68
4-7	Yaw Angle of Attack (λ) Definition	4-74
4-8	SIMPO Option Table ("TWC(K)" = B.,)	4-78
4-9	Thrust-Time Table Options (TWC(K)" = $[10A + B]$.,)	4-80

5. PROGRAM OUTPUT

The basic printed output for a given SSSP case may be classified into the following four categories:

- 1) Input
- 2) Synthesis iterations
- 3) Converged case
- 4) Error diagnostics

Printout of the input NAMELIST data blocks occurs in the order in which they are read-in. Control for printing the \$DATA3 (orbiter and booster) and \$DATA2 input data blocks is handled with the IOS card discussed in Section 4.1.2. This data is printed in the NAMELIST format. Printout of the \$DATA1 input block is handled by the \$DATA2 input parameter PRNTX(1) and the \$DATA1 input parameter YOUT (see Section 4.2). PRNTX(1) allows the GTSM subprogram to print the initial conditions, simulation section data, etc., in a special format from the subroutine. TRAJA and YOUT controls the printout of the input tables such as the aerodynamics and atmosphere tables. Special care should be used in interpreting this output since many of the simulation section conditions and initial conditions for the trajectories are computed during the synthesis process and therefore will be printed with only their internally initialized values.

During the basic synthesis iterations the only printout that occurs is an abbreviated output summarizing the results of a particular iteration. This includes information concerning the results of sizing loop: gross stage weights, mass ratios, thrust levels and the results of the ascent trajectory simulation comparing the orbiter mass ratio used to size the orbiter in the WTVOL subprogram to that required to satisfy the terminal velocity requirement at parking orbit insertion (GTSM subprogram). This abbreviated output is illustrated in Figure 5-1 and shows the additional WTVOL iteration process performed on the system payload in order to

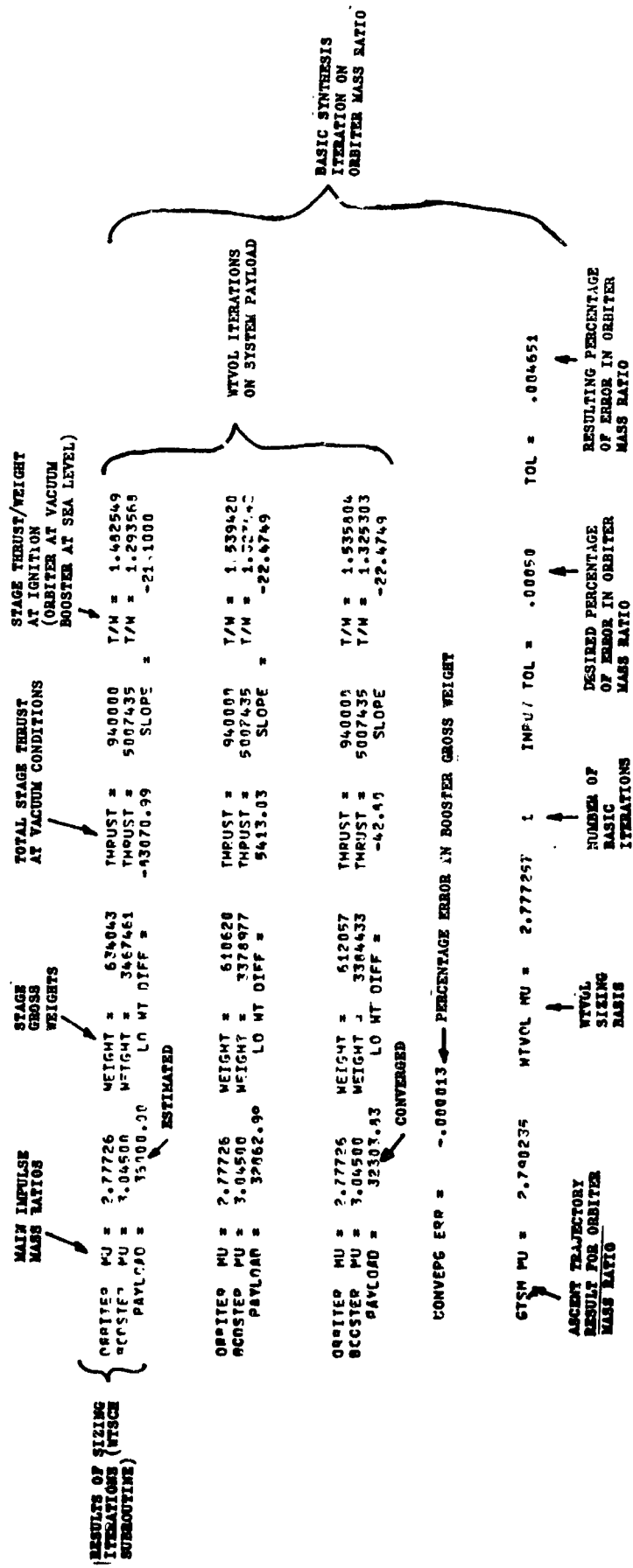


Figure 5-1. Abbreviated Synthesis Iteration Output

satisfy a booster gross weight constraint prior to testing the orbiter mass ratio in the GTSM subprogram. For this example the process will be continued since convergence to within the desired tolerance has not been met. These data are output from the subroutines WTVOL and ITER8 during the synthesis process. Detailed output during the various iterations may be obtained by setting certain input flags however, this technique should only be used in diagnosing problems with a particular configuration. If problems do occur during the iterations, the SSSP automatically prints a number of error diagnostics which usually suffice in tracking and solving the particular problem. This will be discussed in Section 5.2. For convenience the following list summarizes the various input flags* which allow detailed output during the iteration processes if desired:

\$DATA3 - (orbiter or booster):

NLISTO. If this parameter is input at a non-zero value, a NAMELIST[†] printout of the computed variables in the WTSCH subroutine is output for each pass through the routine for the stage. Since the iterations through WTSCH occurs many times during each sizing, this technique is very expensive in computer print time and should only be used if a problem is suspected in the basic setup of the \$DATA3 input list. This NAMELIST is automatically output if the solution is diverging. This is discussed in more detail in Section 5.2.

\$DATA2:

PRNTX(2). If this parameter is input at a non-zero value the trajectory flight history will be output for each pass through the GTSM subprogram during the basic synthesis iteration process. This technique should be reserved for solving initial problems with the ascent trajectory simulation if they occur*.

SLVOUT. An input value of 2., or 3., for this parameter allows print-out of the Newton-Raphson iteration parameters in the subroutine SOLVE which drive the WTSCH sizing process to convergence for each stage. If SLVOUT = 2.,

*Also see Section 4.2

[†]"MASS1" and "MASS2". See Volume II for a complete list of the computed variables.

these parameters are output only for the final iteration for each stage. If SLVOUT = 3., these parameters are output for each pass through WTSCH during the sizing iterations. Use of this input parameter should be limited to the tracking of problems associated with divergent solutions. If this occurs the basic \$DATA3 input should be rechecked prior to using this parameter.

WTOUT. If this parameter is input at 1.. a complete printout occurs of the final results of the WTVOL subprogram and GTSM subprogram during each basic synthesis iteration. Use of this parameter should be reserved for those cases when the basic process is diverging.

\$DATA1:

PRNT. See Section 4.2.3, \$DATA1. Use of this parameter is utilized to control the amount of output desired during the iteration process for converging the ascent trajectory simulation.

Printout of the converged case for the SSSP run occurs automatically and in the following sequence:

- 1) Orbiter stage data
- 2) Booster stage data
- 3) Ascent Trajectory*
- 4) Space Shuttle Synthesis summary

The parameters output for the trajectory simulation(s) are discussed in Section 5.1.2 below. The orbiter and booster stage data and the synthesis summary output is discussed in Section 5.1.1.

The error diagnostics are discussed in Section 5.2 along with comments concerning corrective measures to be taken.

* Followed by the booster entry trajectory if used.

5.1 OUTPUT PARAMETERS

5.1.1 WEIGHTS AND GEOMETRY. This section discusses the basic output data blocks for the WTVOL subprogram. This output consists of calculated data and print-back of pertinent input data for reference. The calculated data for each stage includes:

1. Weight Breakdown
2. Volume Breakdown and Design Data Summary
3. Sizing Basis and Performance Weight Summary

The input data that is printed back for reference purposes include the basic scaling coefficients utilized in sizing each stage.

For convenience this section also discusses the Space Shuttle Synthesis Summary output sheet though it is the last set of data printed for a given run and occurs after the printout of the trajectory simulation data.

5.1.1.1 Weight Breakdown. The weight breakdown format is a modified MIL-M-38310A STANDARD with the weights printed in both English and standard international units. This format is modified in the sense that certain pages of the MIL-STANDARD have been combined under a single title. An example of this is the personnel provisions and crew station controls and panels. This was done to reduce the number of printouts for small items that have little effect on the overall vehicle weight. The other variation is concerned with the breakdown of a specific page. An example of this is the aerodynamic surfaces. The MIL-STANDARD subdivides the surfaces into fixed surfaces, moving surfaces and fairings. However, a more usable form has been to subdivide into wing, horizontal, vertical and fairing. This form also lends itself to better correlation with historical data. It also requires less input data.

Figure 5-2 illustrates this weight breakdown technique provides the user with the internally computed acronym under which each item is output. The definitions of each acronym (in English units) are described in the Weight/Volume Handbook,

1	GMREF=3304390	MR=3.045	FLYACK=2	TITLES
WEIGHT BREAKDOWN - ORBITAL STAGE				
AERODYNAMIC SURFACES				
WING & WING MOUNTED CONTROL SURFACES				
HORIZONTAL SURFACES				
VERTICAL SURFACES				
FAIRINGS, SHROUDS AND ASSOCIATED STRUCTURE				
BODY STRUCTURE				
STRUCTURAL FUEL CONTAINERS				
STRUCTURAL OXIDIZER CONTAINERS				
BASIC BODY STRUCTURE				
SECONDARY STRUCTURE				
THRUST STRUCTURE				
INDUCED ENVIRONMENTAL PROTECTION				
COVER PANELS, NON-STRUCTURAL				
VEHICLE INSULATION				
LAUNCH, RECOVERY AND DOCKING				
LAUNCH GEAR				
DEPLOYABLE AERODYNAMIC DEVICES				
ALIGNING GEAR				
DOCKING STRUCTURE				
PROPULSION				
ENGINES AND ACCESSORIES				
SECONDARY ENGINES AND ACCESSORIES				
FUEL CONTAINERS AND SUPPORTS (NON-STRUCTURAL)				
OXIDIZER CONTAINERS AND SUPPORTS (NON-STRUCTURAL)				
SECONDARY FUEL TANKAGE AND SYSTEMS				
SECONDARY OXIDIZER TANKAGE AND SYSTEMS				
PROPELLANT INSULATION				
FUEL SYSTEM - MAIN				
OXIDIZER SYSTEM - MAIN				
PURGE SYSTEMS				
AIRBREATHING ENGINES AND INSTALLATION				
NACELLES, PODS, PYLONS, SUPPORTS				
AIRBREATHING PROPULSION TANKAGE AND SYSTEMS				
ORIENTATION, SEPARATION AND ULLAGE CONTROL				
SEPARATION SYSTEMS				
STABILITY AND CONTROL, ENGINE GIMBAL				
AERODYNAMIC CONTROLS				
SPATIAL ATTITUDE CONTROL SYSTEM				
CONTROL PROPELLANT TANKAGE AND SYSTEMS				
		POUNDS		KILOGRAMS
		22057 WISURF		10005 KWSURF
		13271 WINDG		6020 KWWING
		6512 WINDRZ		2954 KWHDRZ
		2274 WVERT		1031 KHWERT
		0 WFAIR		0 KWFAR
		53A05 WBOOY		24A05 KWBBOY
		8611 WINFUT		3906 KWIIFU
		2591 WINDRK		1175 KWIIDRK
		36270 WIBASIC		16452 KWBASIT
		2572 WSECS		1167 KWISECS
		3760 WTHRST		1706 KWTNRS
		10775 WCOVER		4887 KWCORE
		0 WINSUL		0 KWINSU
		7289 WLRD		3306 KWLARD
		0 WLAUNCH		0 KWLANC
		0 WIDPLOY		0 KWDPLD
		7289 WLG		3306 KWLGL
		0 WIDOCK		0 KWDOCK
		25659 WPRDP		11639 KWPDRDP
		9094 WENG5		4125 KWEENG5
		0 WENG52		0 KWEENG2
		0 WIFUTK		0 KWFUTK
		0 WIDRTR		0 KWDLTR
		964 WIFUTK2		437 KFUTK2
		476 WIDRTR2		216 KDLTR2
		2043 WINDSTK		927 KWINST
		1800 WIFUSYS		853 KWFUSY
		3290 WIDRSTK		1492 KWDKSY
		735 WIPR3YS		333 KWP3SY
		6022 WIDRPE		3094 KWDPRP
		0 WIDRCEL		0 KWDIACE
		355 WIDRFTK		161 KWDABFT
		6114 WDRSUL		2773 KWDORSU
		1175 WIAUKT		533 KWAUKT
		300 WISTAB		136 KWIATAB
		1730 WIAERO		785 KWAERO
		2377 WIAACS		1078 KWAACS
		532 WIDGSTK		241 KWDACST

Figure 5-2. Sample Weight Breakdown

PRIME POWER SOURCE AND DISTRIBUTION	2518	2518	2518	2518	2518
POWER SOURCE UNITS	198	198	198	198	198
POWER SOURCE TANKAGE AND SYSTEMS	1211	1211	1211	1211	1211
POWER CONVERSION AND DISTRIBUTION	1211	1211	1211	1211	1211
HYDRAULIC/PNEUMATIC	1211	1211	1211	1211	1211
GUIDANCE AND NAVIGATION	590	590	590	590	590
INSTRUMENTATION	1300	1300	1300	1300	1300
COMMUNICATION	600	600	600	600	600

Figure 5-2. (continued)

PERSONNEL PROVISIONS	2230 WPPROV	1011 KWPPEB
DRY STRUCTURE	(134249) WIDRY	(60094) KWDRY
DESIGN RESERVE (CONTINGENCY)	13425 WCDUT	6009 KWCOUT
PERSONNEL	500 WPFERS	245 KWFERS
PAYLOAD	20229 WPAVL	12804 KWPAYL
CARGO OR UPPER STAGE	0 WIPASS	0 KWPASS
MISSION EQUIPMENT/PAYLOAD	20229 WCARGO	12804 KWCARG
RESIDUAL PROPELLANT AND SERVICE ITEMS	2138 WIRESD	970 KWRRESI
TANK PRESSURIZATION AND PURGE GASES	0 WIGASP	0 KWGASP
TRAPPED FUEL	476 WIFUTR	216 KWFUTR
TRAPPED OXIDIZER	1262 WOLTRP	572 KWOLTR
SERVICE ITEMS RESIDUALS	400 WSETRP	101 KWSRTR
RESERVE PROPELLANT AND SERVICE ITEMS	3242 WRESRV	1471 KWRRESR
FUEL-MATH PROPU' SION	463 WIFURES	210 KWFURE
OXIDIZER-MAIN PROPULSION	2779 WOKRES	1261 KWOXRE
POWER SOURCE PROPELLANTS	0 WPOWRS	0 KWPOWR
LUBRICANTS	0 WOLIRS	0 KWOILR
ATTITUDE CONTROL PROPELLANTS	0 WAGRES	0 KWARRE
OPERATING WEIGHT EMPTY	(101023) KWNET	(82473) KWNWET
IN-FLIGHT LOSSES	5464 WLOSS	2470 KWL0SS
ICE AND FROST	0 WFR0ST	0 FR0ST
FUEL VENTED	0 WFUL0S	0 KWFUL0
OXIDIZER VENTED	0 WOKLOS	0 KWOXL0
POWER SOURCE PROPELLANTS	1500 WPOWFG	600 KWP0WF
ATTITUDE CONTROL PROPELLANTS	886 WAGSF0	402 KWARSE
LUBRICANTS AND OTHER SERVICE ITEMS	1300 W0IL	590 KWOIL
FLYBACK FUEL	1777 WABFU	806 KWA0FU
THRUST DECAY PROPELLANTS	1349 WIDECAY	612 KWD0CA
PROPELLANTS	399112 WIFUL0X	181034 KWFUL0X
FUEL - MAIN	53820 WIFUL	24412 KWFUL
OXIDIZER - MAIN	322921 WOKID	146475 KWOXID
FUEL - SECONDARY	3728 WIFUL2(1)	1691 KWFUL2
OXIDIZER - SECONDARY	18642 WOK2(1)	8456 KWOX2
MASS AT FULL THRUST	587747 WTD	266598 KWT0
PRE-IGNITION LOSSES	0 WPREIG	0 KWPRIE
MAXIMUM GROSS WEIGHT	(587747) WGR0SS	(266598) KNGR0S

Figure 5-2. (continued)

Volume II of this report. This breakdown is output from the subroutine PRINTW and under identical acronyms for each stage. (Figure 5-1 is a sample for the orbiter.) The conversion from pounds to kilograms is also performed in PRINTW.

5.1.1.2 Volume Breakdown and Design Data Summary. The volume summary prints each value computed and zero for those sub-volumes which are omitted. The design data includes the computed wetted areas, planform areas, unit weight of the aerodynamic surfaces, and wing and body dimensional data. Figure 5-3 illustrates this output data block and provides the user with the internally computed acronym under which each item is output. The definitions of each acronym (in English units) are described in the Weight/Volume Handbook, Volume II of this report. These parameters are output under identical acronyms for each stage. The volume data is output from the subroutine PRINTV and the design data from the subroutine PROTHR. The conversion from English units to international units is performed in each respective subroutine.

5.1.1.3 Sizing Basis and Performance Weight Summary. This output data block provides the current bases for the sizing process within the WTSCH subroutine and a summary of the performance weights utilized by the GTSM subprogram for each stage. The bases for the sizing process includes the mass ratio, propellant mixture ratio, effective specific impulse, and characteristic velocity increment utilized for each flight phase* as described in the Weight/Volume Handbook, Volume II of this report. The performance weight summary shows the total weight and expended weights associated with each phase of flight as also described in Volume II. The performance weight summary is the only output format for the stages that varies significantly between the orbiter and booster due to the differences in the phases of flight following separation. Figures 5-4 and 5-5 illustrate these differences, however the internally computed value for each item is output under the identical acronym for each stage. These figures include the sizing bases as a sample for each stage and the acronyms under which each item is output. This data is output from the

*These flight phases should not be confused with the ascent flight phases discussed in Section 2.3.1.2, Trajectory Simulation Driver.

1 GPER=3384390 MR=3.045 FLYRCK=2 TITLES		CU FT	CU M
VOLUMES			
BODY STRUCTURE		VSTRUC	5316.49
CARGO AND PASSENGER COMPARTMENTS		VCREN	1500.00
CARGO COMPARTMENT		VCRG	10600.00
LANDING GEAR BAYS		VLSBA	0.00
PROPULSION BAY WITHIN BODY		VPRPB	0.00
FUEL CONTAINERS		VFUTK	13517.59
Oxidizer Containers		VOKTK	4852.45
PROPELLANT INSULATION		VINST	0.00
SECONDARY FUEL CONTAINERS		VFKTL	1285.67
SECONDARY OXIDIZER CONTAINERS		VOKTL	380.45
OTHER BODY VOLUME		VOTHE	31638.59
TOTAL ENTRY VEHICLE VOLUME		VBDY	69091.2
			1935.3 K/BODY
DESIGN DATA			
WETTED AREAS		SQ. FT.	SQ.M.
GROSS BODY (SBODY)		SBODY	11814.49
FUEL TANKS (SFUTK)		SFUTK	4045.69
OXIDIZER TANKS (SOKTK)		SOKTK	0.00
			1897.57 KSBODY
			379.56 SFUTK
			0.00 SOKTK
PLAN AREAS		SQ. FT.	SQ.M.
THEORETICAL WING AREA (SWING)		SWING	168.89
EXPOSED WING AREA (SXPDS)		SXPDS	122.61
PLANFORM, INCL. ELEVONS (SPLAN)		SPLAN	268.09
VERTICAL SURFACES (SVERT)		SVERT	40.93
HORIZONTAL SURFACES (SHORZ)		SHORZ	82.45
			168.89 KSWING
			122.61 KXPDS
			268.09 KSPLAN
			40.93 KSVERT
			82.45 KSHORZ
UNIT WEIGHTS		LB/SQ.FT.	KG/SQ.M.
WING		UNWING	7.30
VERTICAL SURFACES		UNWVERT	5.16
HORIZONTAL SURFACES		UNWHORZ	7.34
WING LOADING		WLOADERS	98.75
			35.64 KUWWIN
			25.19 KUWVERT
			35.82 KUWHORZ
			482.11 KWOVER
DIMENSIONAL DATA		FEET	METERS
WING			
STRUCTURAL SPAN		CSPAN	112.97
ROOT CHORD LENGTH		CRDRT	23.15
THICKNESS RATIO		TDRER	.14
BODY			
LENGTH (LBODY)		LBODY	145.64
WIDTH (WBODY)		WBODY	22.93
HEIGHT (HBODY)		HBODY	29.35
			44.79 KLBODY
			6.99 KWBODY
			8.95 KHBODY

Figure 5-3. Sample Volume and Design Data Summary.

I	MR(I)	O/F RATIO	ISP	THEORETICAL VELOCITY INCREMENT	
				DELTA V (I) 0.0	DELTA V (II) 0.0
1=	1.0000	0.0000	0.0	0.0	0.0
2=	1.0000	0.0000	0.0	0.0	0.0
3=	2.7855	6.0000	451.6	14884.5	4536.6
4=	1.0156	6.0000	431.6	225.0	68.6
5=	1.1186	5.0000	416.0	1500.0	457.2
6=	1.0000	0.0000	0.0	0.0	0.0

WEIGHT SUMMARY

WEIGHT SUMMARY	POUNDS	KILOGRAMS
MANIKUM GROSS WEIGHT (GROSS)	WGRDSS	587747
PRE-IGNITION LOSSES	WPREIS	0
MASS AT IGNITION	WNAIT(1)	587747
FUEL	WFUEL(1)	0
OXIDIZER	WOK(1)	0
JETTISON	WJET(1)	0
END OF PRE-FLIGHT PHASE (WTO)	WNAIT(2)	587746
FUEL	WFUEL(2)	-53820
OXIDIZER	WOK(2)	-322421
JETTISON	WJET(2)	0
WEIGHT IN ORBIT (WOBIT)	WNAIT(4)	211005
JETTISON	WJET(4)	-6729
RETURN COMDITION (WETURN)	WNAIT(5)	204276
FUEL	WFUEL(5)	-3728
OXIDIZER	WOK(5)	-18642
JETTISON	WJET(5)	-2386
ENTRY WEIGHT (WENTRY)	WNAIT(6)	179319
FLYBACK FUEL	WABFU	-1777
LANDING WEIGHT (WLAND)	WNAIT	177742

Figure 5-4. Sample Sizing and Weight Summary - Orbiter

1 GREQ-3384390 MR-3.045 FLYBACK-2				THEORETICAL VELOCITY INCREMENT FEET/SECOND METERS/SECOND	
		O/F RATIO	ISP		
MASS RATIO DURING FLIGHT PHASE 1=	1.0000	0.0000	0.0	0.0	0.0
MASS RATIO DURING FLIGHT PHASE 2=	1.0000	0.0000	0.0	0.0	0.0
MASS RATIO DURING FLIGHT PHASE 3=	3.0450	6.0000	428.7	15359.3	4681.3
MASS RATIO DURING FLIGHT PHASE 4=	1.0000	6.0000	0.0	0.0	0.0
MASS RATIO DURING FLIGHT PHASE 5=	1.0000	5.0000	0.0	0.0	0.0
MASS RATIO DURING FLIGHT PHASE 6=	1.0000	0.0000	0.0	0.0	0.0
WEIGHT SUMMARY		POUNDS KILOGRAMS			
MAXIMUM GROSS WEIGHT (NGROSS)	WGRDSS	3384486	1535177	KWGRDS	
PRE-IGNITION LOSSES	WPREIG	0	0	KWPRIG	
MASS AT IGNITION	WNAIT(1)	3384486	1535177	KWNAIT(1)	
FUEL	WFUEL(1)	0	0	KWFUEL(1)	
OXIDIZER	WOK(1)	0	0	KWOK(1)	
JETTISON	WJET(1)	0	0	KWJET(1)	
END OF PRE-FLIGHT PHASE (WTO)	WNAIT(2)	3384487	1535176	KWNAIT(2)	
FUEL	WFUEL(3)	-324714	-147208	KWFUEL(3)	
OXIDIZER	WOK(3)	-1948283	-883726	KWOK(3)	
PAYLOAD ADDED/REMOVED	WJET(3)	-587747	-266598	KWJET(3)	
INITIAL SEPARATION	WNAIT(4)	523743	237566	KWNAIT(4)	
THRUST DECAY AND RESIDUALS	WJET(4)	-19121	-8673	KWJET(4)	
RETURN CONDITION (WETURN)	WNAIT(5)	504622	228892	KWNAIT(5)	
JETTISON AND EXPENDABLES	WJET(5)	-1730	-785	KWJET(5)	
INITIAL FLYBACK	WNAIT(6)	592891	229108	KWNAIT(6)	
FLYBACK FUEL	WABFU	-40947	-18573	KWABFU	
LANDING WEIGHT (WLAND)	WNAIT	461944	209534	KWNAIT2	

Figure 5-5. Sample Sizing and Weight Summary - Booster.

subroutine PRWTSM. The conversion from English units to international units is also performed in PRWTSM.

5.1.1.4 Scaling Coefficients. This block of output data is primarily printback of the stage \$DATA3 input and provides the user with easy reference to only the actual weight and geometry coefficients utilized in sizing the stages. This data includes the main and secondary vacuum thrust levels computed for scaling purposes, the jettison weight summaries, number of main engines, and propellant densities. Figure 5-6 illustrates this output data block which is printed from subroutines PRITEQ and PRITVA, respectively. This data is printed similarly for the booster however, at the end, an extra set of data is output which describes the current basis of the booster return cruise requirements and the flight conditions occurring at the point of maximum dynamic pressure during the ascent trajectory. This additional data is illustrated in Figure 5-7 and is output from the subroutine PRITVA during the print cycle for the booster.

5.1.1.5 Space Shuttle Synthesis Summary. This output data block is printed following the output for the trajectory simulation(s) and is used to briefly summarize the results of all of the basic output data blocks for convenience to the user. This data summarizes the individual stage, weights, mission propellant requirements, volume, geometry, and propulsion characteristics. Pertinent trajectory summary results include the flight conditions at staging and parking orbit insertion. Figure 5-8 illustrates the output for a sample summary sheet and includes the internal acronyms under which the items are printed from the subroutine SUMOUT. These acronyms are basically the internally stored values of each of the stages computed values for these terms. The propulsion characteristics are those utilized to simulate the ascent trajectory for each stage and do not necessarily agree with those used in sizing the stages (see Section 2.3.1.2, Trajectory Simulation Driver). The number of copies printed for this summary sheet is controlled by the \$DATA2 input parameter COPIES. Use of this summary is helpful to the user as a brief log of a particular

WEIGHT COEFFICIENTS = C(I),

1	1624.00000
4	1.19700
6	.00550
10	.61700
12	.54400
13	3.07000
23	.15000
29	.10000
30	.00916
33	4500.00000
43	.50000
45	.00200
48	.00350
51	.04000
52	.04000
55	.07710
62	47.62700
63	.47300
64	600.00000
65	.32600
66	.84900
68	950.00000
70	1300.00000
72	600.00000
74	.00146
75	760.00000
76	350.00000
96	.10000
97	270.00000
103	20229.05278
114	400.00000
127	1500.00000
129	47000.00000
131	1300.00000
139	1.24600
146	.53400
153	.00200
155	.17300
156	265.00000
160	.75000
161	300.00000
164	.60000
166	.00143
168	.00400
169	800.00000
170	.75000
171	1.25000

Figure 5-6. Sample Output of Scaling Coefficients.

174 .00420
 176 1.00000
 181 .91200
 182 1.12400
 183 .50010
 185 1.00000
 210 .03800
 212 .20003
 214 .01000
 220 100.00000
 221 .52000
 225 .00110
 226 .00006
 227 .00050
 228 .00114

VOLUME COEFFICIENTS = $K(I)$, $I = 1, 30$ =

1	.05000	7	.05000	2	0.00000	8	0.00000	3	0.00000	4	0.00000	5	0.00000
6	1500.00000	12	0.00000	13	0.00000	14	0.00000	15	10600.00000	16	0.00000	17	.45000
11	0.00000	17	0.00000	18	.59500	19	0.00000	20	0.00000	21	0.00000	22	0.00000
16	0.00000	22	0.00000	23	0.05000	24	0.00000	25	0.00000	26	0.00000	27	0.00000
21	0.00000	27	0.00000	28	403.95877	29	0.00000	30	0.00000				

1 GWR00=3384.390 MR=3.045 FLYACK=2 TITLES

CB000V = WIDTH OF BODY / VBODY ** .3333 = .5588
 CM000V = HEIGHT OF BODY / VBODY ** .3333 = .7154
 CL000V = LENGTH OF BODY / VBODY ** .3333 = 3.5493
 CS000V = WETTED AREA OF BODY / VBODY ** .6667 = 7.0167
 CSFAIR = FAIRING AREA / BODY WETTED AREA = 0.0000
 CSFUTK = WETTED AREA OF FUEL TANKS / TANK VOLUME ** .6667 = 7.2000
 CSF0RZ = STABILIZER AREA/TWO WING AREA = .4882
 CS0X7K = WETTED AREA OF OXIDIZER TANKS / TANK VOLUME ** .6667 = 0.0000
 CSPLAN = VEHICLE PLANFORM AREA / BODY VOLUME ** .6667 = 1.7139
 CSVERT = AREA OF EACH VERTICAL SURFACE / VBODY ** .6667 = .2617
 CTPRTY = ENGINE THRUST / WTD = 0.0000

```

C1W512 = AIRBREATHING ENGINE THRUST / WENTRY = 0.0000

NCREW = NUMBER OF CREW = 2
NWENG = NUMBER OF ENGINES = 2
RPOFU = FUEL DENSITY = 4.40
RPOFU2 = SECONDARY PROPUSSION FUEL DENSITY = 2.90
RNOX = OXIDIZER DENSITY = 71.00
RNOX2 = SECONDARY PROPUSSION OXIDIZER DENSITY 49.00
T101 = MAIN PROPUSSION THRUST = CTHRST * WORBIT + C(129) * WENG5 +
      0.000 * 587746 + 479000 * 2 = 940000
      CTHRST MVAIT(5) C(129) NEWG5 T101
T1012 = AIRBREATHING ENGINE THRUST = C1W512 * T101 *
      0.000 + 179519 + 0 = 0
      C1W512 MVAIT(6) C(158) T1012

NJET(1) = 0.0
NJET(2) = 0.0
NJET(3) = 0.0
NJET(4) = 6729.1
NJET(5) = 2366.2
NJET(6) = 0.0

```

Figure 5-6. (continued)

CTHS12 = AIRBREATHING ENGINE THRUST / WENTRY = 0.0000

NCREW = NUMRER OF CREW = ?
NENG5 = NUMBER OF ENGINES = 11
RPOFU = FUEL DENSITY = 4.40
RMOFU2 = SECONDARY PROPUSSION FUEL DENSITY = 0.00
PVOX = OXIDIZER DENSITY = 71.00
PVOX2 = SECONDARY PROPUSSION OXIDIZER DENSITY 0.00
TTCY = MAIN PROPUSSION TPURST = CTHRST * MORBIT + C(129) * NENG5 =
0.000 + 33844.87 + 455221 + 11 = 5007435

TYOT2 = AIRBREATHING ENGINE THRUST = CTAS12 X WETURN C(158) = *
0.000 + 502091 + 0 = 0

WJET(1) = 0.0
WJET(2) = 0.0
WJET(3) = 567747.1
WJET(4) = 19121.3
WJET(5) = 1730.3
WJET(6) = 0.0

SPECIFIC FUEL CONSUMPTION = .30000
CRUISE BACK VELOCITY = 290.00 FT/SEC
TPURST RATIO = .96856
L/N = 0.50000
FLYBACK RANGE = 413.17 N. MI.
RANK ANGLE DURING REENTRY = 0.00 DEG

MINIMUM ASCENT DYNAMIC PRESSURE
TIME AT MAX Q = 499.41 PSF
ALTITUDE AT MAX Q = 65.000 SEC
MACH NUMBER AT MAX Q = 29353 FT
RELATIVE VELOCITY AT MAX Q = 1.019
RELATIVE FLIGHT PATH ANGLE AT MAX Q = 1048.68 FPS
RELATIVE FLIGHT PATH ANGLE AT MAX Q = 68.25 DEG

SW(12)
SW(14)
SW(6)
SW(11)
SW(15)
SQ(10,2)
SE(9)
SQ(5,1)
SQ(6,1)
SQ(9,1)
SQ(7,1)
SQ(8,1)

Figure 5-7. Sample Output of Scaling for the Booster.

[illegible]

Figure 5-8. Sample Space Shuttle Synthesis Summary Sheet.

series of SSSP runs for a given configuration. This summary is only printed after a run has successfully converged. After the specified number of these summaries are output, one final data block is also printed which includes pertinent data concerning the ascent velocity losses, ascent pitch rates used in the final converged trajectory, and notes concerning the method of and input constants for calculating the booster cruise range and fuel requirements. This final data block is also helpful to the user in tracking his design philosophy for a given configuration. Figure 5-9 illustrates this output which is printed only once and also from the subroutine SUMOUT.

1 CWRQ=3364390 MR=3.045 FLYRCK=2 ~~TITLES~~

SPACE SHUTTLE SYNTHESIS SUMMARY

GRAV LOSS AT STAGING = 679.62 SQ (36,1)

GRAVITY LOSS AT STAGING = 4286.34 SQ (36,2)

GRAV LOSS AT INJECTION = 679.62 SQ (36,3)

GRAVITY LOSS AT INJECTION = 4396.96 SQ (36,4)

MISALIGNMENT LOSS AT INJECTION = 86.29 SQ (36,5)

PITX = 1.7550270E+00 Q (62,1)

SALP(5) = -1.4260032E+01 Q (45,5)

PSLOT(5) = 9.0554000E+01 Q (35,5)

BOOSTER BURN TIME (SEC. 2) = 222.90 SQ (37,1)

ORBITER BURN TIME (SEC. 5,6,7) = 191.36 SQ (37,2)

FLYRACK

F(Q STAGE), Q AT STAGING = 2.000 SW(12)

CRUISE, RANGE = 413.17 SW(15)

SFC = .3000 SW(12)

L/D = 0.5000 SW(11)

CRUISE VELOCITY = 298.00 SW(14)

DRNG = 0.00 SQ(10,3)

WFLYX = 0.00 SQ(32,4)

CRUISE FUEL = 48947.24 WFLYX

FLYRACK FUEL = 48947.24 KLABFUB

C(1214) = .00064 SCB(214)

Figure 5-9. Sample Summary Data.

5.1.2 TRAJECTORY SIMULATION. The output parameters which are available in GTSM can be classified by thirteen basic groups. These are:

1. Time	8. Weight
2. Position	9. Propulsion
3. Velocity	10. Velocity
4. Acceleration	11. Orbital Elements
5. Attitude	12. Optional VECTOR Components
6. Environment	13. Position in Radar Coordinates
7. Aerodynamics	

The parameters which comprise groups 1 through 10 are the "regular" output parameters and are computed at least once during each integration step. Groups 11, 12 and 13 are optional output parameters and must be called out by input if required. If called, the optional output parameters are computed once during each integration step. Any parameter* which is computed once during each integration step can be used as either an iteration end condition or a simulation section termination parameter by setting the iteration end condition code ("ECC1(M)", "ECC2(M)", or "ECC3(M)") or the simulation section code ("STGC(K)"), as appropriate, to the value of the output parameter code.

This Output Parameters section lists all the possible output parameters with their definitions according to their basic group; within each basic group the output parameters are listed according to their output parameter code number. At the end of each definition, three quantities appear in parentheses; these are:

- a. The printed output title
- b. The parameter units
- c. The corresponding internal parameter.

* Providing selection of a parameter does not conflict with the parameters which are defined as part of the synthesis process.

5.1.2.1 Regular Output Parameters. These parameters are printed at either each integration step or at integral numbers of integration steps according to the value of input parameter "XOUT(K)" - see Section 4.2.3.5 Program Control.

5.1.1 TIME.

- 1. Absolute time. This is the time from the initiation of simulation Section 1 + "T0" (input parameter - see Section 4.2.3.1 Vehicle State) to the current time
(ABS TIME) (SEC) (T)
- 0. Section relative time. This time is the time from the initiation of the current simulation section to the current time.
(REL TIME) (SEC) (TR)

POSITION.

- 1. Geocentric Radius Magnitude
(RADIUS) (FT) (Z(1))
- 5. Geocentric Latitude - Positive in the northern hemisphere
(LATITUDE) (DEG) (Z(5))
- 6. Geocentric Longitude - Positive to the east of the Greenwich meridian
(LONGITUDE) (DEG) (Z(6))
- 8. Altitude above the surface of the oblate central body
(ALTITUDE) (FT) (Z(8))
- 49. Down range angle (central angle) from the initial geocentric radius vector which is defined by the initial latitude ("LAT") and the initial longitude ("LNG"). See Figure 4-2 in Section 4.2.3.3 Targeting.
(DOWN RANGE) (DEG) (Z(49))
- 50. Cross range angle. The cross range angle is the angle between two planes, S_1 and S_2 , where:

S_1 is the plane which contains the initial geocentric radius vector defined by "LAT" and "LNG", and a reference geocentric vector defined by "TLAT(M)" and "TLNG(M)".

S_2 is the plane which contains the initial geocentric radius vector defined by "LAT" and "LNG", and the current geocentric position radius vector.

This parameter is only computed if the targeting parameter "CRGF(M)" has a value greater than zero. See the discussion and Figure 4-3 in Section 4.2.3.3 Targeting

(CROSS RANGE)	(DEG)	(Z(50))

51. Target miss angle. This angle is the central angle between the current position geocentric radius vector and the reference geocentric vector defined by "TLAT(M)" and "TLNG(M)". This parameter is only computed if the targeting parameter "CRGF(M)" has a value greater than zero. See Section 4.2.3.3 Targeting.

(TARGET DIST)	(DEG)	(Z(51))

VELOCITY.

2. Earth Relative Velocity Magnitude

(REL VELOCITY)	(FT/SEC)	(Z(2))

3. Earth Relative Flight Path Angle.

(REL GAMMA)	(DEG)	(Z(3))

4. Earth Relative Azimuth Measured Clockwise from North.

(REL AZIMUTH)	(DEG)	(Z(4))

41. Inertial Velocity Magnitude.

(INRTL VELOCITY)	(FT/SEC)	(Z(41))

42. Inertial Flight Path Angle.

(INRTL GAMMA)	(DEG)	(Z(42))

65. Inertial Azimuth Measured Clockwise from North.

(INRTL AZIMUTH)	(DEG)	(Z(65))

ACCELERATION.

24. The aerodynamic yaw force which is the aerodynamic force directed along the standard pitch axis ($\hat{\eta}$ axis). The algebraic sign is in accordance with the standard* $\hat{\xi} - \hat{\eta} - \hat{\zeta}$ coordinate system (see Figure 5-10) (positive to the right)

(YAW F(AERO))

(LB)

(Z(24))

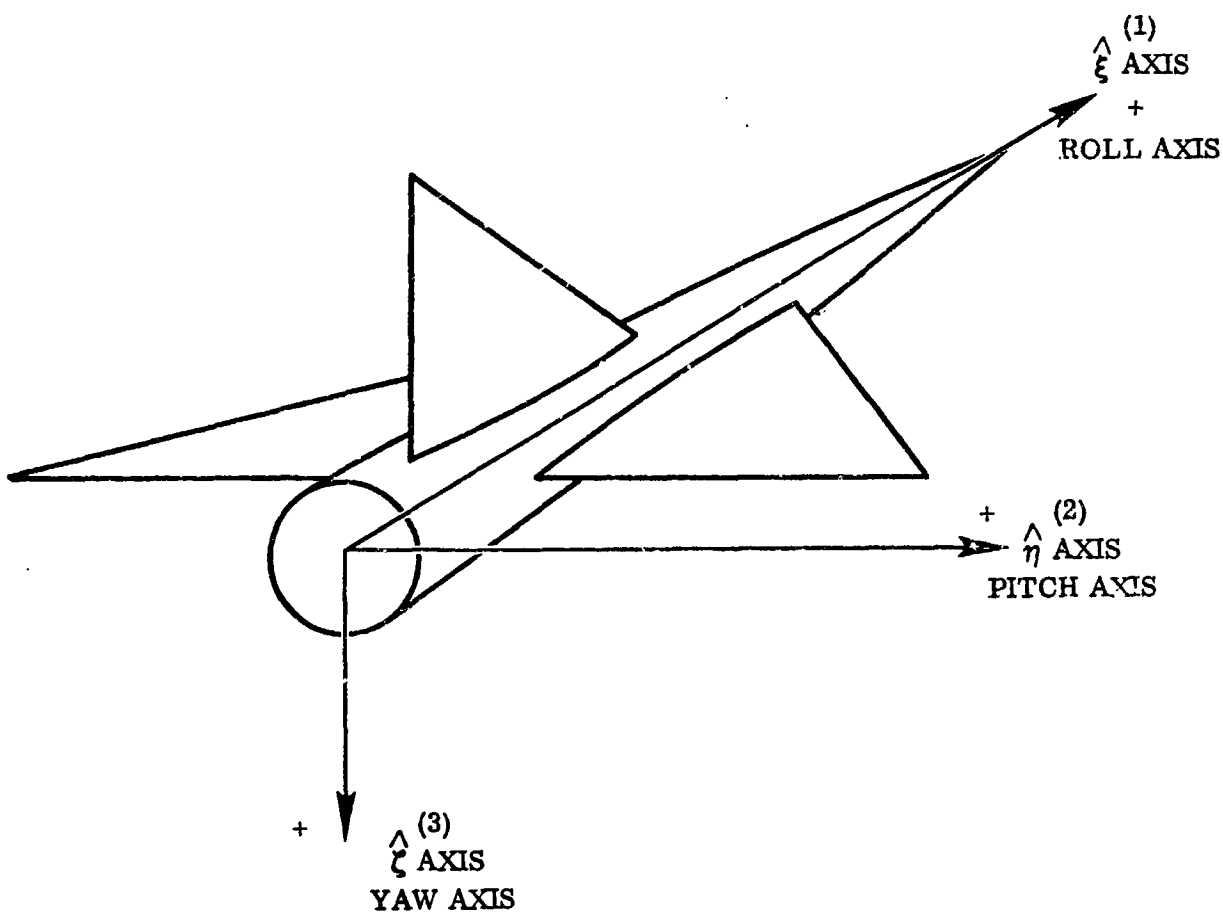


Figure 5-10. Standard $\hat{\xi} - \hat{\eta} - \hat{\zeta}$ Vehicle Coordinate System

25. The aerodynamic axial force which is the aerodynamic force directed along the standard roll axis ($\hat{\xi}$ axis). The algebraic sign is in accordance with the standard $\hat{\xi} - \hat{\eta} - \hat{\zeta}$ coordinate system (see Figure 5-10) (positive forward)

(AXIAL F(AERO))

(LB)

(Z(25))

* References 2, 3, and 4.

26. The aerodynamic normal force which is the aerodynamic force directed along the standard yaw axis ($\hat{\zeta}$ axis). The algebraic sign is in accordance with the standard $\xi - \eta - \zeta$ coordinate system (see Figure 5-10) (positive downward)
(NORM F(AERO)) (LB) (Z(26))
29. The aerodynamic yaw load factor which is directed along the standard pitch axis ($\hat{\eta}$ axis). See Figure 5-10. (Positive to the right.)
(YAW AERO LD) (g) (Z(29))
30. The aerodynamic axial load factor which is directed along the standard roll axis ($\hat{\xi}$ axis). See Figure 5-10. (Positive forward)
(AXIAL AERO LD) (g) (Z(30))
31. The aerodynamic normal load factor which is directed along the standard yaw axis ($\hat{\zeta}$ axis). See Figure 5-10. (Positive downward)
(NORM AERO LD) (g) (Z(31))
32. The thrust load factor along the standard roll axis ($\hat{\xi}$ axis). See Figure 5-1. (Positive forward).
(THRUST LD) (g) (Z(32))
33. Total applied* force load factor along the standard roll axis ($\hat{\xi}$ axis). See Figure 5-10. (Positive forward)
(AXIAL LD) (g) (Z(33))
34. Total aerodynamic load factor defined by the vector addition of output parameters 29., 30., and 31..
(TOTAL AERO LD) (g) (Z(34))
35. Total applied* force load factor defined by the vector addition of output parameters 29., 33., and 31..
(TOTAL LD) (g) (Z(35))
44. Total applied* force component along a vector ($\hat{i}_\theta$) in the horizontal plane which is perpendicular to the plane containing the relative velocity vector and the geocentric radius vector. The positive sense is toward increasing azimuth. See Figure 5-11.
(FORCE (AZIMUTH)) (LB) (Z(44))

* Includes aerodynamic and propulsive forces only - does not include gravitational forces.

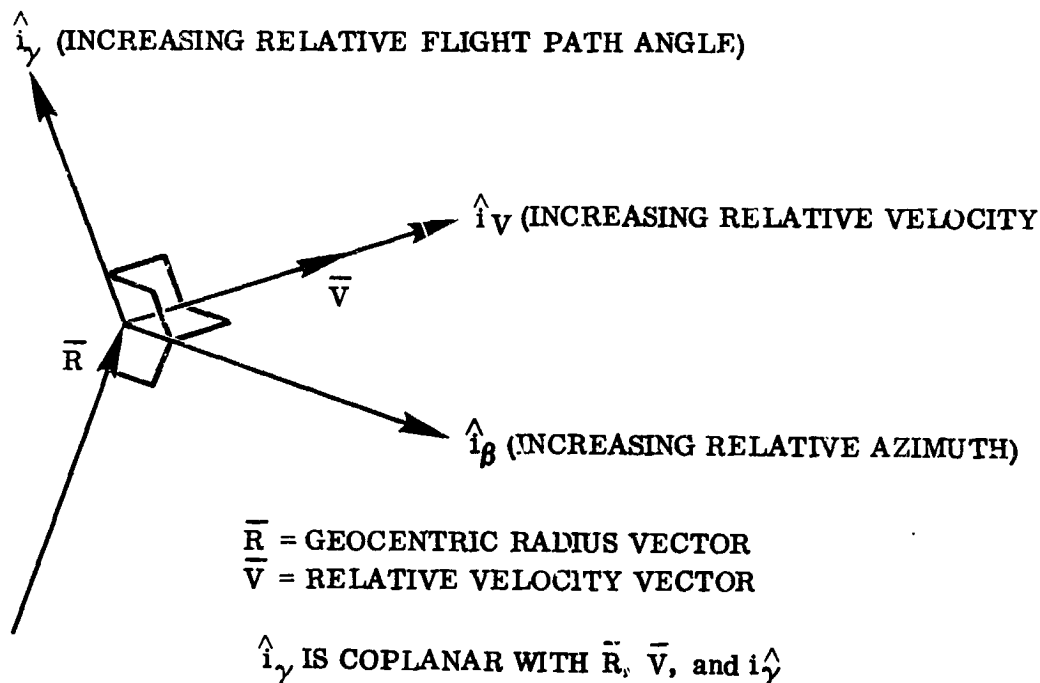


Figure 5-11. Standard $\hat{i}_\beta - \hat{i}_V - \hat{i}_\gamma$ Velocity Coordinate System

45. Total applied* force component along the relative velocity vector ($\hat{i}_V$) with positive sense toward increasing velocity magnitude. See Figure 5-11.
 (FORCE (VEL)) (LB) (Z(45))
46. Total applied* force component along a vector ($\hat{i}_\gamma$) which lies in the plane defined by the relative velocity ($\bar{V}$) and the geocentric radius vector and is perpendicular to $\bar{V}$ with positive sense toward increasing relative flight path angle. See Figure 5-11.
 (FORCE (GAMMA)) (LB) (Z(46))

ATTITUDE. The attitude of the vehicle is defined relative to the current geocentric radius vector ($\bar{R}$) and the current relative velocity vector ($\bar{V}$) by the use of three parameters, the roll (bank) angle (σ), the pitch angle of attack (α), and the yaw angle of attack (λ). The definition of σ , α , and λ is illustrated in Figure 5-12. The current attitude is equivalent to an initial alignment of the vehicle with respect to the $\bar{R}$ and $\bar{V}$ vectors with three subsequent instantaneous sequenced rotations. The vehicle axes (the standard $\hat{\xi} - \hat{\eta} - \hat{\zeta}$ axes shown in Figure 5-10) are initially aligned with the standard $\hat{i}_\beta - \hat{i}_V - \hat{i}_\gamma$ shown in Figure 5-11 such that the $\hat{\xi}$ axis is aligned with the $\hat{i}_V$ axis

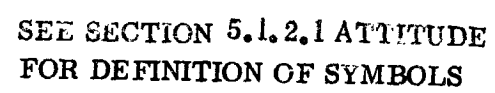
* Includes aerodynamic and propulsive forces only - does not include gravitational forces.

with the same sense, the $\hat{\eta}$ axis is aligned with the $\hat{i}_\beta$ axis with the same sense, and the $\hat{\xi}$ axis is aligned with the $\hat{i}_\gamma$ axis but with opposite sense (see Figure 5-12). The first of the instantaneous sequenced rotations is a rotation about the $\hat{\xi}$ axis through the roll (bank) angle (σ), the second is a rotation about the $\hat{\eta}$ axis through the pitch angle of attack (α), and the last is a rotation about the $\hat{\xi}$ axis through the yaw angle of attack (λ). The algebraic sign of all rotations is in accordance with right-handed rotations of the standard $\hat{\xi} - \hat{\eta} - \hat{\xi}$ coordinate system.

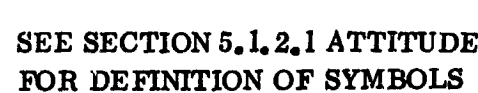
11. Roll (bank) angle σ described in the introductory paragraph of this section and in Figure 5-12. The algebraic sign is in accordance with right-handed rotations of the standard $\hat{\xi} - \hat{\eta} - \hat{\xi}$ coordinate system. Roll (bank) right is positive. See Figure 5-12.
 (ROLL ANGLE) (DEG) (Z(11))
12. Pitch angle of attack α described in the introductory paragraph of this section and in Figure 5-12. This algebraic sign is in accordance with right-handed rotations of the standard $\hat{\xi} - \hat{\eta} - \hat{\xi}$ coordinate system. Pitch up is positive. See Figure 5-12.
 (ALPHA) (DEG) (Z(12))
13. Yaw angle of attack λ described in the introductory paragraph of this section and in Figure 5-12. The algebraic sign is in accordance with right-handed rotations of the standard $\hat{\xi} - \hat{\eta} - \hat{\xi}$ coordinate system. Yaw right is positive. See Figure 5-12.
 (YAW ANGLE) (DEG) (Z(13))
43. Relative pitch angle. This angle is the angle from the current geocentric radius vector (up direction) to the vehicle roll axis ($\hat{\xi}$ axis)
 (PITCH ANGLE) (DEG) (Z(43))
78. The relative pitch rate. This rate is the time rate of change of the relative pitch angle (output parameter 43). The pitch rate is positive for increasing pitch angles.
 (PITCH RATE) (DEG/SEC) (Z(78))

ENVIRONMENT.

9. Dynamic pressure
 (DYN PRESS) (LB/FT²) (Z(9))
10. Total gravitational acceleration
 (TOTAL GRVTY) (FT/SEC²) (Z(10))

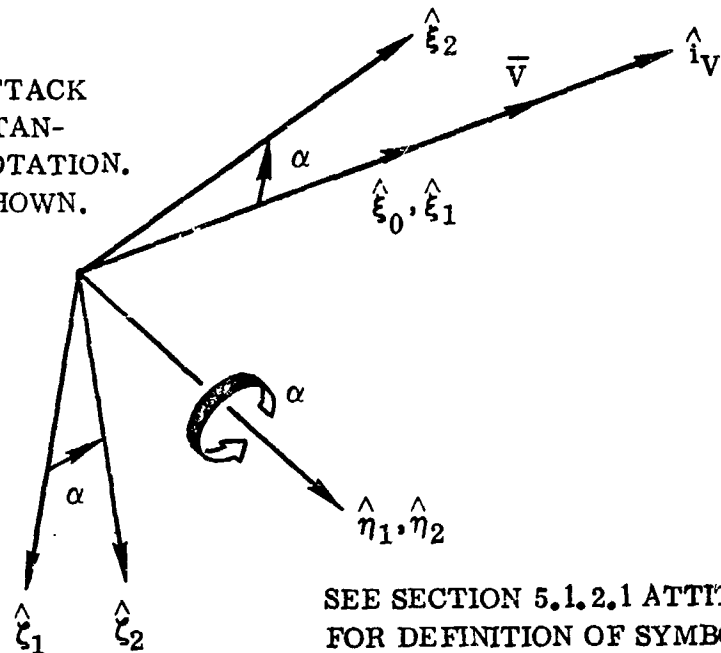


3



0

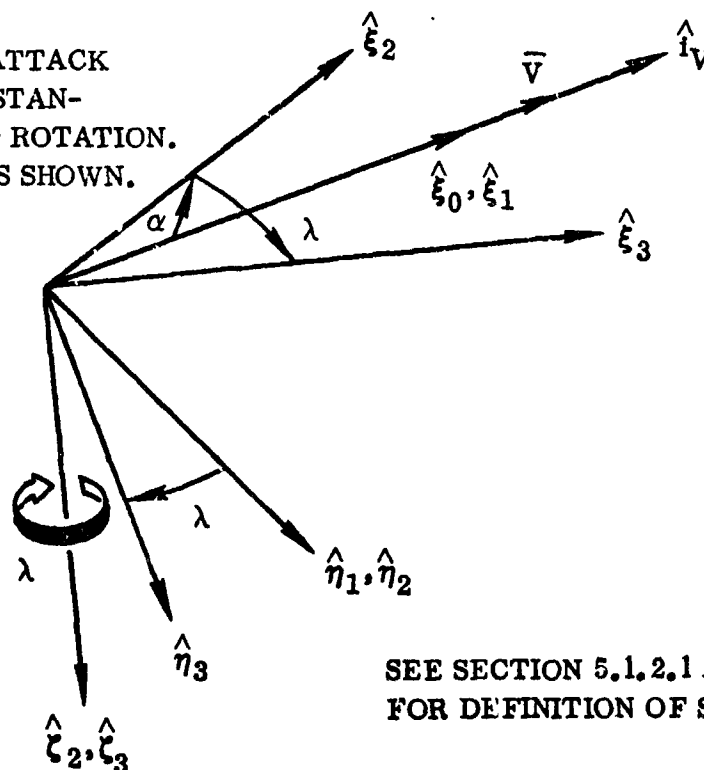
THE PITCH ANGLE OF ATTACK
OF α IS THE SECOND INSTAN-
TANEOUS SEQUENCED ROTATION.
POSITIVE ROTATION IS SHOWN.



SEE SECTION 5.1.2.1 ATTITUDE
FOR DEFINITION OF SYMBOLS

5-12C. SECOND INSTANTANEOUS SEQUENCED ROTATION,
PITCH ANGLE OF ATTACK

THE YAW ANGLE OF ATTACK
OF λ IS THE THIRD INSTAN-
TANEOUS SEQUENCED ROTATION.
POSITIVE ROTATION IS SHOWN.



SEE SECTION 5.1.2.1 ATTITUDE
FOR DEFINITION OF SYMBOLS

5-12D. THIRD INSTANTANEOUS SEQUENCED ROTATION,
YAW ANGLE OF ATTACK

14. Local surface radius of the central body
(SURF RADIUS) (FT) (Z(14))
15. Local gravitational acceleration along the geocentric radius vector. Positive downward.
(RADIAL GRVTY) (FT/SEC²) (Z(15))
16. Local gravitational acceleration along the vector which is tangential to the local meridian with positive sense toward the north
(MERIDIAN GRVTY) (FT/SEC²) (Z(16))
36. Current atmospheric ambient pressure.
(ATM PRESS) (LB/IN²) (Z(36))
37. Current velocity of sound.
(VEL SOUND) (FT/SEC) (Z(37))
38. Heat flux. The heat flux is the product of the current relative velocity (output parameter 2) with the current dynamic pressure (output parameter 9)
(HEAT FLUX) (LB/(FT-SEC)) (Z(38))
39. Heating parameter. The heating parameter is the time integral:

$$\int_{t_0}^t qV dt$$

where:

q is the current dynamic pressure (output parameter 9)

V is the relative velocity (output parameter 2)

t is the absolute time (output parameter -1)

t₀ is the time at the initiation of simulation Section 1 (input parameter "T0")

(HEATING PAR) (LB/FT) (Z(39))

40. Mach number.
(MACH NUMBER) (NONE) (Z(40))

AERODYNAMICS.

21. Aerodynamic yaw force coefficient directed along the standard pitch axis ($\hat{\eta}$ axis). The algebraic sign is in accordance with the standard $\hat{\xi} - \hat{\eta} - \hat{\zeta}$ coordinate system (see Figure 5-10). Positive to the right.
(C (YAW)) (NONE) (Z(21))
22. Aerodynamic axial force coefficient* directed along the standard roll axis ($\hat{\xi}$ axis). The algebraic sign is in accordance with the standard $\hat{\xi} - \hat{\eta} - \hat{\zeta}$ coordinate system (see Figure 5-10). Positive forward.
(C (AXIAL)) (NONE) (Z(22))
23. Aerodynamic normal force coefficient* directed along the standard yaw axis ($\hat{\zeta}$ axis). The algebraic sign is in accordance with the standard $\hat{\xi} - \hat{\eta} - \hat{\zeta}$ coordinate system (see Figure 5-10). Positive downward.
(C (NORMAL)) (NONE) (Z(23))

WEIGHT.

7. Current total weight of the vehicle
(WEIGHT) (LB) (Z(7))
47. Total time rate of change of the vehicle weight (includes GIMAC flow rate if any). Positive for decreasing weight.
(WEIGHT DOT) (LB/SEC) (Z(47))

PROPULSION.

27. Current total thrust acting on the vehicle along the standard roll axis ($\hat{\xi}$ axis). Positive toward the front of the vehicle.
(THRUST) (LB) (Z(27))
48. Current specific impulse for all the vehicle engines and motors in operation.
(ISP) (SEC) (Z(48))

VELOCITY LOSSES.

17. Current total propulsive ideal velocity.
(IDEAL VEL) (FT/SEC) (Z(17))

* Two tables, one for the booster and one for the orbiter, are combined algebraically in Simulation Sections 1, 2, and 3. 5-31

18. Current total gravity relative velocity loss.
(GRAVITY LOSS) (FT/SEC) (Z(18))
19. Current total aerodynamic relative velocity loss.
(DRAG LOSS) (FT/SEC) (Z(19))
20. Current total thrust misalignment relative velocity loss.
(MISALIGN LOSS) (FT/SEC) (Z(20))

5.1.2.2 Optional Output Parameters. These parameters are computed once per integration step and printed at either each integration step or at integral numbers of integration steps according to the value of input parameter "XOUT(K)" - (see Section 4.2.3.5 program control) only if they are called for by the appropriate input.

Orbital Elements. It is possible to compute the instantaneous orbital elements at each integration step during specified simulation sections. To obtain these orbital elements for simulation section "K", it is necessary to input "ORB(K) = 1., ". In addition to the regular orbital elements (output parameters 66, to 77, 91, and 96 to 100) it is possible to compute as part of the orbital elements the instantaneous impact conditions for either oblate or spherical central body surface geometry during simulation section "K" by the input of "ORB(K)" = 2., or 3.,, respectively, (see "ORB(K)" in Section 4.2.3.5 Program Control). In this case ("ORB(K)" = 2., or 3.,) both the regular orbital elements and the instantaneous impact conditions are computed and printed.

Regular Orbital Elements

66.	Inclination, 0 to 180 Deg.		
	(INCLINATION)	(DEG)	(Z(66))
67.	Longitude of the ascending node. Positive to the east of the Greenwich meridian.		
	(NODAL LONG)	(DEG)	(Z(67))
68.	Orbital energy (English units).		
	(ENERGY (FPS) ²)	(FT ² /SEC ²)	(Z(68))
69.	Local circular velocity.		
	(CIRCULAR VEL)	(FT/SEC)	(Z(69))
70.	Geocentric apogee radius magnitude*.		
	(APOGEE RAD)	(FT)	(Z(70))
71.	Apogee velocity*.		
	(APOGEE VEL)	(FT/SEC)	(Z(71))
72.	Orbital energy (metric units).		
	(ENERGY (KPS) ²)	(KM ² /SEC ²)	(Z(72))

*Set equal to zero if parabolic or hyperbolic conditions exist.

73.	Orbital eccentricity.		
	(ECCENTRICITY)	(NONE)	(Z(73))
74.	Orbital parameter. The orbital parameter is the magnitude of the geocentric radius vector at a true anomaly of 90 or 270 degrees.		
	(ORBITAL PAR)	(FT)	(Z(74))
75.	Keplerian area constant.		
	(AREA CONSTANT)	(FT ² /SEC)	(Z(75))
76.	True anomaly measured positively from the perigee in the direction of motion.		
	(TRUE ANOMALY)	(DEG)	(Z(76))
77.	Orbital period.*		
	(PERIOD)	(SEC)	(Z(77))
91.	Argument of perigee measured from the ascending node in the plane of motion and positively in the direction of orbital motion.		
	(PERIGEE ARG)	(DEG)	(Z(91))
96.	Current geocentric perigee radius magnitude.		
	(PERIGEE RAD)	(FT)	Z(93)
97.	Geocentric latitude of perigee		
	(PERIGEE LAT)	(DEG)	Z(97)
98.	Geocentric latitude of apogee		
	(APOGEE LAT)	(DEG)	Z(98)
99.	Perigee altitude.		
	(PERIGEE ALT)	(FT)	Z(99)
100.	Apogee altitude.		
	(APOGEE ALT)	(FT)	Z(100)

* Set equal to zero if parabolic or hyperbolic conditions exist.

Instantaneous Impact Conditions†

84.	Predicted impact latitude.		
	(IMPACT LAT)	(DEG)	(Z(84))
85.	Predicted impact longitude.		
	(IMPACT LONG)	(DEG)	(Z(85))
86.	Surface radius at impact.		
	(IMPACT RADIUS)	(FT)	(Z(86))
87.	Time to impact. This time is measured from the current conditions to the predicted impact point.		
	(TIME TO IMPACT)	(SEC)	(Z(87))
88.	Predicted impact longitude for a non-rotating central body.		
	(INST IMPCT LNG)	(DEG)	(Z(88))
89.	Central angle to impact. This angle is the central angle between the current position and the predicted impact point.		
	(ANGLE TO IMPCT)	(DEG)	(Z(89))

Optional Vector Components. It is possible to compute and output position, velocity, and acceleration vector components in the two coordinate systems which are defined in Section 2.2.2.10 Position, Velocity, and Acceleration Vector Components (see Figures 5-13 and 5-14). The inertial coordinate system orientation is specified by the input parameters "AZMI" and "GTIP". Two options are available. "Option A" and "Option B". Selection of an option is accomplished with the input parameter "ZOUT" (see "ZOUT" Section 4.2.3.5 Program Control).

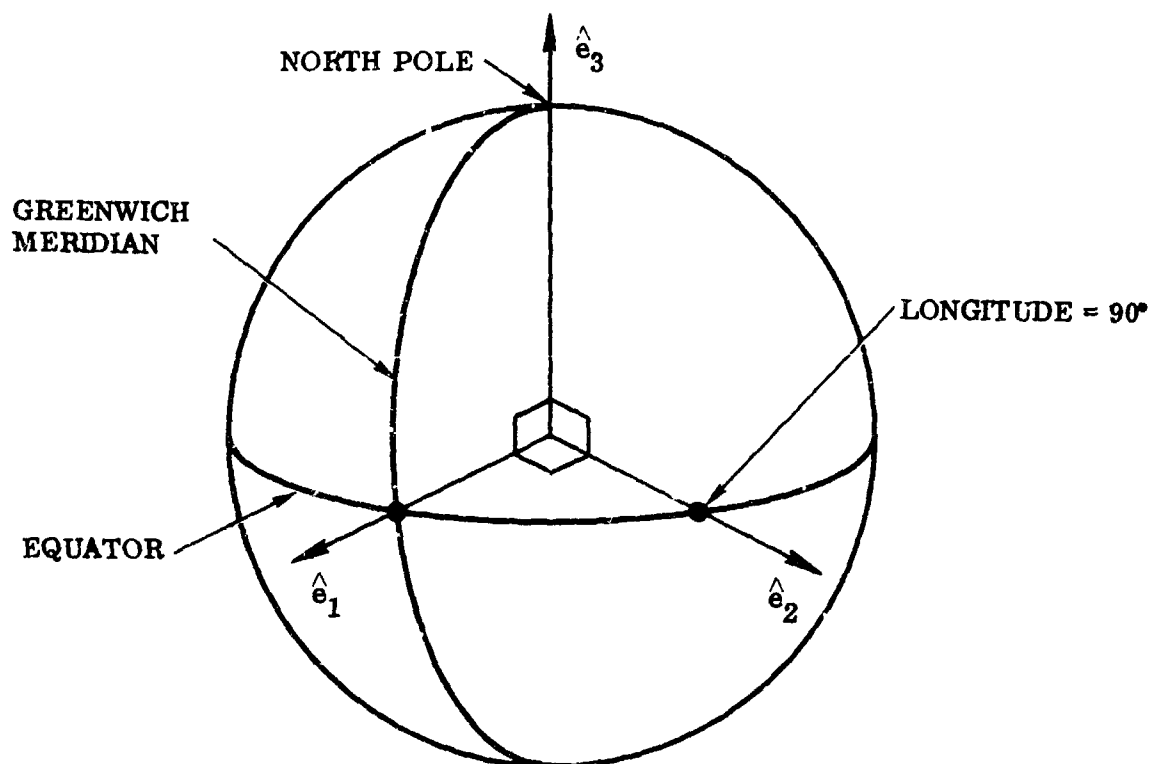
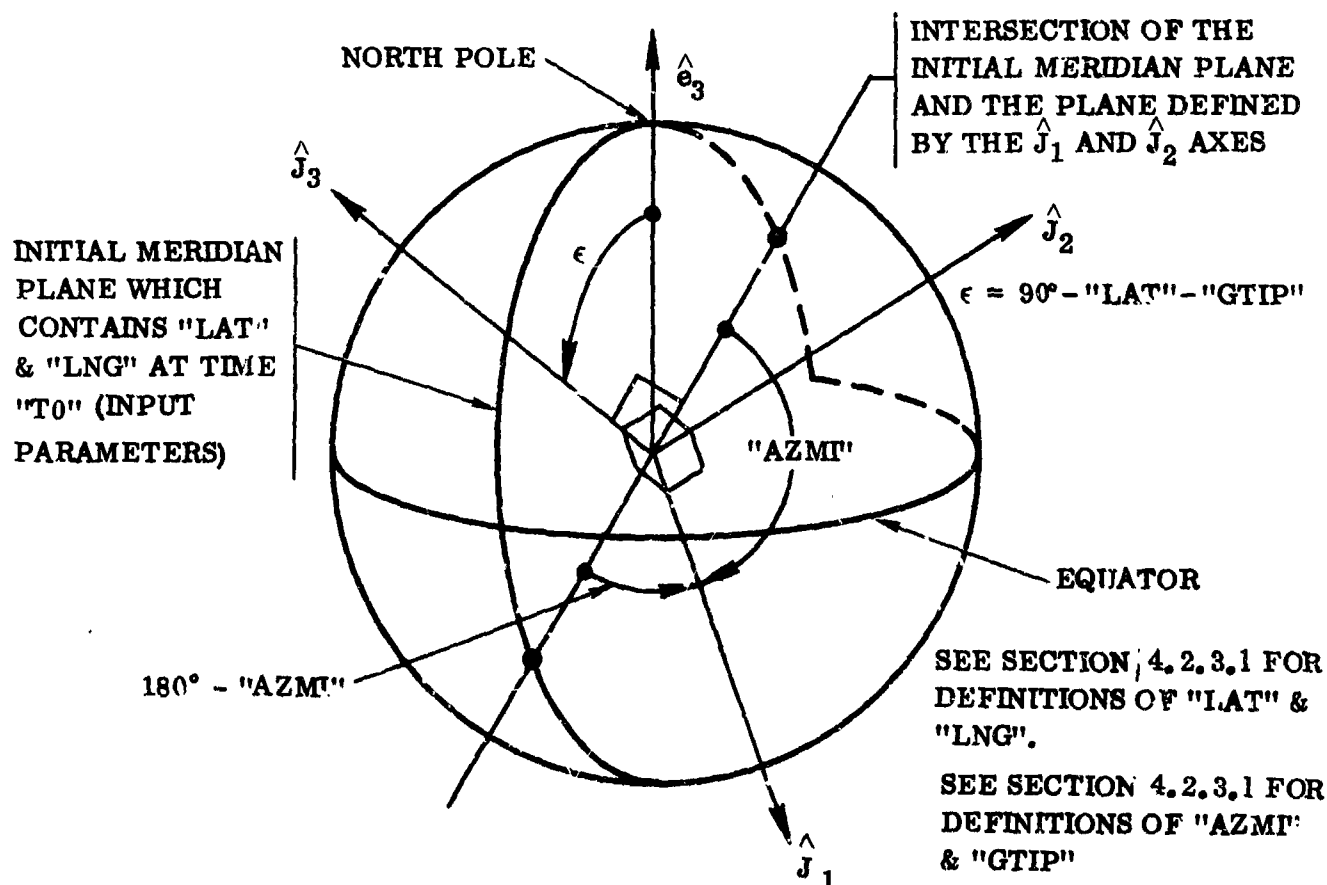
Option A

53.	Geocentric radius vector component along the $\hat{e}_1$ axis.		
	(RADIUS(1))	(FT)	(Z(53))
54.	Geocentric radius vector component along the $\hat{e}_2$ axis		
	(RADIUS(2))	(FT)	(Z(54))

†Computed only if the conditions which are specified in Sections 2.2.2.10 Orbital Elements and Section 4.2.3.5 Program Control are satisfied. If these conditions are not met, output parameters 84 to and including 89 are set equal to zero.

55.	Geocentric radius vector component along the $\hat{e}_3$ axis		
	(RADIUS(3))	(FT)	(Z(55))
56.	Relative velocity vector component along the $\hat{e}_1$ axis.		
	(VELOCITY(1))	(FT/SEC)	(Z(56))
57.	Relative velocity vector component along the $\hat{e}_2$ axis.		
	(VELOCITY(2))	(FT/SEC)	(Z(57))
58.	Relative velocity vector component along the $\hat{e}_3$ axis.		
	(VELOCITY(3))	(FT/SEC)	(Z(58))
59.	Net* acceleration vector component along the $\hat{J}_1$ axis.		
	(ACCEL(1))	(FT/SEC ²)	(Z(59))
60.	Net* acceleration vector component along the $\hat{J}_2$ axis.		
	(ACCEL(2))	(FT/SEC ²)	(Z(60))
61.	Net* acceleration vector component along the $\hat{J}_3$ axis.		
	(ACCEL(3))	(FT/SEC ²)	(Z(61))

*Includes aerodynamic, propulsive, and gravitational accelerations

Figure 5-13. The $\hat{e}_1 - \hat{e}_2 - \hat{e}_3$ Earth Fixed Coordinate SystemFigure 5-14. The $\hat{J}_1 - \hat{J}_2 - \hat{J}_3$ Inertial Coordinate System

Option B

53. Geocentric radius vector component along the $\hat{J}_1$ axis.
(RADIUS(1)) (FT) (Z(53))
54. Geocentric radius vector component along the $\hat{J}_2$ axis.
(RADIUS(2)) (FT) (Z(54))
55. Geocentric radius vector component along the $\hat{J}_3$ axis.
(RADIUS(3)) (FT) (Z(55))
56. Inertial velocity vector component along the $\hat{J}_1$ axis.
(VELOCITY(1)) (FT/SEC) (Z(56))
57. Inertial velocity vector component along the $\hat{J}_2$ axis.
(VELOCITY (2)) (FT/SEC) (Z(57))
58. Inertial velocity vector component along the $\hat{J}_3$ axis.
(VELOCITY (3)) (FT/SEC) (Z(58))
59. Total applied* force acceleration vector along the $\hat{J}_1$ axis.
(ACCEL(A1)) (FT/SEC²) (Z(59))
60. Total applied* force acceleration vector along the $\hat{J}_2$ axis.
(ACCEL(A2)) (FT/SEC²) (Z(60))
61. Total applied* force acceleration vector along the $\hat{J}_3$ axis.
(ACCEL(A3)) (FT/SEC²) (Z(61))
62. Gravitational acceleration vector component along the $\hat{J}_1$ axis.
(ACCEL(G1)) (FT/SEC²) (Z(62))
63. Gravitational acceleration vector component along the $\hat{J}_2$ axis.
(ACCEL(G2)) (FT/SEC²) (Z(63))
64. Gravitational acceleration vector component along the $\hat{J}_3$ axis.
(ACCEL(G3)) (FT/SEC²) (Z(64))

*Includes aerodynamic and propulsive forces only - does not include gravitational forces.

Position in Radar Coordinates.

It is possible to compute and output the vehicle position in radar coordinates from the initial trajectory coordinates defined by input parameters "LAT", "LNG", and either "ALT0" or "RAD", and from the target radar site coordinates defined by input parameters "TLAT(M)", "TLNG(M)", and "RDALT(M)". Specification of these radar coordinate position vectors is accomplished with the input parameter "WOUT(K)" (see "WOUT(K)" in Section 4.2.3.5 Program Control).

52.	Azimuth of the vehicle measured at the initial trajectory coordinates.		
	(AZIMUTH)	(DEG)	(Z(52))
79.	Slant range of the vehicle measured at the initial trajectory coordinates.		
	(SLANT RANGE)	(FT)	(Z(79))
80.	Elevation angle of the vehicle measured at the initial trajectory coordinates.		
	(ELEV ANGLE)	(DEG)	(Z(80))
81.	Azimuth of the vehicle measured at the current target radar site coordinates.		
	(AZIMUTH(T))	(DEG)	(Z(81))
82.	Slant range of the vehicle measured at the current target radar site coordinates.		
	(SLANT RANGE(T))	(FT)	(Z(82))
83.	Elevation angle of the vehicle measured at the current target radar site coordinates.		
	(ELEV ANGLE(T))	(DEG)	(Z(83))
93.	Azimuth to the target site as seen from the vehicle.		
	(AZIMUTH(V))	(DEG)	(Z(93))
94.	Difference between the azimuth of the target site from the vehicle and the azimuth of the vehicle.		
	(AZIMUTH DIFF)	(DEG)	(Z(94))
95.	Surface Range in nautical miles between the vehicle and the target site.		
	(SURF RANGE)	(N. MI.)	(Z(95))

5.1.2.3 Output Parameter Index. This Section, Output Parameter Index, lists all the possible output parameters with their code numbers, printed output titles, units, and corresponding internal parameters.

OUTPUT PARAMETER CODE NUMBER	PRINTED OUTPUT TITLE	UNITS	CORRESPONDING INTERNAL PARAMETER
1. <u>REGULAR OUTPUT PARAMETERS</u>			
<u>TIME</u> (Section 5.1.1)			
-1	ABS TIME	SEC	T
0	REL TIME	SEC	TR
<u>POSITION</u>			
1	RADIUS	FT	Z(1)
5	LATITUDE	DEG	Z(5)
6	LONGITUDE	DEG	Z(6)
8	ALTITUDE	FT	Z(8)
49	DOWN RANGE	DEG	Z(49)
50	CROSS RANGE	DEG	Z(50)
51	TARGET DIST	DEG	Z(51)
<u>VELOCITY</u>			
2	REL VELOCITY	FT/ SEC	Z(2)
3	REL GAMMA	DEG	Z(3)
4	REL AZIMUTH	DEG	Z(4)
41	INRTL VELOCITY	FT/ SEC	Z(41)
42	INRTL GAMMA	DEG	Z(42)
65	INRTL AZIMUTH	DEG	Z(65)

OUTPUT PARAMETER CODE NUMBER	PRINTED OUTPUT TITLE	UNITS	CORRESPONDING INTERNAL PARAMETER
<u>ACCELERATION</u>			
24	YAW F(AERO)	LB	Z(24)
25	AXIAL F(AERO)	LB	Z(25)
26	NORM F(AERO)	LB	Z(26)
29	YAW AERO LD	g	Z(29)
30	AXIAL AERO LD	g	Z(30)
31	NORM AERO LD	g	Z(31)
32	THRUST LD	g	Z(32)
33	AXIAL LD	g	Z(33)
34	TOTAL AERO LD	g	Z(34)
35	TOTAL LD	g	Z(35)
44	FORCE (AZIMUTH)	LB	Z(44)
45	FORCE (VEL)	LB	Z(45)
46	FORCE (GAMMA)	LB	Z(46)
<u>ATTITUDE</u>			
11	ROLL ANGLE	DEG	Z(11)
12	ALPHA	DEG	Z(12)
13	YAW ANGLE	DEG	Z(13)
43	PITCH ANGLE	DEG	Z(43)
78	PITCH RATE	DEG/ SEC	Z(78)
<u>ENVIRONMENT</u>			
9	DYN PRESS	LB/FT ²	Z(9)
10	TOTAL GRVTY	FT/SEC ²	Z(10)
14	SURF RADIUS	FT	Z(14)
15	RADIAL GRVTY	FT/SEC ²	Z(15)
16	MERIDIAN GRVTY	FT/SEC ²	Z(16)
36	ATM PRESS	LB/IN ²	Z(36)
37	VEL SOUND	FT/SEC	Z(37)
38	HEAT FLUX	LB/(FT- SEC)	Z(38)
39	HEATING PAR	LB/FT	Z(39)
40	MACH NUMBER	NONE	Z(40)
* 92	LOG ₁₀ [Z(9)]	—	Z(92)

*Not printed output

OUTPUT PARAMETER CODE NUMBER	PRINTED OUTPUT TITLE	UNITS	CORRESPONDING INTERNAL PARAMETER
<u>AERODYNAMICS</u>			
21	C (YAW)	NONE	Z(21)
22	C (AXIAL)	NONE	Z(22)
23	C (NORMAL)	NONE	Z(23)
<u>WEIGHT</u>			
7	WEIGHT	LB	Z(7)
47	WEIGHT DOT	LB/SEC	Z(47)
<u>PROPULSION</u>			
27	THRUST	LB	Z(27)
48	ISP	SEC	Z(48)
<u>VELOCITY LOSSES</u>			
17	IDEAL VEL	FT/SEC	Z(17)
18	GRAVITY LOSS	FT/SEC	Z(18)
19	DRAG LOSS	FT/SEC	Z(19)
20	MISALIGN LOSS	FT/SEC	Z(20)
2. <u>OPTIONAL OUTPUT PARAMETERS</u>			
a. <u>ORBITAL ELEMENTS</u>			
(1) <u>REGULAR ORBITAL ELEMENTS</u>			
66	INCLINATION	DEG	Z(66)
67	NODAL LONG	DEG	Z(67)
68	ENERGY (FPS) ²	FT ² / SEC ²	Z(68)
69	CIRCULAR VEL	FT/SEC	Z(69)
70	APOGEE RAD	FT	Z(70)
71	APOGEE VEL	FT/SEC	Z(71)
72	ENERGY (KPS) ²	KM ² / SEC ²	Z(72)
73	ECCENTRICITY	NONE	Z(73)
74	ORBITAL PAR	FT	Z(74)
75	AREA CONST	FT ² /SEC	Z(75)
76	TRUE ANOMALY	DEG	Z(76)
77	PERIOD	SEC	Z(77)
91	PERIGEE ARG	SEC	Z(91)
96	PERIGEE RAD	FT	Z(96)
97	PERIGEE LAT	DEG	Z(97)
98	APOGEE LAT	DEG	Z(98)
99	PERIGEE ALT	FT	Z(99)
100	APOGEE ALT	FT	Z(100)

OUTPUT PARAMETER CODE NUMBER	PRINTED OUTPUT TITLE	UNITS	CORRESPONDING INTERNAL PARAMETER
2. <u>OPTIONAL OUTPUT PARAMETERS</u> (Continued)			
a. <u>ORBITAL ELEMENTS</u> (Continued)			
(2) <u>INSTANTANEOUS IMPACT CONDITIONS</u>			
84	IMPACT LAT	DEG	Z(84)
85	IMPACT LONG	DEG	Z(85)
86	IMPACT RADIUS	FT	Z(86)
87	TIME TO IMPACT	SEC	Z(87)
88	INST IMPCT LNG	DEG	Z(88)
89	ANGLE TO IMPCT	DEG	Z(89)
b. <u>OPTIONAL VECTOR COMPONENTS</u>			
(1) <u>OPTION A</u>			
53	RADIUS(1)	FT	Z(53)
54	RADIUS(2)	FT	Z(54)
55	RADIUS(3)	FT	Z(55)
56	VELOCITY(1)	FT/SEC	Z(56)
57	VELOCITY(2)	FT/SEC	Z(57)
58	VELOCITY(3)	FT/SEC	Z(58)
59	ACCEL(1)	FT/SEC ²	Z(59)
60	ACCEL(2)	FT/SEC ²	Z(60)
61	ACCEL(3)	FT/SEC ²	Z(61)
(2) <u>OPTION B</u>			
53	RADIUS(1)	FT	Z(53)
54	RADIUS(2)	FT	Z(54)
55	RADIUS(3)	FT	Z(55)
56	VELOCITY(1)	FT/SEC	Z(56)
57	VELOCITY(2)	FT/SEC	Z(57)

CODE PARAMETER CODE NUMBER	PRINTED OUTPUT TITLE	UNITS	CORRESPONDING INTERNAL PARAMETER
(2) <u>OPTION B</u>		(Contd)	
58	VELOCITY(3)	FT/SEC	Z(58)
59	ACCEL(A1)	FT/SEC ²	Z(59)
60	ACCEL(A2)	FT/SEC ²	Z(60)
61	ACCEL(A3)	FT/SEC ²	Z(61)
62	ACCEL(G1)	FT/SEC ²	Z(62)
63	ACCEL(G2)	FT/SEC ²	Z(63)
64	ACCEL(G3)	FT/SEC ²	Z(64)
c. <u>POSITION IN RADAR COORDINATES</u>			
52	AZIMUTH	DEG	Z(52)
79	SLANT RANGE	FT	Z(79)
80	ELEV ANGLE	DEG	Z(80)
81	AZIMUTH(T)	DEG	Z(81)
82	SLANT RANGE(T)	FT	Z(82)
83	ELEV ANGLE(T)	DEG	Z(83)
93	AZIMUTH(V)	DEG	Z(93)
94	AZIMUTH DIFF	DEG	Z(94)
95	SURF RANGE	N. MI.	Z(95)

5.2 ERROR DIAGNOSTICS

The SSSP was structured such as to minimize the number of aborted runs. This is exemplified by the very nature of the building-block process utilized in its construction (preset ascent trajectory, etc.) and by the building-in of default values for the most important parameters or by initializing the parameters at a zero value. As was mentioned earlier the majority of the aborted cases are usually caused by misplacement of inputs within the basic input data blocks (mixing input) or by not inputting all of the necessary parameters required for successful initiation of the synthesis process. It is therefore suggested that the user first examine these two causes if problems develop prior to setting the debug flags as discussed in the introduction to this section.

Problems will normally occur when analysis of a new configuration is begun where, for example, basic thrust levels may be set but the gross vehicle weight is under-estimated and hence the thrust/weight at liftoff is insufficient for a reasonable ascent trajectory to be simulated. If problems of this nature occur, the SSSP provides for a number of error diagnostics to be output. These output, along with the "abbreviated synthesis iteration" output discussed above, will usually suffice in tracking and solving the problem that has occurred. Possible solutions to the problem example mentioned above would be to increase the number of engines used during liftoff or to attempt an alternative approach such as specifying the liftoff thrust/weight (solve for the engine thrust level) or specifying the booster gross weight (solve for the system payload). Using techniques such as these is normally the most expedient method for diagnosing either (1) problems associated with the basic design or (2) problems arising from unreasonable input estimates (trajectory or weight sizing). In this manner a "solution" is obtained from which to work rather than analyzing debug printout which is usually uneconomical in terms of the manhours expended.

The following subsections discuss the basic SSSP error diagnostics in terms of the particular OVERLAY and subroutine from which the diagnostic is printed and

offers suggestions concerning possible corrective measures to be used.

5.2.1 SYNTHESIS DRIVER. Error diagnostics occurring here usually pertain to the tests made on the basic synthesis iteration process.

ITER8. The following diagnostics may be output during the basic synthesis iteration (and after the trajectory simulation) on the orbiter mass ratio:

1. "FAILED TO CONVERGE - MU = ____ INPUT TOL = ____ TOL = ____"
2. "ITERATION COUNTER EXCEEDED = ____"
3. "ITERATION COUNTER EXCEEDED = ____ INPUT TOL = ____
TOL = ____"

This may be corrected by increasing the iteration counter limit (see \$DATA2, Section 4.2.2), expanding the allowable tolerance, or inputting better estimates for the various parameters for the next case such as \$DATA2 inputs IDVEL, QMAX and booster \$DATA3 input C(214).

RANGE. The following may be output if \$DATA2 input FLYBCK = 1.,:

1. "STAGING ALTITUDE GREATER THAN 250000 FT - FLYBACK RANGE
EQUATIONS INVALID"
2. "STAGING GAMMA GREATER THAN 14 DEG - FLYBACK RANGE
EQUATIONS INVALID"
3. "APOGEE ALTITUDE GREATER THAN 300000 FT - FLYBACK RANGE
EQUATIONS INVALID"

Any one or all of these diagnostics will occur if the current staging conditions from the ascent trajectory simulation have exceeded the capability of the built-in range data. At this point the range increment is set to the maximum default value and the iteration process is allowed to continue (see Section 2.3.4.1). This may be corrected by reshaping the ascent trajectory to lower the staging point or by selecting an alternate method by calculating the booster reference range requirement.

5.2.2 TRAJECTORY SIMULATION. Error diagnostics here pertain to the GTSM subprogram and usually occur due to unreasonable estimates or requirements for

the ascent trajectory (initial pitch rate or section terminal condition) or the booster entry trajectory if used (pitch and bank schedules, etc.).

GTSM. The following may be output:

1. "***** VEHICLE ATTEMPTED SUBTERRANEAN FLIGHT *****"
2. "***** TABLE ARGUMENT ERROR *****"

For the ascent trajectory this may be corrected by improving the initial pitch rate or linear cotangent steering control estimates and usually implies that the current acceleration levels are causing the iteration scheme to be hypersensitive. The table argument error implies that the current altitude (sub-surface) has exceeded the limits of the altitude arguments in the atmosphere tables.

ATMOS. The following may be output:

1. "***** FAILURE IN THE TABLE TOOK-UP OF SUBROUTINE ATMOS
DURING SIMULATION SECTION _____ *****"

This occurs if the current altitude has exceeded the limits of the altitude arguments in the atmosphere tables or there is an error in the input tables.

XNEW1. The following may be output during initial ascent:

1. "***** EXCEEDED MAXIMUM NUMBER OF ITERATIONS
IN XNEW1 *****"
2. "***** CONTROL VARIABLE EXCEEDED LIMIT IN XNEW1
***** CONTROL VARIABLE _____ = _____
LIMITS = _____ TO _____"
3. "***** END CONDITIONS ARE INVARIANT TO CONTROL
PARAMETERS *****"

This may be corrected by increasing the number of allowable iterations (see \$DATA1, Section 4.2.2), expanding the limits on the control or improving the estimates for the control for the next case (poor numerical derivatives have been formed). The third diagnostic will not occur during the ascent trajectory due to nature of the built-in iteration scheme unless an attempt has been made to override the scheme.

XNEW4. Similar diagnostics to those illustrated for XNEW1 may be output during the orbiter ascent phase. The corrective measures also apply here.

AERO2. The following may be output:

1. "**** FAILURE IN THE TABLE LOOK-UP OF SUBROUTINE AERO2
DURING SIMULATION SECTION _____ ****"

This implies that table arguments have been exceeded or that there is an error in the input tables.

AERO3. A similar diagnostic to that illustrated for AERO2 may be output with the same probable causes.

INTGRT. The following may be output:

1. "**** STEPSIZE IS LESS THAN THE INPUT STEPSIZE IN SIMULATION
SECTION _____ **** (error printout follows, see Section 2.2.2)"

Recheck integration tolerances placed on state variables or tighten input stepsize.

2. "**** TERMINATE SECTION _____ ON BACKUP TIME **** (times and tolerances output follows)"
3. "**** FAILED IN BACKUP TIME SECTION _____ TERMINATION ****
(times and tolerances output follows)"
4. "**** FAILED IN NORMAL SECTION _____ TERMINATION **** (times and tolerances output follows)"

Re-examine ascent trajectory profile and vehicle characteristics or entry trajectory file (if used) from the given staging conditions; reshape profiles if necessary.

ANGLE. The following output may occur:

1. "**** FAILURE IN THE TABLE LOOKUP OF SUBROUTINE ANGLE
DURING SIMULATION SECTION _____ ****"

This may be corrected by re-examination of the input pitch rate tables (for completeness) with the current vehicle attitude control model, particularly in conjunction with the booster entry trajectory, if used.

5.2.3 WEIGHT/VOLUME SIZING. Error diagnostics here pertain to the WTVOL subprogram and are usually caused by problems associated with configuration weight sizing bases or propulsion system requirements specified by input.

WTVOL. The following may be output:

1. "***** RUN ABORTED IN WTVOL, FIXED THRUST LOW "

Re-examine the basis for specifying the propulsion system thrust levels (see Section 4.3.1, Propulsion Option). Possible solutions to the problem would be to increase the number of engines used during liftoff or to attempt an alternative approach such as specifying the liftoff thrust/weight (solve for the engine thrust level) or specifying the booster gross weight (solve for the system payload), see Section 4.3.5.

2. "***** ITERATION COUNT EXCEEDED *****"

This is used to signify that various Basic Synthesis Options (Section 2.3.2) which utilize iterative techniques for solution have not converged within the allowable number of iterations. This may be usually corrected by relaxing the constraint or by improving the input estimates for the control for the next case.

WTSCH. A NAMELIST* output will be printed automatically from this subroutine during any pass (orbiter or booster) in which the following baseline limits have been exceeded as a result of the sizing:

booster gross weight $> 20 \times 10^6$ lbs., or
orbiter gross weight $> 10 \times 10^6$ lbs.

This NAMELIST* contains the significant internally computed variables and is useful for tracking the design deficiency (see use of the \$DATA3 input NLISTO discussed in the introduction to this section).

SOLVE. The following is output:

1. "***** BLOWUP IN SOLVE *****"

This implies that the sizing in subroutine WTSCH is divergent for a given stage.

This may usually be corrected by examining the basis for this sizing, viz., the

*"MASS1" and "MASS2." See Volume II for a complete list of computed variables.

\$DATA3 scaling coefficients, or by improving such input estimates as the stage gross weight, mass ratios, etc., (\$DATA3), or the bases for the overall synthesis iteration such as the \$DATA2 input IDVEL.

6. SAMPLE CASE

This section discusses the SSSP sample case listed in Appendix VII. The data base for each input data block will be highlighted in what follows.

6.1 VEHICLE SIZING DATA BASE

The sample case involves a LO_2/LH_2 fueled Space Shuttle configuration which existed as an initial point design developed by personnel at the NASA/MSC in early 1970. This configuration data base was converted to the SSSP input requirements for comparison of the resulting performance. Each stage has a single vertical surface and wings with a 14 degree leading edge sweep. The theoretical (gross) wing area is fixed and the orbiter and booster wing loading is computed internally at the initial entry and initial flyback conditions, respectively. The thermal protection system assumes coverage of the total body wetted area excluding the aerodynamic surfaces. However the corresponding weight coefficient, $C(181)$, is an average value that is representative of a combined insulation-cover panel weight $< 0.75 \text{ lb/ft}^2$ on the upper surface and $> 1.75 \text{ lb/ft}^2$ on the lower surface.

The orbiter has two main engines and the booster has eleven. Both stages have fixed gimble system weights. The subsonic cruise engines for both stages operate with LH_2 propellant stored in main tanks. Each stage has a 1% contingency factor applied to the dry weight.

The system payload volume is fixed at $10,600 \text{ ft}^3$. The orbiter main propellant flight performance reserves are based on 225 fps total characteristic velocity ($\approx 0.75\%$ of mission velocity to parking orbit insertion) with a 1500 fps incremental velocity requirement (O/F mixture ratio = 5) reserved for the post-insertion or on-orbit maneuvers. The booster flight performance reserve is fixed at 1370 lbs. of propellants. For both stages the main impulse propellants (including reserves) utilize an O/F mixture ratio = 6 for the sizing basis. The booster sizing base includes fixing the main impulse mass ratio at a value of 3.045.

6.2 SYNTHESIS ITERATION DATA BASE

The stage burn sequence selected was that of the sequential stage burn. The orbiter main engine unit thrust (vacuum) was fixed at 470 K lbs/engine and the booster main engine unit thrust at vacuum conditions was set at 0.968556 % of the orbiter thrust level (TRATIO). The specific impulses values for each ascent flight phase were constant for each stage.

The booster gross weight was specified at an input value of 3,384,390 lbs. and the payload estimate of 36,000 lbs. was input to the orbiter \$DATA3 block under the coefficient C(103).

The booster reference range method selected was that based on the staging dynamic pressure with the cruise fuel method being the simplified single segment mode of operation.

6.3 TRAJECTORY SIMULATION DATA BASE

The ascent trajectory mission profile includes the following bases:

1. Built-in atmosphere tables (Appendix III) used from liftoff to parking orbit insertion.
2. Table input combined axial aerodynamic characteristics for boost phase only, no normal aero. coefficients and no aero. for orbiter ascent phase.
3. Launch pad at KSC coordinates, launch azimuth = 37.65 degrees.
4. Fourteen-second vertical rise from liftoff, ten-second pitchover, target to a 2 psf staging dynamic pressure ($\log_{10} 2 = 0.3010$), linear cotangent steering during orbiter burn.
5. Throttle booster engines at axial load of 2.5 g limit, orbiter engines at 3.0 g limit.
6. Terminate simulation sections 01 on 2.5 g limit; (02 automatic on booster propellant depletion); 03 and 04 on 3-seconds and 4-seconds (relative), respectively; 05 on 80 seconds (relative); 06 on 3.0 g limit; 07 automatic on specified insertion velocity.
7. Perigee parking orbit insertion at 51/100 n. mi.

For convenience Appendix VII includes a listing of the \$DATA1 input cards for the sample case. This should be used to illustrate the necessary input parameters and for comparison with the normal output of the \$DATA1 input which includes parameters which are computed by the synthesis process and are therefore shown only with their default values.

7. REFERENCES

1. Tramonti, L. G., "General Trajectory Simulation Module (GTSM)," Convair division of General Dynamics, Report GDC-DC68-005, 1 September 1968, Unclassified.
2. "Mass Properties Control Requirements for Missiles and Space Vehicles," USAF Specification No. MIL-M-38810, 15 December 1964.
3. "Mass Properties Standards," NASA Specification SP-6004-CM18-001-1.
4. "Project Apollo Coordinate Systems Standards," NASA Report No. SE-008-001-1, June 1965.
5. Townsend, Jr., et al., "Orbital Flight Handbook for the National Aeronautics and Space Administration," Vol. I, Chapter II, Physical Data, Martin Space Systems Division, 1963.
6. Merson, R. H., "An Operational Method for the Study of Integration Processes," Proceedings of a Symposium on Data Processing, Weapons Research Establishment, Salisbury, South Australia, 1957.
7. Fox, L., "Numerical Solution of Ordinary and Partial Differential Equations," Addison-Wesley Publishing Co., Inc., Reading, Mass., 1962.
8. Wilson, L. G., "Trajex Gravity Program," General Dynamics Convair Division LVP Aeroballistics Letter ABL-68-009, 11 January 1968.
9. Vaughan, W. W., "Natural Environment Criteria for the NASA Space Shuttle Program," George C. Marshall Space Flight Center, NASA TM X-53973, 9 February 1970, Unclassified.

APPENDIX I

ELEMENTAL COORDINATE TRANSFORMATIONS

An elemental coordinate transformation is defined as a transformation from one specified orthogonal coordinate system to another which results from a single rotation about one of the axes of the original coordinate system. Since, for these applications, it is assumed that the coordinate systems are three-dimensional right-handed systems, there are three elemental coordinate transformations.

Let:

- A = Any constant vector
- A_K = The Kth component of $\bar{A}$ along the $\hat{i}_K$ axis
- B_K = The Kth component of A along the $\hat{j}_K$ axis
- $\hat{i}_K$ = The Kth unit vector axis of the initial coordinate system
- $\hat{j}_K$ = The Kth unit vector axis of the resulting coordinate system
- K = Subscript which defines one of the three corresponding axes in both coordinate systems. $K \neq L$, $K \neq M$
- L = Subscript which defines one of the three corresponding axes in both coordinate systems. $L \neq K$, $L \neq M$
- M = Subscript which defines one of the three corresponding axes in both coordinate systems. $M \neq K$, $M \neq L$
- θ_K = The Kth angle of rotation about the $\hat{i}_K$ axis which defines the orientation of the resulting coordinate system axes $\hat{j}_L$, $\hat{j}_M$, with respect to the initial coordinate system axes $\hat{i}_L$, $\hat{i}_M$.

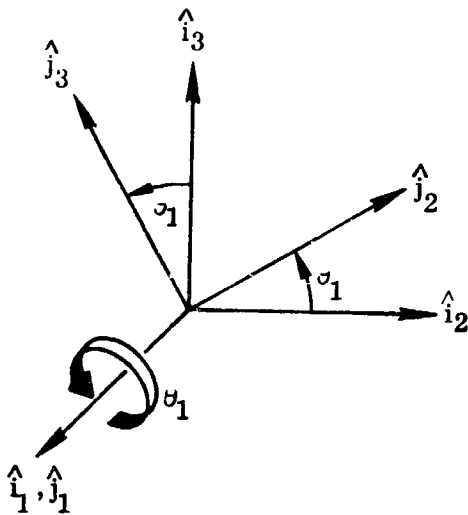
Since $\bar{A}$ is a constant vector, $\bar{A}$ can be expressed in either system

$$\bar{A} = A_1 \hat{i}_1 + A_2 \hat{i}_2 + A_3 \hat{i}_3 = B_1 \hat{j}_1 + B_2 \hat{j}_2 + B_3 \hat{j}_3$$

Transformation matrices are given for both direct transformations from the initial coordinate system to the resulting coordinate system, and for inverse transformations from the resulting coordinate system to the initial coordinate system.

These latter transformations are obtained by using the inverse of the direct transformation matrix. Since these coordinate systems are orthogonal, the inverse matrix is equal to the transpose matrix of the direct transformation matrix.

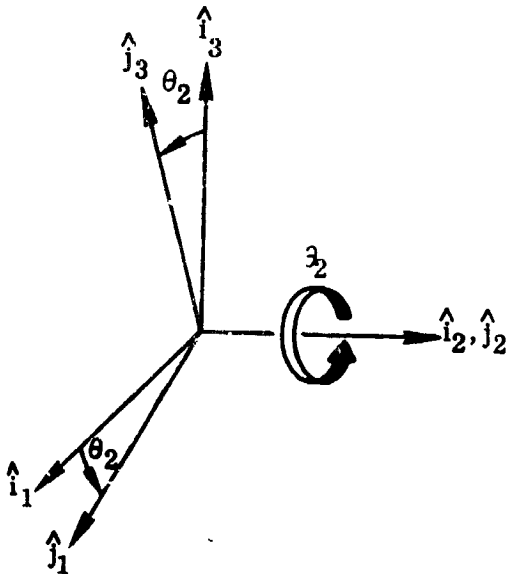
For a rotation θ_1 about $\hat{i}_1$,



$$\left\{ \begin{array}{l} B_1 \\ B_2 \\ B_3 \end{array} \right\} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_1 & \sin \theta_1 \\ 0 & -\sin \theta_1 & \cos \theta_1 \end{pmatrix} \left\{ \begin{array}{l} A_1 \\ A_2 \\ A_3 \end{array} \right\} \quad (A-M1)$$

$$\left\{ \begin{array}{l} A_1 \\ A_2 \\ A_3 \end{array} \right\} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_1 & -\sin \theta_1 \\ 0 & \sin \theta_1 & \cos \theta_1 \end{pmatrix} \left\{ \begin{array}{l} B_1 \\ B_2 \\ B_3 \end{array} \right\}$$

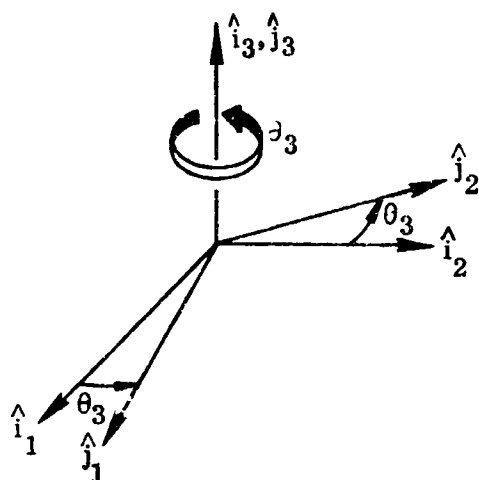
For a rotation θ_2 about $\hat{i}_2$,



$$\left\{ \begin{array}{l} B_1 \\ B_2 \\ B_3 \end{array} \right\} = \begin{pmatrix} \cos \theta_2 & 0 & -\sin \theta_2 \\ 0 & 1 & 0 \\ \sin \theta_2 & 0 & \cos \theta_2 \end{pmatrix} \left\{ \begin{array}{l} A_1 \\ A_2 \\ A_3 \end{array} \right\} \quad (A-M2)$$

$$\left\{ \begin{array}{l} A_1 \\ A_2 \\ A_3 \end{array} \right\} = \begin{pmatrix} \cos \theta_2 & 0 & \sin \theta_2 \\ 0 & 1 & 0 \\ -\sin \theta_2 & 0 & \cos \theta_2 \end{pmatrix} \left\{ \begin{array}{l} B_1 \\ B_2 \\ B_3 \end{array} \right\}$$

For a rotation θ_3 about $\hat{i}_3$,



$$\left. \begin{aligned} \begin{pmatrix} B_1 \\ B_2 \\ B_3 \end{pmatrix} &= \begin{pmatrix} \cos \theta_3 & \sin \theta_3 & 0 \\ -\sin \theta_3 & \cos \theta_3 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \\ A_3 \end{pmatrix} \\ \begin{pmatrix} A_1 \\ A_2 \\ A_3 \end{pmatrix} &= \begin{pmatrix} \cos \theta_3 & -\sin \theta_3 & 0 \\ \sin \theta_3 & \cos \theta_3 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} B_1 \\ B_2 \\ B_3 \end{pmatrix} \end{aligned} \right\} \text{(A-M3)}$$

APPENDIX II

GTSM SUBPROGRAM PRINCIPAL PARAMETERS
AND SUBSCRIPTED ARRAYS

In GTSM, the principal parameters which are required in several subroutines and/or for output are grouped in subscripted arrays according to their usage. These principal arrays are defined in this Appendix; they are the V array, the VQ array, the Q array, the Z array, the W array, and the WQ array. With the exception of the W and WQ arrays, the parameters of these arrays are either defined initially by the input acronyms described in Section 4 or are defined during the simulation as one of the output parameters which are described in Section 5.1.2. For this reason, only the W and WQ arrays are listed with detailed definitions.

APPENDIX II-1 THE V ARRAY

The V array is used to define the trajectory initial conditions. Except for those V array parameters which are defined from the VQ array, all V parameters are defined initially from the input acronyms either by card input or from the internally compiled acronym values. The general rule for V parameter definition is: V parameters which specify quantities are initially set equal to the value of the corresponding input acronym while V parameters used for flags and/or integral values assume the value of the corresponding input acronym plus (minus) 0.1 if the acronym value is positive (negative). This increment (decrement) is used to avoid truncation errors which could cause the immediately lower (higher) integral value to be assumed when converting to a fixed point number.

V parameters with the exception of those defined from the VQ array, can only be subsequently redefined by the special initializing options provided by subroutine **TRAJA**, **TRAJC**, or **TAMPER** or as iteration control variables in a GIS block. The following list defines the V parameter by listing its associated input acronym or VQ parameter and the report section in which the acronym is defined. The "M" subscript for VQ parameters denotes the GIS iteration block number.

<u>V Parameter</u>	<u>Associated Input Acronym or VQ Parameter</u>	<u>Report Section Number</u>
V(1)	RAD	4.2.3.1
V(2)	VEL	
V(3)	GAM	
V(4)	AZM	
V(5)	LAT	
V(6)	LNG	
V(7)	WGH	
V(8)	ALT0	
V(9)	SEC	
V(10)	PLD	
V(11)	VQ(1, M)	Appendix II-2
V(12)	VQ(2, M)	
V(13)	VQ(3, M)	
V(14)	VQ(4, M)	
V(15)	VQ(5, M)	
V(16)	VQ(6, M)	
V(17)	VQ(7, M)	
V(18)	VQ(8, M)	
V(19)	VQ(9, M)	
V(20)	VQ(10, M)	
V(21)	VQ(11, M)	
V(22)	VQ(12, M)	
V(23)	VQ(13, M)	
V(24)	VQ(14, M)	Appendix II-2

<u>V Parameter</u>	<u>Associated Input Acronym or VQ Parameter</u>	<u>Report Section Number</u>
V(25)	VQ(15, M)	Appendix II-2
V(26)	VQ(16, M)	
V(27)	VQ(17, M)	
V(28)	VQ(18, M)	
V(29)	VQ(19, M)	
V(30)	VQ(20, M)	
V(31)	VQ(21, M)	
V(32)	VQ(22, M)	
V(33)	VQ(23, M)	
V(34)	VQ(24, M)	Appendix II-2
V(35)	G0	
V(36)	CK	4.2.3.1
V(37)	WE	
V(38)	D2	
V(39)	D3	
V(40)	D4	
V(41)	AGQ	
V(42)	ATQ	
V(43)	P0	
V(44)	CNV1	
V(45)	CNV2	
V(46)	CNV3	
V(47)	ALTQ	
V(48)	STPF	
V(49)	A	
V(50)	B	
V(51)	T0	
V(52)	VQ(25, M)	Appendix II-2
V(53)	VQ(26, M)	Appendix II-2
V(54)	VQ(27, M)	Appendix II-2
V(55)	VQ(28, M)	Appendix II-2
V(56)	ZOUT	4.2.3.1
V(57)	VQ(29, M)	Appendix II-2
V(58)	VQ(30, M)	Appendix II-2
V(59)	VQ(31, M)	Appendix II-2
V(60)	PSI	
V(61)	SIG	4.2.3.1
V(62)	ALP	
V(63)	LAM	
V(64)	MULT	
V(65)	VELQ	
V(66)	PRNT	

<u>V Parameter</u>	<u>Associated Input Acronym or VQ Parameter</u>	<u>Report Section Number</u>
V(67)	PSIF	4.2.3.1
V(68)	VQ(32, M)	Appendix II-2
V(69)	VQ(33, M)	Appendix II-2
V(70)	VQ(34, M)	Appendix II-2
V(71)	ALTF	4.2.3.1
V(72)	AZMF	
V(73)	TWE	
V(74)	RFCN	
V(75)	AZMI	
V(76)	GTIP	
V(77)	VGAI	
V(78)	YOUT	
V(79)	CNV4	
V(80)	VQ(35, M)	Appendix II-2

APPENDIX II-2 THE VQ ARRAY

The VQ array is used to define the GIS blocks and any required targeting. All the parameters in the VQ array are defined directly from the input acronyms by either card input or from the internally compiled acronym values. As in the case of the V parameters, the VQ parameters are set equal to the value of the corresponding input acronym if the parameter specifies a quantity; if the parameter is a flag or defines an integer, the VQ parameter is set equal to the value of the input acronym plus (minus) 0.1 if the acronym value is positive (negative). The VQ parameters can be subsequently redefined only as part of the targeting - iteration option defined by the input parameter "CRGF(M)" (see "CRGF(M)" in Sec. 4.2.3.3. Although mechanically any VQ parameter can be used as an iteration control variable by specifying the code number of its corresponding V parameter, this would result in a meaningless iteration definition; consequently, no VQ parameter should be used as an iteration control variable. The VQ parameters define the V parameters for each GIS block M. The following list presents the VQ parameters with their corresponding input acronyms and V parameters. In addition, the report section numbers in which the input acronyms are defined are listed.

<u>VQ Parameter</u>	<u>Associated Input Acronym</u>	<u>Associated V Parameter</u>	<u>Report Section Number</u>
VQ(1, M)	CVC1(M)	V(11)	4.2.3.2 ↓
VQ(2, M)	CVC2(M)	V(12)	
VQ(3, M)	CVC3(M)	V(13)	
VQ(4, M)	CVN1(M)	V(14)	
VQ(5, M)	CVN2(M)	V(15)	
VQ(6, M)	CVN3(M)	V(16)	
VQ(7, M)	ECC1(M)	V(17)	
VQ(8, M)	ECC2(M)	V(18)	
VQ(9, M)	ECC3(M)	V(19)	
VQ(10, M)	ECN1(M)	V(20)	
VQ(11, M)	ECN2(M)	V(21)	
VQ(12, M)	ECN3(M)	V(22)	
VQ(13, M)	DCV1(M)	V(23)	
VQ(14, M)	DCV2(M)	V(24)	
VQ(15, M)	DCV3(M)	V(25)	
VQ(16, M)	CVM1(M)	V(26)	
VQ(17, M)	CVM2(M)	V(27)	
VQ(18, M)	CVM3(M)	V(28)	
VQ(19, M)	CVL1(M)	V(29)	
VQ(20, M)	CVL2(M)	V(30)	
VQ(21, M)	CVL3(M)	V(31)	
VQ(22, M)	ECT1(M)	V(32)	
VQ(23, M)	ECT2(M)	V(33)	

<u>VQ Parameter</u>	<u>Associated Input Acronym</u>	<u>Associated V Parameter</u>	<u>Report Section Number</u>
VQ(24, M)	ECT3(M)	V(34)	4.2.3.2
VQ(25, M)	EC1(M)	V(52)	
VQ(26, M)	EC2(M)	V(53)	
VQ(27, M)	EC3(M)	V(54)	
VQ(28, M)	GIS(M)	V(55)	
VQ(29, M)	ITT1(M)	V(57)	
VQ(30, M)	ITT2(M)	V(58)	
VQ(31, M)	ITT3(M)	V(59)	
VQ(32, M)	CRGF(M)	V(68)	
VQ(33, M)	TLAT(M)	V(69)	
VQ(34, M)	TLNG(M)	V(70)	
VQ(35, M)	RDALT(M)	V(80)	4.2.3.3
			

APPENDIX II-3 THE Q ARRAY

The Q array is used to define the trajectory simulation sections. All Q array parameters are initially defined from the input acronyms and can only be redefined during an iteration block when specified as an iteration control variable. Similarly to the V and VQ arrays, the Q parameters are set equal to the value of the corresponding input acronym if the acronym specifies a quantity. If the acronym defines a flag or an integer, the VQ parameter is set equal to the value of the input acronym plus (minus) 0.1 if the acronym value is positive (negative) in order to avoid truncation difficulties when converting to fixed point values.

The Q array is double-subscripted; the first subscript defines the meaning of the parameter while the second subscript, K, defines the corresponding simulation section. The Q parameters together with their corresponding input acronyms and the report section number in which the acronyms are defined are presented in the following list.

<u>Q Parameter</u>	<u>Associated Input Acronym</u>	<u>Report Section Number</u>
Q(1, K)	TWD11(K)	4.2.3.6 ↓
Q(2, K)	TWD12(K)	
Q(3, K)	TWD13(K)	
Q(4, K)	AREF1(K)	
Q(5, K)	AREF2(K)	
Q(6, K)	AREF3(K)	
Q(7, K)	STEP(K)	4.2.3.4 ↓
Q(8, K)	TOL1(K)	
Q(9, K)	TOL2(K)	
Q(10, K)	TOL3(K)	
Q(11, K)	TOL4(K)	
Q(12, K)	TOL5(K)	
Q(13, K)	TOL6(K)	
Q(14, K)	TOL7(K)	
Q(15, K)	HTL1(K)	
Q(16, K)	HTL2(K)	
Q(17, K)	HTL3(K)	
Q(18, K)	HTL4(K)	
Q(19, K)	HTL5(K)	
Q(20, K)	HTL6(K)	
Q(21, K)	HTL7(K)	
Q(22, K)	STGC(K)	4.2.3.5 ↓
Q(23, K)	STGV(K)	
Q(24, K)	STGT(K)	4.2.3.6 ↓
Q(25, K)	STGD(K)	
Q(26, K)	JETW(K)	
Q(27, K)	ATMC(K)	
Q(28, K)	AC1(K)	↓

<u>Q Parameter</u>	<u>Associated Input Acronym</u>	<u>Report Section Number</u>
Q(29, K)	AC2(K)	4.2.3.6
Q(30, K)	AC3(K)	
Q(31, K)	TWC(K)	
Q(32, K)	SIGC(K)	
Q(33, K)	ALPC(K)	
Q(34, K)	LAMC(K)	
Q(35, K)	PSIDT(K)	
Q(36, K)	SIGDT(K)	
Q(37, K)	ALPDT(K)	
Q(38, K)	LAMDT(K)	
Q(39, K)	HMIN(K)	4.2.3.4
Q(40, K)	HCOEF(K)	
Q(41, K)	BCKT(K)	4.2.3.5
Q(42, K)	BCKTT(K)	
Q(43, K)	SPSI(K)	4.2.3.6
Q(44, K)	SSIG(K)	
Q(45, K)	SALP(K)	
Q(46, K)	SLAM(K)	
Q(47, K)	TWD1(K)	
Q(48, K)	TWD2(K)	
Q(49, K)	TWD3(K)	
Q(50, K)	TWD4(K)	
Q(51, K)	TWD10(K)	
Q(52, K)	GDOT(K)	
Q(53, K)	XOUT(K)	4.2.3.5
Q(54, K)	TOL8(K)	4.2.3.4
Q(55, K)	HTL8(K)	
Q(56, K)	ORB(K)	4.2.3.5
Q(57, K)	GMC1(K)	4.2.3.6
Q(58, K)	GMC2(K)	
Q(59, K)	GMC3(K)	
Q(60, K)	GMC4(K)	
Q(61, K)	TWD6(K)	
Q(62, K)	PITX(K)	
Q(63, K)	PSEC(K)	
Q(64, K)	PTME(K)	
Q(65, K)	PRIF(K)	4.2.3.5
Q(66, K)	TWD14(K)	4.2.3.6
Q(67, K)	VSET(K)	4.2.3.5
Q(68, K)	VINC(K)	
Q(69, K)	WOUT(K)	
Q(70, K)	HMAX(K)	4.2.3.4
Q(71, K)	HMODE(K)	
Q(72, K)	RCOEF(K)	

APPENDIX II-4 THE Z ARRAY

The Z array consists of the principal parameters which are computed during the trajectory at least once during each integration step if they are specified. These parameters can all be used for section termination, for iteration end conditions, and with the exception of Z(28), Z(75), Z(90), Z(92), Z(101), Z(102) and Z(103) they comprise the trajectory output. These Z parameters, together with their corresponding output code and the report section in which they are defined are presented in the following list.

<u>Z Parameter</u>	<u>Output Parameter Code Number</u>	<u>Report Section Number</u>
Z(1)	1	5.1.2.1
Z(2)	2	
Z(3)	3	
Z(4)	4	
Z(5)	5	
Z(6)	6	
Z(7)	7	
Z(8)	8	
Z(9)	9	
Z(10)	10	
Z(11)	11	
Z(12)	12	
Z(13)	13	
Z(14)	14	
Z(15)	15	
Z(16)	16	
Z(17)	17	
Z(18)	18	
Z(19)	19	
Z(20)	20	
Z(21)	21	
Z(22)	22	
Z(23)	23	
Z(24)	24	
Z(25)	25	
Z(26)	26	
Z(27)	27	
Z(28)	28*	2.2.2.8

* This parameter is not output, however, it can be used as a simulation section termination parameter or as an iteration end condition by specifying this code number.

<u>Z Parameter</u>	<u>Output Parameter Code Number</u>	<u>Report Section Number</u>
Z(29)	29	5.1.2.1
Z(30)	30	
Z(31)	31	
Z(32)	32	
Z(33)	33	
Z(34)	34	
Z(35)	35	
Z(36)	36	
Z(37)	37	
Z(38)	38	
Z(39)	39	
Z(40)	40	
Z(41)	41	
Z(42)	42	
Z(43)	43	
Z(44)	44	
Z(45)	45	
Z(46)	46	
Z(47)	47	
Z(48)	48	
Z(49)	49	
Z(50)	50	
Z(51)	51	
Z(52)	52	5.1.2.2
Z(53)	53	
Z(54)	54	
Z(55)	55	
Z(56)	56	
Z(57)	57	
Z(58)	58	
Z(59)	59	
Z(60)	60	
Z(61)	61	
Z(62)	62	
Z(63)	63	
Z(64)	64	
Z(65)	65	5.1.2.1
Z(66)	66	5.1.2.2
Z(67)	67	

<u>Z Parameter</u>	<u>Output Parameter Code Number</u>	<u>Report Section Number</u>
Z(68)	68	5.1.2.2
Z(69)	69	
Z(70)	70	
Z(71)	71	
Z(72)	72	
Z(73)	73	
Z(74)	74	
Z(75)	75*	
Z(76)	76	
Z(77)	77	
Z(78)	78	5.1.2.1
Z(79)	79	5.1.2.2
Z(80)	80	
Z(81)	81	
Z(82)	82	
Z(83)	83	
Z(84)	84	
Z(85)	85	
Z(86)	86	
Z(87)	87	
Z(88)	88	
Z(89)	89	
Z(90)	90*	2.2.2.10
Z(91)	91	5.1.2.2
Z(92)	92*	2.2.2.8
Z(93)	93	5.1.2.2
Z(94)	94	
Z(95)	95	
Z(96)	96	
Z(97)	97	
Z(98)	98	
Z(99)	99	
Z(100)	100	
Z(101)	101*	
Z(102)	102*	2.2.2.10
Z(103)	103*	

*This parameter is not output, however, it can be used as a simulation section termination parameter or as an iteration end condition by specifying this code number.

APPENDIX II-5 THE W ARRAY

The W array consists of program flags and certain minor parameters which are evaluated once to several times (not as frequently as each integration step, however) during a trajectory simulation. The purpose of the W array is to reduce redundancy while making these parameters available to each subroutine by means of the blank COMMON statement. As in the case of the V array, certain elements of the W array are redefined for each GIS block. These particular W parameters are directly defined from the WQ array for the Mth GIS block. The W array parameters together with their definitions are presented in the following list.

<u>W Parameter</u>	<u>Definition</u>
W(1)	The ATMOS subroutine initialization flag. $W(1) \leq 0.5$ indicates the initial call to subroutine ATMOS during a simulation section. $W(1) > 0.5$ indicates calls to subroutine ATMOS subsequent to the initial call during a simulation section.
W(2)	The AERO1 subroutine initialization flag. * $W(2) \leq 0.5$ indicates the initial call to subroutine AERO1 during a simulation section. $W(2) > 0.5$ indicates calls to subroutine AERO1 subsequent to the initial call during a simulation section.
W(3)	The AERO2 subroutine initialization flag. $W(3) \leq 0.5$ indicates the initial call to subroutine AERO2 during a simulation section. $W(3) > 0.5$ indicates calls to subroutine AERO2 subsequent to the initial call during a simulation section.
W(4)	The AERO3 subroutine initialization flag. $W(4) \leq 0.5$ indicates the initial call to subroutine AERO3 during a simulation section. $W(4) > 0.5$ indicates calls to subroutine AERO3 subsequent to the initial call during a simulation section.

*Not used in the current version SSSP program.

W ParameterDefinition

W(5)	The PRPSN subroutine initialization flag. $W(5) \leq 0.5$ indicates the initial call to subroutine PRPSN during a simulation section. $0.5 < W(5) \leq 1.5$ indicates calls to subroutine PRPSN subsequent to the initial call during a simulation section whenever only a SIMPO propulsion model is used. $W(5) > 1.5$ indicates calls to subroutine PRPSN subsequent to the initial call during a simulation section whenever a propulsion table model is used.
W(6)	The ANGLE subroutine initialization flag. $W(6) \leq 0.5$ indicates the initial call to subroutine ANGLE during a simulation section. $W(6) > 0.5$ indicates calls to subroutine ANGLE subsequent to the initial call during a simulation section.
W(7)	The iteration print suppression flag. $W(7) \leq 0.5$ indicates iteration trajectory print according to the value of PRNT (Section 4.1.2). $W(7) > 0.5$ indicates complete trajectory print (Section 4.1.2).
W(8)	Denotes the number of iteration control variable groups or the number of end conditions for a GIS block.
W(9)	Denotes the total number of calls to subroutine TRAJB during a GIS block or from subroutine ITRTN.
W(10)	Denotes the number of calls to subroutine TRAJB during a step of subroutine XNEW1, program GTSM.
W(11)	Not used in the current version SSSP program.
W(12)	Not used in the current version SSSP program.
W(13)	$WQ(1, M)$ for the Mth GIS block (see Appendix II-6).

W ParameterDefinition

W(14)	WQ(2, M) for the Mth GIS block (See Appendix II-6).
W(15)	WQ(3, M) for the Mth GIS block (See Appendix II-6).
W(16)	WQ(4, M) for the Mth GIS block (See Appendix II-6).
W(17)	WQ(5, M) for the Mth GIS block (See Appendix II-6).
W(18)	WQ(6, M) for the Mth GIS block (See Appendix II-6).
W(19)	WQ(7, M) for the Mth GIS block (See Appendix II-6).
W(20)	$\sin(\theta_L) = \sin(\text{ARG})$ where $\text{ARG} = V(5)/V(45)$.
W(21)	$\cos(\theta_L) = \cos(\text{ARG})$ where $\text{ARG} = V(5)/V(45)$.
W(22)	$\sin(\phi_L) = \sin(\text{ARG})$ where $\text{ARG} = V(6)/V(45)$.
W(23)	$\cos(\phi_L) = \cos(\text{ARG})$ where $\text{ARG} = V(6)/V(45)$.
W(24)	WQ(8, M) for the Mth GIS block (See Appendix II-6).
W(25)	$\sin(\epsilon) = \sin(\text{ARG})$ where $\text{ARG} = (90. - V(5) - V(76))/V(45)$.
W(26)	$\cos(\epsilon) = \cos(\text{ARG})$ where $\text{ARG} = (90. - V(5) - V(76))/V(45)$.
W(27)	$\sin(D) = \sin(\text{ARG})$ where $\text{ARG} = (180. - V(75))/V(45)$.
W(28)	$\cos(D) = \cos(\text{ARG})$ where $\text{ARG} = (180. - V(75))/V(45)$.
W(29)	The $\hat{J}_1 - \hat{J}_2 - \hat{J}_3$ Inertial Coordinate System orientation flag. W(29) ≤ 900 . denotes that $V(75) = \text{AZMI}$ (the input value). W(29) > 900 . denotes that $V(75) = V(4)$ (subject to change when $V(4)$ is an iteration control variable).
W(30)	The iteration error flag. W(30) ≤ 0.5 indicates that there are no iteration problems. W(30) > 0.5 indicates that either the maximum number of iterations have been exceeded, or a predicted control variable value was outside the control variable bounds, or that the iteration end conditions are invariant to the selected iteration control variables.

W ParameterDefinition

W(31)	The subroutine ANGLE mode flag. $W(31) < 0.5$ indicates normal operation of subroutine ANGLE $W(31) \geq 0.5$ indicates that the current time argument for the pitch rate table is outside the table time argument bounds and that a specified pitch angle of attack is used rather than a relative pitch rate from the table.
W(32)	The table look-up error flag. $W(32) \leq 0.5$ indicates normal error free operation of all the table look-up procedures which are specified from subroutines ANGLE, ATMOS, AERO1, AERO2, AERO3, and PRPSN. $W(32) > 0.5$ indicates a table look-up error in one of the specified subroutines using a table look-up procedure (ANGLE, ATMOS, AERO2, AERO3, and PRPSN).
W(33)	Not used.
W(34)	$WQ(9, M)$ for the Mth GIS block (See Appendix II-6).
W(35)	$WQ(10, M)$ for the Mth GIS block (See Appendix II-6).
W(36)	GTSM error flag. $W(36) \leq 0.5$ indicates no error in GTSM $W(36) > 0.5$ indicates an error in GTSM
W(37)	GTSM multiple run flag $W(37) \leq 0.5$ No multiple case follows the current case $W(37) > 0.5$ An additional case follows the current case

APPENDIX II-6 THE WQ ARRAY

The WQ array consists of certain minor parameters which are evaluated once to several times (not as frequently as each integration step, however) during a GIS block. The WQ array parameters are used to define or redefine the corresponding W parameter during each GIS block M. The WQ array parameters together with their corresponding W parameters and definitions are presented in the following list.

<u>WQ Parameter</u>	<u>W Parameter</u>	<u>Definition</u>
WQ(1, M)	W(13)	$\sin(\theta_T) = \sin(\text{ARG})$ where $\text{ARG} = V(69)/V(45)$.
WQ(2, M)	W(14)	$\cos(\theta_T) = \cos(\text{ARG})$ where $\text{ARG} = V(69)/V(45)$.
WQ(3, M)	W(15)	$\sin(\phi_T) = \sin(\text{ARG})$ where $\text{ARG} = V(70)/V(45)$.
WQ(4, M)	W(16)	$\cos(\phi_T) = \cos(\text{ARG})$ where $\text{ARG} = V(70)/V(45)$.
WQ(5, M)	W(17)	$\cos(\text{DR}_T) = \cos(\text{ARG})$ where $\text{ARG} = W(19)/V(45)$.
WQ(6, M)	W(18)	$\sin(\text{DR}_T) = \sin(\text{ARG})$ where $\text{ARG} = W(19)/V(45)$.
WQ(7, M)	W(19)	The target downrange central angle DR_T which is defined in Section 2.2.2.9
WQ(8, M)	W(24)	The relative azimuth of plane S_1 at θ_L and ϕ_L which is defined by Eq. (2-108) in Section 2.2.2.9 such that $0^\circ \leq WQ(8, M) < 180^\circ$.
WQ(9, M)	W(34)	The relative azimuth of plane S_1 at θ_L and ϕ_L which is defined by Eq. (2-108) in Section 2.2.2.9 and logic similar to that of Eq. (2-107) which results in $0^\circ \leq WQ(9, M) < 360^\circ$.
WQ(10, M)	W(35)	The radius magnitude of the current target radar site which is defined by Eq. (2-184).

APPENDIX III

COMPILED IN ATMOSPHERE TABLE

The Cape Kennedy Reference Atmosphere (CKRA) is internally stored and consequently the CKRA is the atmosphere model which is assumed unless another atmosphere model is input via \$DATA1. This appendix presents a listing of the CKRA atmosphere in terms of the appropriate required \$DATA1 input parameters.

XKA0= 193.,
 P0=14.75055,
 ALT= -30000000.,

0.	1.0000E+03,	2.0000E+03,	3.0000E+03,	4.0000E+03,
5.0000E+03,	6.0000E+03,	7.0000E+03,	8.0000E+03,	9.0000E+03,
1.0000E+04,	1.1000E+04,	1.2000E+04,	1.3000E+04,	1.4000E+04,
1.5000E+04,	1.6000E+04,	1.7000E+04,	1.8000E+04,	1.9000E+04,
2.0000E+04,	2.1000E+04,	2.2000E+04,	2.3000E+04,	2.4000E+04,
2.5000E+04,	2.6000E+04,	2.7000E+04,	2.8000E+04,	2.9000E+04,
3.0000E+04,	3.1000E+04,	3.2000E+04,	3.3000E+04,	3.4000E+04,
3.5000E+04,	3.6000E+04,	3.7000E+04,	3.8000E+04,	3.9000E+04,
4.0000E+04,	4.1000E+04,	4.2000E+04,	4.3000E+04,	4.4000E+04,
4.5000E+04,	4.6000E+04,	4.7000E+04,	4.8000E+04,	4.9000E+04,
5.0000E+04,	5.1000E+04,	5.2000E+04,	5.3000E+04,	5.4000E+04,
5.5000E+04,	5.6000E+04,	5.7000E+04,	5.8000E+04,	5.9000E+04,
6.0000E+04,	6.2000E+04,	6.4000E+04,	6.6000E+04,	6.8000E+04,
7.0000E+04,	7.2000E+04,	7.4000E+04,	7.6000E+04,	7.8000E+04,
8.0000E+04,	8.2000E+04,	8.4000E+04,	8.6000E+04,	8.8000E+04,
9.0000E+04,	9.2000E+04,	9.4000E+04,	9.6000E+04,	9.8000E+04,
1.0000E+05,	1.0200E+05,	1.0400E+05,	1.0600E+05,	1.0800E+05,
1.1000E+05,	1.1200E+05,	1.1400E+05,	1.1600E+05,	1.1800E+05,
1.2000E+05,	1.2300E+05,	1.2600E+05,	1.2900E+05,	1.3200E+05,
1.3500E+05,	1.3800E+05,	1.4100E+05,	1.4400E+05,	1.4700E+05,
1.5000E+05,	1.5300E+05,	1.5600E+05,	1.5900E+05,	1.6200E+05,
1.6500E+05,	1.6800E+05,	1.7100E+05,	1.7400E+05,	1.7700E+05,
1.8000E+05,	1.8500E+05,	1.9000E+05,	1.9500E+05,	2.0000E+05,
2.0500E+05,	2.1000E+05,	2.1500E+05,	2.2000E+05,	2.2500E+05,
2.3000E+05,	2.3500E+05,	2.4000E+05,	2.4500E+05,	2.5000E+05,
2.5500E+05,	2.6000E+05,	2.6500E+05,	2.7000E+05,	2.7500E+05,
2.8000E+05,	2.8500E+05,	2.9000E+05,	2.9500E+05,	3.0000E+05,
3.1000E+05,	3.2000E+05,	3.3000E+05,	3.4000E+05,	3.5000E+05,
3.6000E+05,	3.7000E+05,	3.8000E+05,	3.9000E+05,	4.0000E+05,
4.1000E+05,	4.2000E+05,	4.3000E+05,	4.4000E+05,	4.5000E+05,
4.6000E+05,	4.7000E+05,	4.8000E+05,	4.9000E+05,	5.0000E+05,
5.5000E+05,	6.0000E+05,	6.5000E+05,	7.0000E+05,	7.5000E+05,
8.0000E+05,	8.5000E+05,	9.0000E+05,	9.5000E+05,	1.0000E+06,
1.0500E+06,	1.1000E+06,	1.1500E+06,	1.2000E+06,	1.2500E+06,
1.3000E+06,	1.3500E+06,	1.4000E+06,	1.4500E+06,	1.5000E+06,
1.5500E+06,	1.6000E+06,	1.6500E+06,	1.7000E+06,	1.7500E+06,
1.8000E+06,	1.8500E+06,	1.9000E+06,	1.9500E+06,	2.0000E+06,
2.0500E+06,	2.1000E+06,	2.1500E+06,	2.2000E+06,	2.2500E+06,
2300000.,	5000000.,			

PRS= 14.75055,

1.4751E+01,	1.4244E+01,	1.3751E+01,	1.3273E+01,	1.2808E+01,
1.2357E+01,	1.1919E+01,	1.1494E+01,	1.1082E+01,	1.0682E+01,
1.0295E+01,	9.9191E+00,	9.5550E+00,	9.2022E+00,	8.8602E+00,
8.5288E+00,	8.2076E+00,	7.8965E+00,	7.5950E+00,	7.3029E+00,
7.0200E+00,	6.7459E+00,	6.4804E+00,	6.2233E+00,	5.9744E+00,
5.7333E+00,	5.5000E+00,	5.2741E+00,	5.0556E+00,	4.8442E+00,
4.6397E+00,	4.4421E+00,	4.2510E+00,	4.0664E+00,	3.8882E+00,
3.7161E+00,	3.5496E+00,	3.3893E+00,	3.2349E+00,	3.0862E+00,
2.9432E+00,	2.8057E+00,	2.6737E+00,	2.5469E+00,	2.4252E+00,
2.3086E+00,	2.1970E+00,	2.0901E+00,	1.9878E+00,	1.8901E+00,
1.7969E+00,	1.7079E+00,	1.6230E+00,	1.5422E+00,	1.4653E+00,
1.3921E+00,	1.3226E+00,	1.2567E+00,	1.1941E+00,	1.1359E+00,
1.0801E+00,	9.7726E-01,	8.8474E-01,	8.0151E-01,	7.2660E-01,
6.5912E-01,	5.9832E-01,	5.4349E-01,	4.9402E-01,	4.4935E-01,
4.0899E-01,	3.7249E-01,	3.3945E-01,	3.0952E-01,	2.8238E-01,
2.5774E-01,	2.3534E-01,	2.1492E-01,	1.9638E-01,	1.7954E-01,

1.6424E-01,	1.5033E-01,	1.3767E-01,	1.2614E-01,	1.1564E-01,
1.0607E-01,	9.7349E-02,	8.9389E-02,	8.2123E-02,	7.5488E-02,
6.9427E-02,	6.1296E-02,	5.4182E-02,	4.7950E-02,	4.2486E-02,
3.7687E-02,	3.3468E-02,	2.9753E-02,	2.6477E-02,	2.3583E-02,
2.1022E-02,	1.8752E-02,	1.6736E-02,	1.4941E-02,	1.3342E-02,
1.1915E-02,	1.0635E-02,	9.4846E-03,	8.4520E-03,	7.5254E-03,
6.6941E-03,	5.4953E-03,	4.4977E-03,	3.6693E-03,	2.9829E-03,
2.4160E-03,	1.9491E-03,	1.5660E-03,	1.2528E-03,	9.9776E-04,
7.9096E-04,	6.2403E-04,	4.8991E-04,	3.8268E-04,	2.9739E-04,
2.2989E-04,	1.7677E-04,	1.3519E-04,	1.0282E-04,	7.7776E-05,
5.8766E-05,	4.4409E-05,	3.3564E-05,	2.5370E-05,	1.9239E-05,
1.1291E-05,	6.7945E-06,	4.1841E-06,	2.6533E-06,	1.7334E-06,
1.1626E-06,	8.0470E-07,	5.7813E-07,	4.2822E-07,	3.2670E-07,
2.5872E-07,	2.1098E-07,	1.7601E-07,	1.4954E-07,	1.2897E-07,
1.1263E-07,	9.9395E-08,	8.8517E-08,	7.9451E-08,	7.1756E-08,
4.5742E-08,	3.0813E-08,	2.1430E-08,	1.5233E-08,	1.1035E-08,
8.1197E-09,	6.0518E-09,	4.5646E-09,	3.4813E-09,	2.6823E-09,
2.0842E-09,	1.6320E-09,	1.2873E-09,	1.0226E-09,	8.1773E-10,
6.5807E-10,	5.3254E-10,	4.3298E-10,	3.5364E-10,	2.9013E-10,
2.3903E-10,	1.9775E-10,	1.6425E-10,	1.3683E-10,	1.1430E-10,
9.5740E-11,	8.0400E-11,	6.7891E-11,	5.7134E-11,	4.8334E-11,
4.0963E-11,	3.4778E-11,	2.9578E-11,	2.5200E-11,	2.1506E-11,
0.,	0.,			

VLS= 1137.990,

1.1380E+03,	1.1337E+03,	1.1296E+03,	1.1258E+03,	1.1221E+03,
1.1186E+03,	1.1152E+03,	1.1119E+03,	1.1085E+03,	1.1052E+03,
1.1019E+03,	1.0986E+03,	1.0952E+03,	1.0917E+03,	1.0882E+03,
1.0845E+03,	1.0808E+03,	1.0770E+03,	1.0730E+03,	1.0689E+03,
1.0648E+03,	1.0605E+03,	1.0561E+03,	1.0516E+03,	1.0470E+03,
1.0424E+03,	1.0376E+03,	1.0328E+03,	1.0280E+03,	1.0231E+03,
1.0181E+03,	1.0132E+03,	1.0082E+03,	1.0033E+03,	9.9841E+02,
9.9355E+02,	9.8847E+02,	9.8370E+02,	9.7905E+02,	9.7455E+02,
9.7021E+02,	9.6605E+02,	9.6210E+02,	9.5837E+02,	9.5489E+02,
9.5167E+02,	9.4872E+02,	9.4607E+02,	9.4372E+02,	9.4168E+02,
9.3998E+02,	9.3860E+02,	9.3757E+02,	9.3688E+02,	9.3653E+02,
9.3653E+02,	9.3686E+02,	9.3754E+02,	9.3853E+02,	9.4229E+02,
9.4399E+02,	9.4748E+02,	9.5103E+02,	9.5459E+02,	9.5812E+02,
9.6158E+02,	9.6493E+02,	9.6815E+02,	9.7121E+02,	9.7409E+02,
9.7680E+02,	9.7934E+02,	9.8169E+02,	9.8389E+02,	9.8595E+02,
9.8790E+02,	9.8990E+02,	9.9277E+02,	9.9565E+02,	9.9856E+02,
1.0015E+03,	1.0044E+03,	1.0074E+03,	1.0105E+03,	1.0135E+03,
1.0166E+03,	1.0198E+03,	1.0229E+03,	1.0261E+03,	1.0294E+03,
1.0326E+03,	1.0375E+03,	1.0424E+03,	1.0473E+03,	1.0521E+03,
1.0568E+03,	1.0614E+03,	1.0658E+03,	1.0698E+03,	1.0736E+03,
1.0770E+03,	1.0798E+03,	1.0822E+03,	1.0839E+03,	1.0836E+03,
1.0811E+03,	1.0782E+03,	1.0749E+03,	1.0712E+03,	1.0671E+03,
1.0628E+03,	1.0549E+03,	1.0464E+03,	1.0373E+03,	1.0279E+03,
1.0181E+03,	1.0082E+03,	9.9814E+02,	9.8805E+02,	9.7798E+02,
9.6798E+02,	9.5808E+02,	9.4828E+02,	9.3858E+02,	9.2892E+02,
9.1925E+02,	9.0947E+02,	8.9945E+02,	8.8903E+02,	8.8400E+02,
8.8400E+02,	8.8400E+02,	8.8400E+02,	8.8400E+02,	8.9450E+02,
9.1635E+02,	9.3768E+02,	9.6117E+02,	9.9488E+02,	1.0275E+03,
1.0591E+03,	1.1170E+03,	1.1745E+03,	1.2293E+03,	1.3138E+03,
1.4106E+03,	1.5012E+03,	1.5866E+03,	1.6676E+03,	1.7449E+03,
1.8189E+03,	1.8900E+03,	1.9585E+03,	2.0247E+03,	2.0764E+03,
2.2660E+03,	2.3721E+03,	2.4532E+03,	2.5195E+03,	2.5841E+03,
2.6358E+03,	2.6853E+03,	2.7340E+03,	2.7818E+03,	2.8262E+03,
2.8644E+03,	2.9022E+03,	2.9394E+03,	2.9762E+03,	3.0125E+03,
3.0484E+03,	3.0782E+03,	3.1060E+03,	3.1334E+03,	3.1607E+03,

3.1877E+03, 3.2144E+03, 3.2592E+03, 3.2565E+03, 3.2736E+03,
3.2907E+03, 3.3077E+03, 3.3246E+03, 3.3414E+03, 3.3544E+03,
3.3652E+03, 3.3760E+03, 3.3867E+03, 3.3974E+03, 3.4080E+03,
3417.90, 3417.90,

GDC-DBB70-002

APPENDIX IV
PROGRAM FORTRAN LISTING

PRECEDING PAGE BLANK NOT FILMED

GDC-PRE70-002

Appendix IV-1 Program MAIN

```

C      OVERLAY(TTSS,0,0)
C      PROGRAM MAIN(INPUT,CUTPUT,TAPEK=INPUT,TAPE6=OUTPUT
C      1PUNCH,TAPE8=PUNCH)
      REAL LRODY0,LRODYF
      REAL NFNGS
      REAL KIN
      REAL ISP,K,LF,MR,NCREW,LRODY,NPASS
      REAL NXFCR
      REAL IVACO,IVACB,ISLO,ISLB,IOVEL
      DIMENSION IVACO(7),ISLO(7),TFCTRO(7),TFCTRB(4),IVACB(4),ISLB(4)
1      ,PRNTX(3)
      DIMENSION SKP(30),SCP(300),TB4(6),TB20(10),TB27(6),TB34(6),
1      TB48(10),TB49(10),TB50(10),TB57(6)
      COMMON/CRPINY/
1      TP1,TB2,TB3,TB4,TB5,TB6,TB7,TB8,TB9,TB10,TB11,TP12,TB13,TB14,
2      TB15,TB16,TB17,TP18,TB19,TB20,TB21,TB22,TB23,TB24,TB25,TB26,TB27,
3      TB28,TB29,TB30,TP31,TB32,TB33,TB34,TB35,TB36,TB37,TB38,TB39,TB40,
4      TB41,TB42,TB43,TB44,TB45,TB46,TB47,TB48,TB49,TB50,TB51,TB52,TB53,
5      TB54,TP55,TB56,TB57,SKP,SCP
      COMMON/TAMP/WGRCSC,CTTOT,WFUOXO,TBTO,RTTOT
      COMMON/SUNVW/WOP,          WABFUB,WFUOXB,WDRYR,WGROSB,WOTHR,
1      VFUTKR,VOXTKB,VOTHP,VBODYB,WABFUB,
2      LBODYP,SBODYB,SPLANB,WVRSB,
1      WPAYLO,WDRYO,WOTHO,  WABFUO,
2      VFUTKO,VOXTKO,VCARC,VOTHO,VBODYO,
3      LBODYO,SBODYO,SPLANB,WVRSO
4      ,WOBRTB,WOBRTB,WRTENO,WRTNR,WENTRO,WENTRB,WLANDB,WLANDO
5      ,WCONTC,WCONTB
      COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1      VO(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2      C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3      XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWO7(25,2),TWO8(25,2),SW(50),
4      TWO9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5      PS(15)
      COMMON/HEAD  /TITLES(10)
      COMMON/CINPUT/
1ANENGs  ,ANTANK  ,ASRATO  ,ASWEEP  ,C(300)  ,CBBODY  ,CFUEL(6),
2CHBODY  ,CLBODY  ,CSBODY  ,CSFAIR  ,CSFUTK  ,CSHORZ  ,CSOXTK ,
3CSPLAN  ,CSVERT  ,CSWING  ,CTHRST  ,CTHST2  ,DEF(5)  ,FXMOVS ,
4ISP(6)  ,ITPS    ,K(30)   ,KIN     ,LF      ,MR(6)   ,NCREW  ,
5NENGs   ,NLISTO  ,NFASS  ,NWL     ,PCHAM  ,OX      ,RHOFU  ,
6RHOFU2  ,RHOX    ,RHOX2  ,SBODY   ,TOL     ,TOVFRG  ,TPRATO  ,
7TYTAIL  ,VBODY   ,WGRCSS
      COMMON/VLCAL/BBODY,CROOT ,CSPAN  ,CTIP   ,GAL    ,GSPAN  ,
2HBODY   ,LBODY   ,RTOD   ,SFAIR  ,SFUTK  ,SHORZ  ,SOXTK  ,
3SPLAN   ,STPS(1) ,SVERT  ,SWING  ,SXPOS  ,TOEL   ,TROOT  ,
4TTOT    ,TTOT2   ,TTOTAL ,VBODYA ,VBODY1  ,VBODY2  ,VCARGO ,
5VCREW   ,VFUTK   ,VFUTK2 ,VINSTK ,VLGRAY  ,VOTHER  ,VOXTK  ,
6VOXTK2  ,VPROP   ,VSTRUC
      COMMON/WTALC/  ABFSYS  ,WABFTK  ,WABFU   ,WABFR   ,WACRES  ,
1WACS    ,WACSFO  ,WACSTK  ,WAFRO   ,WAUXT   ,WBASIC  ,WBODY   ,

```

```

2WBUMP ,WCARGO ,WCOMM ,WCCNT ,WCOVER ,WDECCAY ,WDIST1 ,
3WDIST2 ,WDOCK ,WDPLOY ,WDORANS ,WDRY ,WELCAD ,WEMPTY ,
4WENGMT ,WENGSS ,WENGSS2 ,WFAIR ,WFCNT ,WFDCAY ,WFRST ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFOOX ,WFURES ,
6WFUSYS ,WFUTK ,WFUTK2 ,WFLTOT ,WFUTRP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLCSS ,WLRD ,WNACEL ,WODCAY ,
9WOIL ,WOILRS ,WORSUL ,WCOVERS ,WOX(6) ,WOX2(3) ,WOXIN ,
1WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPERS ,WPCWCD ,WPOWER ,WPOWFO ,
3WPOWRS ,WPONTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
4WRESID ,WRESRV ,WSEAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF ,WTABC ,WTHRST ,WTC ,WTPS ,WVERT ,WAIT(10) ,
6WNET ,WING ,WZROFU
COMMON/XCLVG/CLVG
COMMON/XDRNG/DRNG
COMMON/XWPO/WPOREQ,WOREQ,GWREQ
COMMON/DATA2X/ ALD,FBPAR,IDVEL,ISLB,ISLO,IVACB,IVACO,PERISP,QMAX,
1 QMXS,SFC,SLVOLT,COPIES,SYNIT,TFCTRB,TFCTRO,TOLMU,TOLTW,
2 TRATIO,TWLOI,TWLOI,WTOUT,FIRE,BOOTH,VCRUSE
3 , NXFOB,PRNTX,FSEC
4 ,SOLID,AS,BS,SISP,SINERT,SAE,TSBO
6 ,FRFUEL,CA,CB,WFLYX,RT,P1,R3,SFC1,SFC2,SFC3,ALD1,ALD2,ALD3,VFLY1,
7 VFLY2,VFLY3
COMMON/FLY/FLYBCK
COMMON/APV/ALT(200),PRS(200),VLS(200)
DATA CLVG/ 1.0/
DATA DRNG/ 0.0/
DATA WPOREQ,WOREQ,GWREQ / 0., 0., 3500000. /
DATA FLYBCK / 2. /
DATA FRFUEL,CA,CB,WFLYX,RT,P1,R3,SFC1,SFC2,SFC3,ALD1,ALD2,ALD3,
1 VFLY1,VFLY2,VFLY3 / 1., 9*0., 6*1. /
DATA SOLID,AS,BS,SISP,SINERT,SAE,TSBO / 7*0./
DATA NXFOB / 0. /
DATA PRNTX,FSEC / 1., 0., 1., 0. /
DATA IVACO,ISLO,TFCTRO,TFCTRB / 7*450., 7*390., 7*1., 4*1. /
DATA IVACB,ISLB/4*450., 4*390. /
DATA COPIES/6./
DATA TOLMU,SYNIT,TRATIO,PERISP,BOOTH,ALD,SFC,FBPAR,VCRUSE,FIRE
1 / .00050, 5.0, 1.0, 0.810, 0., 6.0, 0.200,250.0,300., 2. /
DATA WTOUT,SLVOLT,TWLOI,TOLTW,TWLOI,QMXS,QMAX
1 / 0., 0., 1.371, 0.801, -1., 900., 550. /
THE CAPE KENNEDY REFERENCE ATMOSPHERE (CKRA)
ALSO KNOWN AS THE 1963 PATRICK REFERENCE ATMOSPHERE
BASED ON MSFC COMP LAB SUBROUTINE PRA-63
REFERENCE TM X-53973 NATURAL ENVIRONMENT CRITERIA FOR THE
SPACE SHUTTLE PROGRAM W W VAUGHAN, 2/9/70
DATA (ALT(I), I= 1, 96) / -30000000.,
1 0., 1.0000E+03, 2.0000E+03, 3.0000E+03, 4.0000E+03,
2 5.0000E+03, 6.0000E+03, 7.0000E+03, 8.0000E+03, 9.0000E+03,
3 1.0000E+04, 1.1000E+04, 1.2000E+04, 1.3000E+04, 1.4000E+04,
4 1.5000E+04, 1.6000E+04, 1.7000E+04, 1.8000E+04, 1.9000E+04,

```

5	2.0000E+04,	2.1000E+04,	2.2000E+04,	2.3000E+04,	2.4000E+04,
6	2.5000E+04,	2.6000E+04,	2.7000E+04,	2.8000E+04,	2.9000E+04,
7	3.0000E+04,	3.1000E+04,	3.2000E+04,	3.3000E+04,	3.4000E+04,
8	3.5000E+04,	3.6000E+04,	3.7000E+04,	3.8000E+04,	3.9000E+04,
9	4.0000E+04,	4.1000E+04,	4.2000E+04,	4.3000E+04,	4.4000E+04,
1	4.5000E+04,	4.6000E+04,	4.7000E+04,	4.8000E+04,	4.9000E+04,
2	5.0000E+04,	5.1000E+04,	5.2000E+04,	5.3000E+04,	5.4000E+04,
3	5.5000E+04,	5.6000E+04,	5.7000E+04,	5.8000E+04,	5.9000E+04,
4	6.0000E+04,	6.2000E+04,	6.4000E+04,	6.6000E+04,	6.8000E+04,
5	7.0000E+04,	7.2000E+04,	7.4000E+04,	7.6000E+04,	7.8000E+04,
6	8.0000E+04,	8.2000E+04,	8.4000E+04,	8.6000E+04,	8.8000E+04,
7	9.0000E+04,	9.2000E+04,	9.4000E+04,	9.6000E+04,	9.8000E+04,
8	1.0000E+05,	1.0200E+05,	1.0400E+05,	1.0600E+05,	1.0800E+05,
9	1.1000E+05,	1.1200E+05,	1.1400E+05,	1.1600E+05,	1.1800E+05,
1	1.2000E+05,	1.2300E+05,	1.2600E+05,	1.2900E+05,	1.3200E+05/

DATA (ALT(I), I= 97,193)/

2	1.3500E+05,	1.3800E+05,	1.4100E+05,	1.4400E+05,	1.4700E+05,
3	1.5000E+05,	1.5300E+05,	1.5600E+05,	1.5900E+05,	1.6200E+05,
4	1.6500E+05,	1.6800E+05,	1.7100E+05,	1.7400E+05,	1.7700E+05,
5	1.8000E+05,	1.8500E+05,	1.9000E+05,	1.9500E+05,	2.0000E+05,
6	2.0500E+05,	2.1000E+05,	2.1500E+05,	2.2000E+05,	2.2500E+05,
7	2.3000E+05,	2.3500E+05,	2.4000E+05,	2.4500E+05,	2.5000E+05,
8	2.5500E+05,	2.6000E+05,	2.6500E+05,	2.7000E+05,	2.7500E+05,
9	2.8000E+05,	2.8500E+05,	2.9000E+05,	2.9500E+05,	3.0000E+05,
1	3.1000E+05,	3.2000E+05,	3.3000E+05,	3.4000E+05,	3.5000E+05,
2	3.6000E+05,	3.7000E+05,	3.8000E+05,	3.9000E+05,	4.0000E+05,
3	4.1000E+05,	4.2000E+05,	4.3000E+05,	4.4000E+05,	4.5000E+05,
4	4.6000E+05,	4.7000E+05,	4.8000E+05,	4.9000E+05,	5.0000E+05,
5	5.5000E+05,	6.0000E+05,	6.5000E+05,	7.0000E+05,	7.5000E+05,
6	8.0000E+05,	8.5000E+05,	9.0000E+05,	9.5000E+05,	1.0000E+06,
7	1.0500E+06,	1.1000E+06,	1.1500E+06,	1.2000E+06,	1.2500E+06,
8	1.3000E+06,	1.3500E+06,	1.4000E+06,	1.4500E+06,	1.5000E+06,
9	1.5500E+06,	1.6000E+06,	1.6500E+06,	1.7000E+06,	1.7500E+06,
1	1.8000E+06,	1.8500E+06,	1.9000E+06,	1.9500E+06,	2.0000E+06,
2	2.05E6,	2.10E6,	2.15E6,	2.20E6,	2.25E6,
	2.30E6,	5.00E9/			

DATA (PRS(I), I= 1, 96)/

1	1.4751E+01,	1.4244E+01,	1.3751E+01,	1.3273E+01,	1.2808E+01,
2	1.2757E+01,	1.1919E+01,	1.1494E+01,	1.1082E+01,	1.0682E+01,
3	1.0295E+01,	9.9191E+00,	9.5550E+00,	9.2022E+00,	8.8602E+00,
4	8.5288E+00,	8.2076E+00,	7.8965E+00,	7.5950E+00,	7.3029E+00,
5	7.0200E+00,	6.7459E+00,	6.4804E+00,	6.2233E+00,	5.9744E+00,
6	5.7333E+00,	5.5000E+00,	5.2741E+00,	5.0556E+00,	4.8442E+00,
7	4.6397E+00,	4.4421E+00,	4.2510E+00,	4.0664E+00,	3.8882E+00,
8	3.7161E+00,	3.5456E+00,	3.3893E+00,	3.2349E+00,	3.0862E+00,
9	2.9432E+00,	2.8057E+00,	2.6737E+00,	2.5469E+00,	2.4252E+00,
1	2.3086E+00,	2.1970E+00,	2.0901E+00,	1.9878E+00,	1.8901E+00,
2	1.7969E+00,	1.7079E+00,	1.6230E+00,	1.5422E+00,	1.4653E+00,
3	1.3021E+00,	1.2226E+00,	1.1567E+00,	1.0941E+00,	1.0359E+00,
4	1.0801E+00,	9.7726E-01,	8.8474E-01,	8.0151E-01,	7.2660E-01,
5	6.5912E-01,	5.9832E-01,	5.4349E-01,	4.9402E-01,	4.4935E-01,
6	4.0999E-01,	3.7249E-01,	3.3945E-01,	3.0952E-01,	2.8238E-01,
7	2.5774E-01,	2.3534E-01,	2.1492E-01,	1.9638E-01,	1.7954E-01,

```

8 1.6424E-01, 1.5033E-01, 1.3767E-01, 1.2614E-01, 1.1564E-01,
9 1.0697E-01, 9.7349E-02, 8.9389E-02, 8.2123E-02, 7.5488E-02,
1 6.9427E-02, 6.1296E-02, 5.4182E-02, 4.7950E-02, 4.2486E-02/
DATA (PRS(I), I= 97,193)/
2 3.7687E-02, 3.3468E-02, 2.9753E-02, 2.6477E-02, 2.3583E-02,
3 2.1022E-02, 1.8752E-02, 1.6736E-02, 1.4941E-02, 1.3342E-02,
4 1.1915E-02, 1.0635E-02, 9.4846E-03, 8.4520E-03, 7.5254E-03,
5 6.6941E-03, 5.4053E-03, 4.4977E-03, 3.6693E-03, 2.9829E-03,
6 2.4160E-03, 1.9491E-03, 1.5660E-03, 1.2528E-03, 9.9776E-04,
7 7.9096E-04, 6.2403E-04, 4.8991E-04, 3.8268E-04, 2.9739E-04,
8 2.2989E-04, 1.7677E-04, 1.3519E-04, 1.0282E-04, 7.7776E-05,
9 5.8766E-05, 4.4409E-05, 3.3564E-05, 2.5370E-05, 1.9239E-05,
1 1.1291E-05, 6.7945E-06, 4.1841E-06, 2.6533E-06, 1.7334E-06,
2 1.1626E-06, 8.0470E-07, 5.7813E-07, 4.2822E-07, 3.2670E-07,
3 2.5872E-07, 2.1098E-07, 1.7601E-07, 1.4954E-07, 1.2897E-07,
4 1.1263E-07, 9.9395E-08, 8.8517E-08, 7.9451E-08, 7.1756E-08,
5 4.5742E-08, 3.0813E-08, 2.1430E-08, 1.5233E-08, 1.1035E-08,
6 8.1197E-09, 6.0518E-09, 4.5646E-09, 3.4813E-09, 2.6823E-09,
7 2.0842E-09, 1.6320E-09, 1.2873E-09, 1.0226E-09, 8.1773E-10,
8 6.5807E-10, 5.3254E-10, 4.3298E-10, 3.5364E-10, 2.9013E-10,
9 2.3903E-10, 1.9775E-10, 1.6425E-10, 1.3683E-10, 1.1430E-10,
1 9.5740E-11, 8.0400E-11, 6.7691E-11, 5.7134E-11, 4.8334E-11,
2 4.096E-11, 3.478E-11, 2.958E-11, 2.520E-11, 2.150E-11, 0.,0./
DATA (VLS(I), I= 1, 96)/
1137.99 ,
1 1.1380E+03, 1.1337E+03, 1.1296E+03, 1.1258E+03, 1.1221E+03,
2 1.1186E+03, 1.1152E+03, 1.1119E+03, 1.1085E+03, 1.1052E+03,
3 1.1019E+03, 1.0986E+03, 1.0952E+03, 1.0917E+03, 1.0882E+03,
4 1.0845E+03, 1.0808E+03, 1.0770E+03, 1.0730E+03, 1.0689E+03,
5 1.0648E+03, 1.0605E+03, 1.0561E+03, 1.0516E+03, 1.0470E+03,
6 1.0424E+03, 1.0376E+03, 1.0328E+03, 1.0280E+03, 1.0231E+03,
7 1.0181E+03, 1.0132E+03, 1.0082E+03, 1.0033E+03, 9.9841E+02,
8 9.9355E+02, 9.8847E+02, 9.8370E+02, 9.7905E+02, 9.7455E+02,
9 9.7021E+02, 9.6605E+02, 9.6210E+02, 9.5837E+02, 9.5489E+02,
1 9.5167E+02, 9.4872E+02, 9.4607E+02, 9.4372E+02, 9.4168E+02,
2 9.3998E+02, 9.3860E+02, 9.3757E+02, 9.3688E+02, 9.3653E+02,
3 9.3653E+02, 9.3686E+02, 9.3754E+02, 9.3853E+02, 9.4229E+02,
4 9.4399E+02, 9.4748E+02, 9.5103E+02, 9.5459E+02, 9.5812E+02,
5 9.6158E+02, 9.6493E+02, 9.6815E+02, 9.7121E+02, 9.7409E+02,
6 9.7680E+02, 9.7934E+02, 9.8169E+02, 9.8389E+02, 9.8595E+02,
7 9.8790E+02, 9.8990E+02, 9.9277E+02, 9.9565E+02, 9.9856E+02,
8 1.0015E+03, 1.0044E+03, 1.0074E+03, 1.0105E+03, 1.0135E+03,
9 1.0166E+03, 1.0198E+03, 1.0229E+03, 1.0261E+03, 1.0294E+03,
1 1.0326E+03, 1.0375E+03, 1.0424E+03, 1.0473E+03, 1.0521E+03/
DATA (VLS(I), I= 97,193)/
2 1.0568E+03, 1.0614E+03, 1.0658E+03, 1.0698E+03, 1.0736E+03,
3 1.0770E+03, 1.0798E+03, 1.0822E+03, 1.0839E+03, 1.0836E+03,
4 1.0811E+03, 1.0782E+03, 1.0749E+03, 1.0712E+03, 1.0671E+03,
5 1.0628E+03, 1.0549E+03, 1.0464E+03, 1.0373E+03, 1.0279E+03,
6 1.0181E+03, 1.0082E+03, 9.9814E+02, 9.8865E+02, 9.7798E+02,
7 9.6798E+02, 9.5808E+02, 9.4828E+02, 9.3858E+02, 9.2892E+02,
8 9.1925E+02, 9.0947E+02, 8.9945E+02, 8.8903E+02, 8.8400E+02,
9 8.8400E+02, 8.8400E+02, 8.8400E+02, 8.8400E+02, 8.9450E+02,

```

```

1  9.1635E+02,  9.3768E+02,  9.6117E+02,  9.9488E+02,  1.0275E+03,
2  1.0591E+03,  1.1170E+03,  1.1745E+03,  1.2297E+03,  1.3138E+03,
3  1.4106E+03,  1.5012E+03,  1.5866E+03,  1.6676E+03,  1.7449E+03,
4  1.8189E+03,  1.8900E+03,  1.9585E+03,  2.0247E+03,  2.0764E+03,
5  2.2660E+03,  2.3721E+03,  2.4532E+03,  2.5195E+03,  2.5841E+03,
6  2.6358E+03,  2.6853E+03,  2.7340E+03,  2.7818E+03,  2.8262E+03,
7  2.8644E+03,  2.9022E+03,  2.9394E+03,  2.9762E+03,  3.0125E+03,
8  3.0484E+03,  3.0782E+03,  3.1060E+03,  3.1334E+03,  3.1607E+03,
9  3.1877E+03,  3.2144E+03,  3.2392E+03,  3.2565E+03,  3.2736E+03,
1  3.2907E+03,  3.3077E+03,  3.3246E+03,  3.3414E+03,  3.3544E+03,
2  3365., 3376., 3387., 3397., 3408., 3418., 3418./

```

```

C      FW(A,R,C,D) = WA * (1. - 1./EXP(1.689*A/C*B/D))
C
C      SV(17)=3.5
C      SV(20)=35.
C
C      CALL CHECK
C
10  SW(1) = 1.1
    W(37) = 0.
C
C      **** SYNTH CALL 1 ****
C
99  SW(3) = 0.
    CALL OVERLAY(TTSS,1,0,6HRECALL)
    CALL SYNTH
100 SW(1) = 2.1
C
C      **** WTVOL ****
C      CALL OVERLAY(TTSS,3,0,6HRECALL)
C      CALL WTVOL
C      IF(W(36).GT..5) GO TO 998
C
C      **** GTSM ****
C
    SV(35) = 1.1
    SV(43) = 1.1
C      CALL OVERLAY(TTSS,2,0,6HRECALL)
C      CALL GTSM
C      IF(W(36).GT..5) GO TO 998
C
C      **** SYNTH CALL 2 ****
C      CALL OVERLAY(TTSS,1,0,6HRECALL)
C      CALL SYNTH
C      IF(SV(35).LT.1.5) GO TO 11
C      TSND = IFIX(SV(43))
C      GO TO (21,22,23), ISND
22  SW(2) = .5
    GO TO 21
23  SW(2) = 1.
C 21 CALL OVERLAY(TTSS,2,0,6HRECALL)

```

```

21 CALL GTSM
   GO TO (31,32,33), ISND
32 SW(2) = 1.
   GO TO 31
33 SW(2) = 2.
31 IF(W(36).GT..5) GO TO 998
   SW(15) = Z(95) + SQ(10,3)
C **** SW(15) IS THE FLYBACK RANGE ****
11 CONTINUE
   IFLY = SQ(32,1)
   GO TO (600,650,700), IFLY
600 CONTINUE
   DMUB = TB34(3) - SQ(19,2)
   IF(SW(3).GT.1.5.AND.ABS(DMUB).GT..001)
1 SE(8) = (SW(15) - SQ(19,1))/DMUB
   SQ(19,1) = SW(15)
   SQ(19,2) = TB34(3)
   SQ(19,3) = SW(14)/SW(12)*SW(11)/1.689
   SCB(214) = EXP(SW(15)/SQ(19,3))
   SCB(214) = SCB(214) * SV(34)/(SV(34) - SQ(32,4))
   SCB(214) = SCB(214) - 1.
   GO TO 800
650 CONTINUE
   DWF = SQ(32,2) * SV(34)
   SQ(35,2) = DWF
   WA = SV(34) - DWF
   R2 = SW(15) - SQ(32,5) - SQ(33,1) - SQ(33,2)
   DW = FW(R2,SQ(33,4),SQ(34,2),SQ(34,5))
   SQ(35,3) = DW
   WA = WA - DW
   DWF = DWF + DW
   DW = SQ(32,3) * WA
   SQ(35,4) = DW
   GO TO 700
700 CONTINUE
   WA = SV(34)
   DWF = FW(R1,SQ(33,3),SQ(34,1),SQ(34,4))
   SQ(35,2) = DWF
   WA = SV(34) - DWF
   R2 = SW(15) - SQ(32,5) - SQ(33,1) - SQ(33,2)
   DW = FW(R2,SQ(33,4),SQ(34,2),SQ(34,5))
   SQ(35,3) = DW
   DWF = DWF + DW
   WA = WA - DW
   DW = FW(R3,SQ(33,5),SQ(34,3),SQ(35,1))
   SQ(35,4) = DW
790 CONTINUE
   DWF = DWF + DW + SQ(32,4)
   WA = WA - DW - SQ(32,4)
   SCB(214) = DWF/WA
   SQ(19,3) = SQ(34,5)/1.689*SQ(34,2)/SQ(33,4)
   SQ(35,5) = R2

```

800 CONTINUE
IF(APS(SV(43)-3.1).GE..5) GO TO 34
SW(1) = 3.1
C CALL OVERLAY(TTSS,1,0,6HRECALL)
CALL SYNTH
34 IF(SW(2).GT.1.) GO TO 10
GO TO 100
C
998 IF(V(64).LT..5) GO TO 999
V(64) = .1
SW(1) = 1.1
GO TO 99
999 STOP
END

GDC-08870-002

Appendix IV-2 Subroutine CHECK

```
SUBROUTINE CHECK  
COMMON/IDATX/IDATE,ITIME  
COMMON/HEAD /TITLES(10)  
RETURN  
ENTRY DATE  
WRITE (6,1484) (TITLES(I),I=1,10)  
RETURN
```

```
C  
1484 FORMAT(10A6)  
END
```


Appendix IV-3 Subroutine STORE

```

SUBROUTINE STORE
REAL ISP,K,LF,MP,NCREW,LBODY,NPASS
REAL NENGS
DIMENSION SKO(30),SCO(300),TO4(6),TO20(10),TO27(6),TO34(6),
1 TO48(10),TO49(10),TO50(10),TO57(6)
DIMENSION SKB(30),SCB(300),TB4(6),TB20(10),TB27(6),TB34(6),
1 TB48(10),TB49(10),TB50(10),TB57(6)
DIMENSION OWSAVE(10),BWSAVE(10)

```

SYNTHESIS CONTROL SUBROUTINE FOR TWO-STAGE VEHICLES

C
C
C

```

COMMON/CINPUT/
1ANFNGS ,ANTANK ,ASPATO ,ASWEEP ,C(300) ,CBBODY ,CFUEL(6),
2CHBODY ,CLBODY ,CSBODY ,CSFAIR ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN ,CSVERT ,CSWING ,CTHRET ,CTHST2 ,DEF(5) ,FXWOVS ,
4ISP(6) ,TTPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5NENGS ,NLISTO ,NPASS ,NWL ,NLIAM ,Q ,RHOFU ,
6RHOFU2 ,RHOX ,RHOX2 ,SBOOY ,TOL ,TOVERC ,TPRATO ,
7TYTAIL ,VBODY ,WGROSS
COMMON/VOLCAL/BRODY,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HBODY ,LBODY ,RTOD ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPLAN ,STPS(1) ,SVERT ,SWING ,SXPOS ,TDEL ,TROOT ,
4TTOT ,TTOT2 ,TTOTAL ,VBOOYA ,VBOOY1 ,VBOOY2 ,VCARGO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGBAY ,VOTHER ,VOXTK ,
6VOXTK2 ,VPROP ,VSTRUC
COMMON/WTCALC/ ABFSYS ,WAFETK ,WABFU ,WABPR ,WACRES ,
1WACS ,WACSF0 ,WACSTK ,WAFRO ,WAUXT ,WBASIC ,WBODY ,
2WBUMP ,WCARGO ,WCOMM ,WCONT ,WCOVER ,WDECAY ,WDIST1 ,
3WDIST2 ,WDOCK ,WDPLOY ,WDTRANS ,WDRY ,WELCAD ,WEMPTY ,
4WENGMT ,WENG5 ,WENG52 ,WFATR ,WFCONT ,WFDCAY ,WFROST ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFUOX ,WFURES ,
6WFUSYS ,WFUTK ,WFUTK2 ,WFUTOT ,WFUTRP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLOSS ,WLRD ,WNACEL ,WODCAY ,
9WOIL ,WOILRS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,WOXID ,
1WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPERS ,WPOWCD ,WPOWER ,WPOWFO ,
3WPOWRS ,WPOWTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFFUL ,
4WRESTD ,WRESRV ,WSEAL ,WSFCST ,WSORCE ,WSRTRP ,WSTAR ,
5WSURF ,WTABO ,WTHRST ,WTO ,WTPS ,WVENT ,WWAIT(10),
6WNET ,WNING ,WZROFU
COMMON/ORRINX/
1 TO1,T02,T03,T04,T05,T06,T07,T08,T09,T010,T011,T012,T013,T014,
2 TO15,T016,T017,T018,T019,T020,T021,T022,T023,T024,T025,T026,T027,
3 T028,T029,T030,T031,T032,T033,T034,T035,T036,T037,T038,T039,T040,
4 T041,T042,T043,T044,T045,T046,T047,T048,T049,T050,T051,T052,T053,
5 T054,T055,T056,T057,SKO,SCO
COMMON/ORBINY/
1 TB1,TB2,TB3,TB4,TB5,TB6,TB7,TB8,TB9,TB10,TB11,TB12,TB13,TB14,
2 TB15,TB16,TB17,TB18,TB19,TB20,TB21,TB22,TB23,TB24,TB25,TB26,TB27,
3 TB28,TB29,TB30,TB31,TB32,TB33,TB34,TB35,TB36,TB37,TB38,TB39,TB40,
4 TB41,TB42,TB43,TB44,TB45,TB46,TB47,TB48,TB49,TB50,TB51,TB52,TB53,

```

```

5  TB54,TB55,TB56,TB57,SKB,SCB
   COMMON/ORB0/T054,T059,TB58,TB59
   COMMON/MF/T014A,TR14A
   SWING=CSWING
   IF(CSHORZ.GT.20.) CSHORZ=CSHORZ/SWING
   VBODYR = 1./((VBODY** .666667)
   IF(CSBODY.GT.20.) CSBODY=CSBODY*VBODYR
   IF(CSPLAN.GT.20.) CSPLAN=CSPLAN*VBODYR
   IF(CSVERT.GT.5.) CSVERT=CSVERT*VBODYR
   IF(CSFAIR.GT.20.) CSFAIR = CSFAIR/(CSBODY/VBODYR)
   VBODYR = SQRT(VBODYR)
   IF(CRBODY.GT.5.) CRBODY = CRBODY*VBODYR
   IF(CHBODY.GT.5.) CHBODY = CHBODY*VBODYR
   IF(CLBODY.GT.20.) CLBODY=CLBODY *VBODYR
   RETURN
   ENTRY ORRST0
   T02 = CRBODY
   DO 20 I = 1,6
20  T04(I) = CFUEL(I)
   T05 = CHBODY
   T06 = CLBODY
   T07 = CSBODY
   T010 = CSFAIR
   T011 = CSFUTK
   T012 = CSOXTK
   T013 = CSHORZ
   T015 = CSPLAN
   T016 = CSVERT
   T017 = CSWING
   T018 = CTHRST
   T019 = CTHST?
   DO 24 I = 1,6
24  T027(I) = ISP(I)
   DO 26 I = 1,6
26  T034(I) = MR(I)
   T035 = NCREW
   T036 = NENGs
   T038 = NPASS
   T041 = RHOFU
   T042 = RHOFU2
   T043 = RHOX
   T044 = RHOX2
   T045 = SWING
   T047 = TOVERC
   T051 = VBODY
   T052 = VFUTK
   T053 = VFUTK?
   T054 = VOXTK
   T055 = VOXTK?
   T056 = WGROSS
   DO 30 I = 1,30
30  SK0(I)= K(I)

```

```

DO 50 I = 1,300
50 SC0(I) = C(I)
T059 = LF
T060 = TPRATO
T061 = ASPATO
T062 = FXWOVS
T063 = NWL
T064 = ITPS
T065 = PCHAM
T066 = ANENGs
T067 = ANTANK
T068 = TYTAIL
T069 = NLISTO
T070 = ASWEEP
T071 = WFPOST
T072 = WFUTRP
T073 = WOXTRP
T074 = WSPTRP
T075 = WDFCAY
T076 = WFIRES
T077 = WOXRES
T078 = WACSF0
T079 = WFULOS
T080 = WOXLOS
T081 = WPOWFO
T082 = WGASPR
T083 = WACRES
T084 = WPOWRS
DO 999 I = 1,10
999 OMSAVE(I) = WWAIT(I)
RETURN
ENTRY 900ST0
TB2 = CRR0DY
DO 60 I = 1,6
60 TB4(I) = CFUEL(I)
TB5 = CHBODY
TB6 = CLBODY
TB7 = CSBODY
TB10 = CSFAIR
TB11 = CSFUTK
TB12 = CSOXTK
TB13 = CSHORZ
TB15 = CSPLAN
TB16 = CSVERT
TB17 = CSMING
TB18 = CTHRST
TB19 = CTHST2
DO 64 I = 1,6
64 TB27(I) = ISP(I)
DO 66 I = 1,6
66 TB34(I) = MR(I)
TB35 = NCREW

```

```

TB36 = NENG5
TB38 = NPASS
TB41 = RHOFU
TB42 = RHOFU2
TB43 = RHOX
TB44 = RHOX2
TB45 = SWING
TB47 = TOVERC
TB51 = VBODY
TB52 = VFUTK
TB53 = VFUTK?
TB54 = VOXTK
TB55 = VOXTK?
TB56 = WGROSS
DO 80 I = 1, 30
80 SKR(I) = K(I)
DO 100 I = 1, 300
100 SCR(I) = C(I)
TB59 = LF
TB60=TPRATO
TB61=ASRATO
TB62=FXWOVS
TB63=NWL
TB64=ITPS
TB65=PCHAM
TB66=ANENG5
TB67=ANTANK
TB68=TYTAIL
TB69=NLISTO
TB70=ASWEEP
TB71=WFROST
TB72=WFUTRP
TB73=WOXTRP
TB74=WSRTRP
TB75=WDECAY
TB76=WFURES
TB77=WOXRES
TB78=WACSFO
TB79=WFULOS
TB80=WOXL05
TB81=WPOWFO
TB82=WGASPR
TB83=WACRES
TB84=WPOWRS
DO 997 I=1,10
997 RMSAVE(I)=NWAIT(I)
RETURN
ENTPY ORBCAL
CHBODY = T02
DO 200 I = 1,6
200 CFUEL(I) = T04(I)
CHBODY = T05

```

CLBODY = T06
 CSBODY = T07
 CSFAIP = T010
 CSFUTK = T011
 CSOXTK = T012
 CSHOP7 = T013
 CSPLAN = T015
 CSVERT = T016
 CSWING = T017
 CTHRST = T018
 CTHST2 = T019
 DO 204 I = 1,6
 204 ISP(I) = T027(I)
 DO 206 I = 1,6
 206 MR(I) = T034(I)
 NCREW = T035
 NENGCS = T036
 NPASS = T038
 RHOFU = T041
 RHOFU2 = T042
 RHOX = T043
 RHOX2 = T044
 SWING = T045
 TOVERC = T047
 VBODY = T051
 VFUTK = 0.
 VFUTK2 = T053
 VOXTK = 0.
 VOXTK2 = T055
 WGROSS = T056
 DO 230 I = 1,30
 230 K(I) = SK0(I)
 DO 260 I = 1,300
 260 C(I) = SC0(I)
 LF = T059
 TPRATO=T060
 ASRATO=T061
 FXNOVS=T062
 NWL=T063
 ITPS=T064
 PCHAM=T065
 ANENGCS=T066
 ANTANK =T067
 TYTAIL=T068
 NLIST0=T069
 ASWEEP=T070
 WFPOST=T071
 WFUTRP=T072
 WOXTRP=T073
 WSRTRP=T074
 WDECAY=T075
 WFURES=T076

```

WOXPRES=T077
WACSF0=T078
WFULOS=T079
WCXLOS=T080
WPOWFO=T081
WGASPR=T082
WACRES=T083
WPOWRS=T084
DO 996 I=1,10
996  KWAIT(I)=OWSAVE(I)
      RETURN
      ENTRY BCOCAL
      CBBODY  = TB2
      DO 300I = 1,6
300   CFUEL(I) = TB4(I)
      CHBODY  = TB5
      CLBODY  = TB6
      CSPODY  = TB7
      CSFAIR  = TB10
      CSFUTK  = TB11
      CSOXTK  = TB12
      CSHORZ  = TB13
      CSPLAN  = TB15
      CSVERT  = TB16
      CSWING  = TB17
      CTHPST  = TB18
      CTHST2  = TB19
      DO 304I = 1,6
304   ISP(I)  = TB27(I)
      DO 306I = 1,6
306   MR(I)   = TB34(I)
      NCREW   = TB35
      NENG5   = TB36
      NPASS   = TB38
      RHOFU   = TB41
      RHOFU2  = TB42
      RHOX    = TB43
      RHOX2   = TB44
      SWING   = TB45
      TOVERC  = TB47
      VR0DY   = TB51
      VFUTK   = 0.
      VFUTK2  = TB53
      VOXTK   = 0.
      VOXTK2  = TB55
      DO 400 I = 1,30
400   K(I)    = SKB(I)
      DO 450 I = 1,300
450   C(I)    = SCB(I)
      LF      = TB59
      TPRATO  = TB60
      ASRATO  = TB61

```

```

FXHOVS=TB62
NWL=TB63
IIPS=TB64
PCHAM=TB65
ANENGs=TB66
ANTANK =TB67
TYTAIL=TB68
NLISTO=TB69
ASWEEP=TB70
WFROST=TB71
WFUTRP=TB72
WOXTRP=TB73
WSRTRP=TB74
WDECAY=TB75
WFURES=TB76
WOXRES=TB77
WACSF0=TB78
WFULOS=TB79
WOXLOS=TB80
WPOWFO=TB81
WGASPR=TB82
WACRES=TB83
WPOWRS=TB84
DO 995 I=1,10
995 WAIT(I)=BMSAVE(I)
IF(TB11.EQ.0..AND.SCB(129).EQ.0.) C(129) = TTOT * NENGs
IF(SCB(153).EQ.0.) C(154) = WAUXT * SCB(263)
C(105)=WGROSS
WGROSS = TB56
WJET(3)=C(105)+C(78)
RETURN
ENI

```

PRECEDING PAGE BLANK NOT FILMED

Appendix IV-4 Program SYNTH

GDC-08870-002

```
      SUBROUTINE SYNTH
C      OVERLAY(TTSS,1,0)
C      PROGRAM SYNTH
      COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1  VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2  C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3  XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4  TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5  PS(15)
      IGO = IFIX(SW(1))
      GO TO (100,200,300), IGO
100  CALL FRENCH
      CALL V=HDF
      CALL TRAJA
      RETURN
200  CALL ITER8
      RETURN
300  CALL ITER8
      RETURN
      END
```


PRECEDING PAGE BLANK NOT FILMED

Appendix IV-5 Subroutine FRENCH

GDC-CBP70-002

```

SUBROUTINE FRENCH
REAL ISP,K,LF,MR,NCREW,LBODY,NPASS
REAL NENG5
DIMENSION SKO(30),SCO(300),TO4(6),TO20(10),TO27(6),TO34(6),
1 TO48(10),TO49(10),TO50(10),TO57(6)
DIMENSION SKB(30),SCB(300),TB4(6),TB20(10),TB27(6),TB34(6),
1 TB48(10),TB49(10),TB50(10),TB57(6)
COMMON/PO/MPRNT,MPNCH
COMMON/ORBINK/
1 TO1,TO2,TO3,TO4,TO5,TO6,TO7,TO8,TO9,TO10,TO11,TO12,TO13,TO14,
2 TO15,TO16,TO17,TO18,TO19,TO20,TO21,TO22,TO23,TO24,TO25,TO26,TO27,
3 TO28,TO29,TO30,TO31,TO32,TO33,TO34,TO35,TO36,TO37,TO38,TO39,TO40,
4 TO41,TO42,TO43,TO44,TO45,TO46,TO47,TO48,TO49,TO50,TO51,TO52,TO53,
5 TO54,TO55,TO56,TO57,SKO,SCO
COMMON/ORBINY/
1 TB1,TB2,TB3,TB4,TB5,TB6,TB7,TB8,TB9,TB10,TB11,TB12,TB13,TB14,
2 TB15,TB16,TB17,TB18,TB19,TB20,TB21,TB22,TB23,TB24,TB25,TB26,TB27,
3 TB28,TB29,TB30,TB31,TB32,TB33,TB34,TB35,TB36,TB37,TB38,TB39,TB40,
4 TB41,TB42,TB43,TB44,TB45,TB46,TB47,TB48,TB49,TB50,TB51,TB52,TB53,
5 TB54,TB55,TB56,TB57,SKB,SCB
COMMON / JUMPY / JUMP, W8IG, W800
COMMON/CINPUT/
1ANENG5 ,ANTANK ,ASRATO ,ASWEEP ,C(300) ,CBRODY ,CFUEL(6),
2CHRODY ,CLBODY ,CSBODY ,CSFAIR ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN ,CSVERT ,CSWING ,CTHRST ,CTHST2 ,DEF(5) ,FXWOVS ,
4ISP(6) ,ITPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5NENG5 ,NLISTO ,NPASS ,NWL ,PCHAM ,Q ,RHOFU ,
6RHOFU2 ,RHOX ,RHOX2 ,SBODY ,TOL ,TOVERC ,TPRATO ,
7TYTAIL ,VBOY ,WGRASS
COMMON/VOLCAL/5BODY,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HRODY ,LBODY ,RTOD ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPAN ,STPS(1) ,SVERT ,SWING ,SXPOS ,TDEL ,TROOT ,
4TTOT ,TTOT2 ,TTOTAL ,VBOYDA ,VBOYD1 ,VBOYD2 ,VCARGO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGRAY ,VOTHER ,VOXTK ,
6VOXTK2 ,VPROP ,VSTRUC
COMMON/WTCALC/ ABFSYS ,WABFTK ,WABFU ,WABPR ,WACRES ,
1WACS ,WACSF0 ,WACSTK ,WAERO ,WAUXT ,WBASIC ,WBODY ,
2WBPUMP ,WCARGO ,WCOMM ,WCONT ,WCOVER ,WDECAY ,WDIST1 ,
3WDIST2 ,WDOCK ,WDPLOY ,WDRANS ,WDRY ,WELCAD ,WEMPTY ,
4WENGMT ,WENG5 ,WENG52 ,WFAIR ,WFCONT ,WFDCAY ,WFROST ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFUOX ,WFURES ,
6WFUSYS ,WFUTK ,WFUTK2 ,WFUTOT ,WFUTRP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLOSS ,WLRO ,WNACEL ,WODCAY ,
9WOIL ,WOILRS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,NOXID ,
1WOXLOS ,WOXRES ,WOXSYS ,WOXTL ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPERS ,WPOWCD ,WPOWER ,WPOWFO ,
3WPOWRS ,WPOWTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
4WRESID ,WRESRV ,WSEAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF ,WTARC ,WTHRST ,WTO ,WTPS ,WVERT ,HWAIT(10),
6WNET ,WNING ,WZROFU

```

```

C      JO = 0, USE STORED DATA ,NO CHANGES FROM PRECEEDING CASE OF A
C      MULTIPLE RUN
C      JO = 1, READ ORBITER INPUT DATA
C      JO = 2, READ CHANGES TO ORBITER INPUT FOR A CASE OF A MULTIPLE RUN
C      JB = 0, USE STORED DATA ,NO CHANGES FROM PRECEEDING CASE OF A
C      MULTIPLE RUN
C      JB = 1, READ BOOSTER INPUT DATA
C      JB = 2, READ CHANGES TO BOOSTER INPUT FOR A CASE OF A MULTIPLE RUN

C      MPRNTO = 0, NO PRINT OUT OF INPUT  ORBITER DATA
C      MPRNTO = 1, FOR PRINT OUT OF INPUT  ORBITER DATA
C      MPRNTB = 0, NO PRINT OUT OF INPUT  BOOSTER DATA
C      MPRNTB = 1, FOR PRINT OUT OF INPUT  BOOSTER DATA
C      MPNCHO = 0, NO PUNCH OUT OF INPUT ORBITER DATA
C      MPNCHO = 1, FOR PUNCH OUT OF INPUT ORBITER DATA
C      MPNCHB = 0, NO PUNCH OUT OF INPUT BOOSTER DATA
C      MPNCHB = 1, FOR PUNCH OUT OF INPUT BOOSTER DATA
C      MPRNTO = 0
C      MPRNTB = 0
C      MPNCHO = 0
C      MPNCHB = 0
C      READ(5,1000,END=10)JO,JR,MPRNT0,MPRNTB,MPNCHO,MPNCHB
C      GO TO 15
C      READ(5,1000)JO,JR,MPRNT0,MPRNTB,MPNCHO,MPNCHB
C      IF(EOF,5)10,15
10 CALL EXIT
15 IF(JO.EQ.0) GO TO 20
   IF(JO.EQ.1) CALL SET0
   WRITE (6,1005)
   IF(JO.EQ.2) CALL ORBCAL
   MPNCH = MPNCHO
   MPRNT = MPRNTO
   CALL READY
   CALL STORE
   CALL ORBSTO
20 IF(JB.EQ.0) RETURN
   IF(JB.EQ.1) CALL SET0
   WRITE (6,1005)
   IF(JB.LT.2) GO TO 25
   CALL BOOCAL
   C(3) = SCB(3)
   C(5) = SCB(5)
   C(7) = SCB(7)
   C(146) = SCB(146)
   C(31) = SCB(31)
   C(56) = SCB(56)
   C(129) = SCB(129)
   C(211)=SCB(211)
   C(67) = SCB(67)
25 CONTINUE
   MPRNT = MPRNTB
   MPNCH = MPNCHB

```

GDC-08870-002

CALL READY
CALL STORE
CALL BOOSTO
RETURN
1000 FORMAT (6I2)
1005 FORMAT (2H1)
END

PRECEDING PAGE BLANK NOT FILMED

Appendix IV-6 Subroutine READY

GDC-DBR76-002

SUBROUTINE READY
REAL ISP,K,LF,MR,NCREW,LBODY,NPASS
REAL KIN
REAL NFNGS
COMMON/PO/MPRNT,MPNCH

C *** READY ***

C

C

C

READY - SETS UP FOR AND CALLS REDATA

COMMON/HEAD /TITLES(10)

COMMON/CINPUT/

1ANENGs	,ANTANK	,ASRATO	,ASWEEP	,C(300)	,CBBODY	,CFUEL(6),
2CHBODY	,CLBODY	,CSBODY	,CSFAIR	,CSFUTK	,CSHORZ	,CSOXTK ,
3CSPLAN	,CSVEPT	,CSWING	,CTHRST	,CTHST2	,DEF(5)	,FXWOVS ,
4ISP(6)	,ITPS	,K(30)	,KIN	,LF	,MR(6)	,NCREW ,
5NENGs	,NLISTO	,NPASS	,NWL	,PCHAM	,Q	,RHOFU ,
6RHOFU2	,RHOX	,RHOX2	,SBODY	,TOL	,TOVERC	,TPRATO ,
7TYTAIL	,VBODY	,WGOSS				

COMMON/VOLCAL/RBODY,CROOT

2HBODY	,LBODY	,RTOD	,SFAIR	,SFUTK	,SHORZ	,SOXTK ,
3SPLAN	,STPS(1)	,SVERT	,SWING	,SXPOS	,TDEL	,TROOT ,
4TTOT	,TTOT2	,TTOTAL	,VBODYA	,VBODY1	,VBODY2	,VCARGO ,
5VCREW	,VFUTK	,VFUTK2	,VINSTK	,VLGRAY	,VOTHER	,VOXTK ,
6VOXTK2	,VPROP	,VSTRUC				

COMMON/WTCALC/

1WACS	,WACSFO	,WACSTK	,WAERO	,WAUXT	,WBASIC	,WBODY ,
2WRPUMP	,WCARGO	,WCOMM	,WCONT	,WCOVER	,WDECAY	,WDIST1 ,
3WDIST2	,WDOCK	,WDPLOY	,WDRANS	,WDRY	,WELCAD	,WEMPTY ,
4WENGMT	,WENGs	,WENGs2	,WFAIR	,WFCONT	,WFOCAY	,WFROST ,
5WFU2(3)	,WFUEL(6)	,WFUL	,WFULOS	,WFUNCT	,WFUOX	,WFURES ,
6WFUSYS	,WFUTK	,WFUTK2	,WFUTOT	,WFUTRP	,WGASPR	,WGNV ,
7WHORZ	,WHYCAD	,WINFUT	,WINOXT	,WINSTK	,WINST	,WINSUL ,
8WJET(6)	,WLANCH	,WLG	,WLOSS	,WLRD	,WNACEL	,WOCAY ,
9WOIL	,WOILRS	,WORSUL	,WOVERS	,WOX(6)	,WOX2(3)	,WOXID ,
1WOXLOS	,WOXRES	,WOXSYS	,WOXTK	,WOXTK2	,WOXTOT	,WOXTRP ,
2WP	,WPASS	,WPAYL	,WPERS	,WPOWCD	,WPOWER	,WPOWFO ,
3WPOWRS	,WPOWTK	,WPPROV	,WPREIG	,WPROP	,WPRSYS	,WREFUL ,
4WPRESID	,WPRESRV	,WSEAL	,WSECST	,WSORCE	,WSRTRP	,WSTAB ,
5WSURF	,WTABC	,WTHRST	,WTO	,WTPS	,WVERT	,WWAIT(10),
6WWET	,WWING	,WZROFU				

C

C

C

C

NAMelist TO READ IN DATA

NAMelist/DATA3/

1ANENGs	,ANTANK	,ASRATO	,ASWEEP	,C	,CBBODY	,CFUEL ,
2CHBODY	,CLBODY	,CSBODY	,CSFAIR	,CSFUTK	,CSHORZ	,CSOXTK ,
3CSPLAN	,CSVERT	,CSWING	,CTHRST	,CTHST2	,DEF	,FXWOVS ,
4ISP	,ITPS	,K	,KIN	,LF	,MR	,NCREW ,
5NENGs	,NLISTO	,NPASS	,NWL	,PCHAM	,Q	,RHOFU ,
6RHOFU2	,RHOX	,RHOX2	,SBODY	,TOL	,TOVERC	,TPRATO ,

7TYTAIL,VBODY,WGROSS

NAMFLIST TO PRINT OUT INPUT DATA

NAMFLIST TO PUNCH OUT INPUT DATA

NAMFLIST/PRNTX/

1ANENG	,ANTANK	,ASRATO	,ASWEEP	,C	,CBBODY	,CFUEL	,
2CHBODY	,CLBODY	,CSBODY	,CSFAIR	,CSFUTK	,CSHORZ	,CSOXTK	,
3CSPLAN	,CSVERT	,CSWING	,CTHRST	,CTHST2		,FXWOVS	,
4ISP	,ITPS	,K	,KIN	,LF	,MR	,NCREW	,
5NENG	,NLISTO	,NPASS	,NWL	,PCHAM	,Q	,RHOFU	,
6RHOFU2	,RHOX	,RHOX2	,SBOY	,TOL	,TOVERC	,TPRATO	,
7TYTAIL	,VBOY	,WGROSS					

5000 FORMAT(10A6)

1100 READ(5,5000) TITLES

READ(5,DATA3)

IF(MPRNT.EQ.1) WRITE (6,PRNTX)

IF(MPNCH.NE.1) RETURN

WRITE (8,PRNTX)

RETURN

END

Appendix IV-7 Subroutine SET0

GDC-08870-002

```
SUBROUTINE SET0
COMMON/CINPUT/A(392)
COMMON/VOLCAL/D(37)
COMMON/WTCALC/C(134)
DO 10 I=1,392
10 A(I)=0
DO 30 I=1,134
30 C(I)=0
DO 40 I=1,37
40 D(I)=0
RETURN
END
```

PRECEDING PAGE BLANK NOT FILMED

GDC-DHP70-002

Appendix IV-8 Subroutine VEHDF

```
SUBROUTINE VEHDF
REAL NXFOR
REAL IVACO,IVACB,ISLO,ISLB,IDVEL
DIMENSION IVACO(7),ISLO(7),TFCTRO(7),TFCTRB(4),IVACB(4),ISLB(4)
1 ,PRNTX(3)
DIMENSION SKO(30),SCO(300),T04(6),T020(10),T027(6),T034(6),
1 T048(10),T049(10),T050(10),T057(6)
COMMON/ATA2X/ ALD,FBPAR,IDVEL,ISLB,ISLO,IVACB,IVACO,PERISP,OMAX,
1 QMXS,SFC,SLVOUT,COPIES,SYNIT,TFCTRB,TFCTRO,TOLMU,TOLTH,
2 TRATIO,TWLO,TWLOI,WTOUT,FIRE,BOOTW,VCRUSE
3 , NXFOR,PRNTX,FSEC
4 ,SOLID,AS,BS,SISP,SINERT,SAE,TSRO
6 ,FBFUEL,CA,CR,WFLYX,RT,R1,R3,SFC1,SFC2,SFC3,ALD1,ALD2,ALD3,VFLY1,
7 VFLY2,VFLY3
COMMON/FLY/FLYBCK
COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VO(78,4),WO(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
COMMON/ORBINX/
1 T01,T02,T03,T04,T05,T06,T07,T08,T09,T010,T011,T012,T013,T014,
2 T015,T016,T017,T018,T019,T020,T021,T022,T023,T024,T025,T026,T027,
3 T028,T029,T030,T031,T032,T033,T034,T035,T036,T037,T038,T039,T040,
4 T041,T042,T043,T044,T045,T046,T047,T048,T049,T050,T051,T052,T053,
5 T054,T055,T056,T057,SKO,SCO
DIMENSION SKB(30),SCB(300),TB4(6),TB20(10),TB27(6),TB34(6),
1 TB48(10),TB49(10),TB50(10),TB57(6)
COMMON/ORRINY/
1 TB1,TB2,TB3,TB4,TB5,TB6,TB7,TB8,TB9,TB10,TB11,TB12,TB13,TB14,
2 TB15,TB16,TB17,TB18,TB19,TB20,TB21,TB22,TB23,TB24,TB25,TB26,TB27,
3 TB28,TB29,TB30,TB31,TB32,TB33,TB34,TB35,TB36,TB37,TB38,TB39,TB40,
4 TB41,TB42,TB43,TB44,TB45,TB46,TB47,TB48,TB49,TB50,TB51,TB52,TB53,
5 TB54,TB55,TB56,TB57,SKB,SCB
COMMON/XCLVG/CLVG
COMMON/XDRNG/DRNG
COMMON/XWPO/WPOREQ,WOREQ,GWREQ
COMMON/PO/MPRNT,MPNCH
NAMELIST/ATA2/ALD,FBPAR,IDVEL,ISLB,ISLO,IVACB,IVACO,PERISP,OMAX,
1 QMXS,SFC,SLVOUT,COPIES,SYNIT,TFCTRB,TFCTRO,TOLMU,TOLTH,
2 TRATIO,TWLO,TWLOI,WTOUT,FIRE,BOOTW,VCRUSE
3 ,NXFOR,PRNTX,FSEC,CLVG,DRNG
4 ,SOLID,AS,BS,SISP,SINERT,SAE,TSRO ,FLYBCK
5 ,WPOREQ,WOREQ,GWREQ
6 ,FBFUEL,CA,CR,WFLYX,RT,R1,R3,SFC1,SFC2,SFC3,ALD1,ALD2,ALD3,VFLY1,
7 VFLY2,VFLY3
C **** READ SYNTHESIS DATA ****
READ(5,DATA2)
IF(MPRNT.EQ.1) WRITE(6,DATA2)
SE(1) = IVACB(1)
SE(2) = IVACO(1)
```

SE(3) = ISLB(1)
 SE(4) = ISLC(1)
 SF(5) = FIRE+.1
 SE(6) = BOOTW
 SF(7) = QMXS
 SE(8)=FBPAR
 IF(FLYACK.EQ.3.) SE(8)=0.
 SE(9)=QMAX
 SF(11)= IVACB(2)
 SE(12)= IVACO(2)
 SF(13)= ISLB(2)
 SE(14)= ISLO(2)
 SF(15)= IVACO(5)
 SE(16)= ISLO(5)
 SF(17)= IVACO(6)
 SE(18)= ISLO(6)
 SF(19)=IVACO(7)
 SE(20)=ISLO(7)
 SE(21)=TFCTRO(1)
 SE(22)=TFCTRO(2)
 SE(23)=TFCTRO(5)
 SE(24)=TFCTRO(6)
 SE(25)=TFCTRO(7)
 SF(26)=TFCTRB(1)
 SE(27)=TFCTRB(2)
 SF(29) = IVACB(3)
 SF(30) = IVACO(3)
 SE(31) = ISLB(3)
 SE(32) = ISLO(3)
 SE(33) = IVACB(4)
 SE(34) = IVACO(4)
 SE(35) = ISLB(4)
 SE(36) = ISLO(4)
 SE(37) = TFCTRB(4)
 SF(38) = TFCTRO(4)
 SE(39) = NXFOB
 SW(2) = .5
 SW(4) = SYNTT
 SW(5) = TOLNU
 SW(6) = TRATIO
 SW(7) = PERISP
 SW(9) = CLVG
 SW(11) = ALD
 SW(12) = SFC
 SW(13) = SLVOUT
 SW(15)=0.
 SW(14) = VCRUSE
 SW(16) = WTOUT
 SW(17) = TWLO
 SW(18) = TOLTW
 SW(19) = THLOI
 SV(2) = IDVEL


```

SV(36) = 7.1
SV(29)=COPIES
DO 15 I = 1,3
15 SQ(1,I) = PRNTX(I)
SQ(2,1) = FSEC
SQ(10,1) = WOREQ
SQ(10,3) = DRNG
SQ(16,1) = GWREQ
SQ(19,1)=0.
SQ(19,2)=0.
IF(FBFUEL.EQ.1.) SQ(19,3) = SW(11)/SW(12) * SW(14)/1.689
SQ(13,1) = WPREQ
SQ(19,5) = FLYBCK
SQ(32,1) = FBFUEL
SQ(32,2) = CA
SQ(32,3) = CR
SQ(32,4) = WFLYX
SQ(32,5) = PT
SQ(33,1) = R1
SQ(33,2) = R3
SQ(33,3) = SFC1
SQ(33,4) = SFC2
SQ(33,5) = SFC3
SQ(34,1) = ALD1
SQ(34,2) = ALD2
SQ(34,3) = ALD3
SQ(34,4) = VFLY1
SQ(34,5) = VFLY2
SQ(35,1) = VFLY3
IF(FBFUEL.EQ.2. .OR. FBFUEL.EQ.3.) SQ(19,3) = SQ(34,5)/1.689 *
1 SQ(34,2) / SQ(33,4)
T027(3) = SE(15)
T027(4) = SE(15)
T027(3) = (1.-SW(7)) * SE(3) + SW(7) * SE(11)
C ADJUST MASS RATIO FOR ORBITER
400 T034(3) = EXP( (SV(2) - T027(3) * 32.174049 * ALOG(TB34(3) ) ) /
1 (T027(3) * 32.174049) )
C
C FIXED SOLID STRAPON SYNTHESIS OPTION *****
SQ(20,1) = SOLID
SQ(20,2) = AS
SQ(20,3) = BS
SQ(20,4) = SISP
SQ(20,5)=SINERT
SQ(21,1) = SAE
SQ(21,2) = TSBO
IF(SOLID.LE.0.) RETURN
SQ(21,3) = (SOLID/SISP)*TSBO*(AS+.5*BS*TSBO)
SQ(21,4) =(AS - 14.69 * SAE ) * SOLID
SQ(21,5)=SINERT*SOLID+SQ(21,3)
SQ(22,1) = SQ(21,3) / SOLID
SQ(22,2) = SQ(21,4) / SOLID

```

GDC-08870-002

C
C
FIXED SOLID STRAPON SYNTHESIS OPTION

RETURN
END

SUBROUTINE TRAJA

REAL

1 LAT, LNG, ITT1, ITT2, ITT3, LAM, MULT, JETW, LAMC,
 2 LAMDT, LTW, LTX, LTY, LTZ

DIMENSION

1 TWD11(15), TWD12(15), TWD13(15), AREF1(15), AREF2(15), AREF3(15),
 2 STEP(15), TOL1(15), TOL2(15), TOL3(15), TOL4(15), TOL5(15),
 3 TOL6(15), TOL7(15), HTL1(15), HTL2(15), HTL3(15), HTL4(15),
 4 HTL5(15), HTL6(15), HTL7(15), STGC(15), STGV(15), STGT(15),
 5 STGD(15), JETW(15), ATMC(15), AC1(15), AC2(15), AC3(15),
 6 TWC(15), SIGC(15), ALPC(15), LAMC(15), PSIDT(15), SIGOT(15),
 7 ALPDT(15), LAMDT(15), HMIN(15), HCOEF(15), BCKT(15), RCKTT(15),
 8 SPSI(15), SSIG(15), SALP(15), SLAM(15), XKA1(1,1), XKA2(2,5),
 9 XKA3(2,5), TWD1(15), TWD2(15), TWD3(15), TWD4(15), TWD5(2),
 1 TWD6(15), TWD10(15), GDOT(15), XOUT(15), CVC1(4), CVC2(4),
 2 CVC3(4), CVN1(4), CVN2(4), CVN3(4), ECC1(4), ECC2(4),
 3 ECC3(4), ECN1(4), ECN2(4), ECN3(4), DCV1(4), DCV2(4),
 4 CVM1(4), CVM2(4), CVM3(4), CVL1(4), CVL2(4),
 5 CVL3(4), ECT1(4), ECT2(4), ECT3(4), EC1(4), EC2(4),
 6 EC3(4), GIS(4), ITT1(4), ITT2(4), ITT3(4), CRGF(4),
 7 TLAT(4), TLNG(4), TOL8(15), HTL8(15), ORB(15), GMC1(15),
 8 GMC2(15), GMC3(15), GMC4(15), XKPT(4), PITX(15), PSEC(15),
 9 PTME(15), PRIF(15), TWD14(15), VSET(15), VINC(15), WOUT(15),
 1 RDALT(4)

COMMON/APV/ALT(200),PRS(200),VLS(200)

COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
 1 VO(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
 2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
 3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
 4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SF(40),
 5 PS(15)

COMMON/DATAX/ HANM,HPNM,TRANOM ,

1 RAD, VEL, GAM, AZM, LAT, LNG, WGH, ALTO, SEC,
 2 PLD, CVC1, CVC2, CVC3, CVN1, CVN2, CVN3, ECC1, ECC2,
 3 ECC3, ECN1, ECN2, ECN3, DCV1, DCV2, DCV3, CVM1, CVM2,
 4 CVM3, CVL1, CVL2, CVL3, ECT1, ECT2, ECT3, GO, CK,
 5 WE, D2, D3, D4, AGO, ATQ, PO, CNV1, CNV2,
 6 CNV3, ALTO, STPF, A, B, RDALT, TO, EC1, EC2, EC3,
 7 GIS, ZOUT, ITT1, ITT2, ITT3, PSI, SIG, ALP, LAM,
 8 MULT, VELO, PRNT, PSIF, CRGF, TLAT, TLNG, ALTF, AZMF,
 9 TWE, TWD11, TWD12, TWD13, AREF1, AREF2, AREF3, STEP, TOL1,
 1 TOL2, TOL3, TOL4, TOL5, TOL6, TOL7, HTL1, HTL2, HTL3,
 2 HTL4, HTL5, HTL6, HTL7, STGC, STGV, STGT, STGD, JETW,
 3 ATMC, AC1, AC2, AC3, TWC, SIGC, ALPC, LAMC, PSIDT,
 4 SIGOT, ALPDT, LAMDT, HMIN, HCOEF, BCKT, RCKTT, SPSI, SSIG,
 5 SALP, SLAM, YOUT, CNV4, XKA0, XKA1, VSET, VINC, WOUT,
 6 XKA2, XKA3, PFCN, AZMI, GTIP, TWD1, TWD2, TWD3, TWD4,
 7 TWD5, TWD6, TWD14, PRIF, PTME, TWD10, VGAI, GDOT, XOUT,
 8 TOL8, HTL8, ORB, GMC1, GMC2, GMC3, GMC4, XKPT, PITX,
 9 PSEC

NAMELIST/DATA1/HANM,HPNM,TRANOM,

1 RAD, VEL, GAM, AZM, LAT, LNG, WGH, ALTO, SEC,

```

2 PLD, CVC1, CVC2, CVC3, CVN1, CVN2, CVN3, FCC1, ECC2,
3 ECC3, ECN1, ECN2, ECN3, DCV1, DCV2, DCV3, CVM1, CVM2,
4 CVM3, CVL1, CVL2, CVL3, ECT1, ECT2, ECT3, GO, CK,
5 WE, D2, D3, D4, AGQ, ATQ, PO, CNV1, CNV2,
6 CNV3, ALTQ, STPF, A, B, ROALT, TO, EC1, EC2, EC3,
7 GIS, ZOUT, ITT1, ITT2, ITT3, PSI, SIG, ALP, LAM,
8 MULT, VELQ, PRNT, PSIF, CRGF, TLAT, TLNG, ALTF, AZMF,
9 TWE, TWD11, TWD12, TWD13, AREF1, AREF2, AREF3, STEP, TOL1,
1 TOL2, TOL3, TOL4, TOL5, TOL6, TOL7, HTL1, HTL2, HTL3,
2 HTL4, HTL5, HTL6, HTL7, STGC, STGV, STGT, STGD, JETW,
3 ATMC, AC1, AC2, AC3, TWC, SIGC, ALPC, LAMC, PSIDT,
4 SIGDT, ALPDT, LAMDT, HMIN, HCOEF, BCKT, BCKTT, SPSI, SSIG,
5 SALP, SLAM, ALT, PRS, VLS, YOUT, CNV4, XM1, A1,
6 C1, XM2, A2, C2, XM3, A3, C3, XKA0, XKA1,
7 XKA2, XKA3, RFCN, AZMI, GTIP, TWD1, TWD2, TWD3, TWD4,
8 TWD5, TWD6, TWD7, TWD8, TWD9, TWD10, VGAI, GDOT, XOUT,
9 TOL8, HTL8, ORB, GMC1, GMC2, GMC3, GMC4, XKPT, PITX,
1 PSEC, PTME, PRIF, TWD14, PT, PDT, VSET, VINC, WOUT
1002 FORMAT(1H1,41X,35H ***** INITIAL CONDITIONS *****//5X,2H A,18X
1,4H AGQ,16X,4H ALP,16X,4H ATQ,16X,5H ALTF,15X,5H ALTQ/5X,5H ALTO,1
25X,4H AZM,16X,5H AZMF,15X,5H AZMI,15X,2H B,18X,3H CK/5X,5H CNV1,15
3X,5H CNV2,15X,5H CNV3,15X,5H CNV4,15X,3H D2,17X,3H D3/5X,3H D4,17X
4,4H GAM,16X,5H GTIP,15X,3H GO,17X,4H LAM,16X,4H LAT//5X,4H LNG,16X
5,5H MULT,15X,4H PLD,16X,5H PRNT,15X,4H PSI,16X,5H PSIF/5X,3H PO,17
6X,4H RAD,16X,5H RFCN,15X,4H SEC,16X,4H SIG,16X,5H STPF/5X,4H TWE,1
76X,3H TO,17X,4H VEL,16X,5H VELQ,15X,5H VGAI,15X,3H WE/5X,4H WGN,16
8X,5H YOUT,15X,5H ZOUT////4(6E20.8//)/3(6E20.8//),3E20.8////)
1003 FORMAT(1H1,44X,29H ***** SECTION DATA *****//5X,4H AC1,16X,4H
1AC2,16X,4H AC3,16X,5H ALPC,15X,6H ALPDT,14X,6H AREF1/5X,6H AREF2,
214X,6H AREF3,14X,5H ATMC,15X,5H BCKT,15X,6H BCKTT,14X,5H GDOT/5X,
36H GMC1,14X,6H GMC2,14X,6H GMC3,14X,6H GMC4,14X,6H HCOEF,14X,5
4H HMIN/5X,5H HTL1,15X,5H HTL2,15X,5H HTL3,15X,5H HTL4,15X,5H HTL5,
515X,5H HTL6/5X,5H HTL7,15X,5H HTL8,15X,5H JETW,15X,5H LAMC,15X,6H
6 LAMDT,14X,4H ORB/5X,5H PITX,15X,6H PSIDT,14X,5H PRIF,15X,5H PSEC,
715X,5H PTME,15X,5H SALP//5X,5H SIGC,15X,6H SIGDT,14X,5H SLAM,15X,5
8H SPSI,15X,5H SSIG,15X,5H STEP/5X,5H STGC,15X,5H STGD,15X,5H STGT,
915X,5H STGV,15X,5H TOL1,15X,5H TOL2/5X,5H TOL3,15X,5H TOL4,15X,5H
1TOL5,15X,5H TOL6,15X,5H TOL7,15X,5H TOL8/5X,4H TWC,16X,5H TWD1,15X
2,5H TWD2,15X,5H TWD3,15X,5H TWD4,15X,5H TWD6/5X,6H TWD10,14X,6H TH
3D11,14X,6H TWD12,14X,6H TWD13,14X,6H TWD14,14X,5H VINC/5X,5H VSET,
415X,5H WOUT,15X,5H XOUT/)
1004 FORMAT(1X,I2,E17.8,5E20.8/5(6E20.8//)/5(6E20.8//),3E20.8/)
1005 FORMAT(6E20.8)
1006 FORMAT(20H1 ATMOSPHERE TABLE///8X,14H ALTITUDE (FT)//)
1007 FORMAT(1H0/8X,18H PRESSURE (LB/IN2)//)
1008 FORMAT(1H0/8X,27H VELOCITY OF SOUND (FT/SEC)//)
1009 FORMAT(1H1,)
1010 FORMAT(48X,39H ***** TARGET POINT REFERENCE *****//5X,8H CRGF(
11),12X,9H ROALT(1),11X,8H TLAT(1),12X,8H TLNG(1),12X,8H CRGF(2),12
2X,9H ROALT(2)/5X,8H TLAT(2),12X,8H TLNG(2),12X,8H CRGF(3),12X,9H R
3OALT(3),11X,8H TLAT(3),12X,8H TLNG(3)/5X,8H CRGF(4),12X,9H ROALT(
44),11X,8H TLAT(4),12X,8H TLNG(4)//2(6E20.8//),4E20.8//)

```

```

1011 FORMAT(1H0/17H      AERO(1) TABLE,I3//8X,12H MACH NUMBER//)
1012 FORMAT(1H0/8X,26H  YAW ANGLE OF ATTACK (DEG)//)
1013 FORMAT(1H0/8X,24H  AERODYNAMIC COEFFICIENT//)
1014 FORMAT(1H0/17H      AERO(2) TABLE,I3//8X,12H MACH NUMBER//)
1015 FORMAT(1H0/8X,28H  PITCH ANGLE OF ATTACK (DEG)//)
1016 FORMAT(1H0/17H      AERO(3) TABLE,I3//8X,12H MACH NUMBER//)
1017 FORMAT(1H0,)
1018 FORMAT(1H0/20H      PROPULSION TABLE,I3//8X,11H TIME (SEC)//)
1019 FORMAT(1H0/8X,34H  VACUUM THRUST FOR ONE ENGINE (LB)//)
1020 FORMAT(1H0/8X,41H  - TIME RATE OF CHANGE IN WEIGHT (LB/SEC)//)
1021 FORMAT(29X,32H ***** GENERAL ITERATION BLOCK,I3,19X,6H GIS =,F3.
10,8H *****//5X5H CVC1,15X,5H CVC2,15X,5H CVC3,15X,5H CVL1,15X,5H
2 CVL2,15X,5H CVL3/5X,5H CVM1,15X,5H CVM2,15X,5H CVM3,15X,5H CVN1,1
35X,5H CVN2,15X,5H CVN3/5X,5H DCV1,15X,5H DCV2,15X,5H DCV3,15X,4H E
4C1,16X,4H EC2,16X,4H EC3//5X,5H ECC1,15X,5H ECC2,15X,5H ECC3,15X,5
5H ECN1,15X,5H ECN2,15X,5H ECN3/5X,5H ECT1,15X,5H ECT2,15X,5H ECT3,
615X,5H ITT1,15X,5H ITT2,15X,5H ITT3///3(6E20.8//)2(6E20.8//)///)
1022 FORMAT(1H0/20H      PITCH RATE TABLE,I3//8X,11H TIME (SEC)//)
1023 FORMAT(1H0/8X,27H  PITCH RATE STEPS (DEG/SEC)//)
1025 FORMAT(1H0/49H      PITCH ANGLE OF ATTACK *** MACH NUMBER TABLE,I3
1//8X,12H MACH NUMBER//)
1026 FORMAT(1H0/8X,28H  PITCH ANGLE OF ATTACK (DEG)//)
      XMULT = W(37)
      DO 71 L=1,2
71 TWO9(1,L) = -10000.
      READ(5,DATA1)
      CALL TRAJC
      IF(SO(19,5).EQ.4.) ORB(3) = 2.
      V(1) = RAD
      V(2) = VEL
      V(3) = GAM
      V(4) = AZM
      V(5) = LAT
      V(6) = LNG
      V(7) = WGH
      V(8) = ALTO
      V(9) = SFC + .1
      V(10) = PLD
      V(35) = G0
      V(36) = CK
      V(37) = WE
      V(38) = D2
      V(39) = D3
      V(40) = D4
      V(41) = AGQ
      V(42) = ATQ
      V(43) = P0
      V(44) = CNV1
      V(45) = CNV2
      V(46) = CNV3
      V(47) = ALTQ
      V(48) = STPF + .1

```

```

V(49) = A
V(50) = B
IF(V(50).GT.V(49)-1..OR.V(50).LT.1.) V(50) = V(49)
V(51) = T0
V(56) = ZOUT + .1
V(60) = PSI
V(61) = SIG
V(62) = ALP
V(63) = LAM
V(64) = MULT + SIGN(.1,MULT)
V(65) = VELQ
V(66) = PRNT + .1
V(67) = PSIF + .1
V(71) = ALTF
V(72) = AZMF
V(73) = TWE
V(74) = RFCN + .1
V(75) = AZMI
V(76) = GTIP
V(77) = VGAI + .1
V(78) = YOUT + .1
V(79) = CNV4
LSEC = IFIX(V(9))
DO 90 K=1,LSEC
Q(1,K) = TWD11(K)
Q(2,K) = TWD12(K)
Q(3,K) = TWD13(K)
Q(4,K) = AREF1(K)
Q(5,K) = AREF2(K)
Q(6,K) = AREF3(K)
Q(7,K) = STEP(K)
Q(8,K) = TOL1(K)
Q(9,K) = TOL2(K)
Q(10,K) = TOL3(K)
Q(11,K) = TOL4(K)
Q(12,K) = TOL5(K)
Q(13,K) = TOL6(K)
Q(14,K) = TOL7(K)
Q(15,K) = HTL1(K)
Q(16,K) = HTL2(K)
Q(17,K) = HTL3(K)
Q(18,K) = HTL4(K)
Q(19,K) = HTL5(K)
Q(20,K) = HTL6(K)
Q(21,K) = HTL7(K)
Q(22,K) = STGC(K) + SIGN(.1,STGC(K))
Q(23,K) = STGV(K)
Q(24,K) = STGT(K)
Q(25,K) = STGD(K) + SIGN(.1,STGD(K))
Q(26,K) = JETW(K)
Q(27,K) = ATWC(K) + .1
Q(28,K) = AC1(K) + .1

```

```

Q(29,K) = AC2(K) + .1
Q(30,K) = AC3(K) + .1
Q(31,K) = TWC(K) + .1
Q(32,K) = SIGC(K) + .1
Q(33,K) = ALPC(K) + .1
Q(34,K) = LAMC(K) + .1
Q(35,K) = PSIDT(K)
Q(36,K) = SIGDT(K)
Q(37,K) = ALPDT(K)
Q(38,K) = LAMDT(K)
Q(39,K) = HMIN(K)
Q(40,K) = HCOEF(K)
Q(41,K) = BCKT(K)
Q(42,K) = BCKTT(K)
Q(43,K) = SPSI(K)
Q(44,K) = SSIG(K)
Q(45,K) = SALP(K)
Q(46,K) = SLAM(K)
Q(47,K) = TWD1(K) + .1
Q(48,K) = TWD2(K)
Q(49,K) = TWD3(K)
Q(50,K) = TWD4(K)
Q(51,K) = TWD10(K)
Q(52,K) = GDOT(K)
Q(53,K) = XOUT(K) + .1
Q(54,K) = TOLR(K)
Q(55,K) = HTLR(K)
Q(56,K) = ORB(K) + .1
Q(57,K) = GMC1(K)
Q(58,K) = GMC2(K)
Q(59,K) = GMC3(K)
Q(60,K) = GMC4(K)
Q(61,K) = TWD6(K)
Q(62,K) = PITX(K)
Q(63,K) = PSEC(K) + .1
Q(64,K) = PTME(K)
Q(65,K) = PRIF(K) + .1
Q(66,K) = TWD14(K)
Q(67,K) = VSET(K) + .1
Q(68,K) = VINC(K)
Q(69,K) = WOUT(K) + .1
90 CONTINUE
DO 60 L=1,4
VQ(28,L) = GIS(L) + .1
VQ(32,L) = CRGF(L) + .1
VQ(33,L) = TLAT(L)
VQ(34,L) = TLNG(L)
VQ(35,L) = RDALT(L)
IF(GIS(L).LT..5) GO TO 60
VQ(1,L) = CVC1(L) + .1
VQ(2,L) = CVC2(L) + .1
VQ(3,L) = CVC3(L) + .1

```

```

VQ(4,L) = CVN1(L) + .1
VQ(5,L) = CVN2(L) + .1
VQ(6,L) = CVN3(L) + .1
VQ(7,L) = ECC1(L) + SIGN(.1,ECC1(L))
VQ(8,L) = ECC2(L) + SIGN(.1,ECC2(L))
VQ(9,L) = ECC3(L) + SIGN(.1,ECC3(L))
VQ(10,L) = ECN1(L) + .1
VQ(11,L) = ECN2(L) + .1
VQ(12,L) = ECN3(L) + .1
VQ(13,L) = DCV1(L)
VQ(14,L) = DCV2(L)
VQ(15,L) = DCV3(L)
VQ(16,L) = CVM1(L)
VQ(17,L) = CVM2(L)
VQ(18,L) = CVM3(L)
VQ(19,L) = CVL1(L)
VQ(20,L) = CVL2(L)
VQ(21,L) = CVL3(L)
VQ(22,L) = ECT1(L)
VQ(23,L) = ECT2(L)
VQ(24,L) = ECT3(L)
VQ(25,L) = EC1(L)
VQ(26,L) = EC2(L)
VQ(27,L) = EC3(L)
VQ(29,L) = ITT1(L) + .1
VQ(30,L) = ITT2(L) + .1
VQ(31,L) = ITT3(L) + .1

```

```
60 CONTINUE
```

```

XNA0 = XKA0 + .1
DO 91 L=1,2
XNP(L) = TND5(L) + .1

```

```
91 CONTINUE
```

```

DO 56 L=1,4
XNPT(L) = XKPT(L) + .1

```

```
56 CONTINUE
```

```

XNA1(1,1) = XKA1(1,1) + .1
DO 54 L=1,5
DO 55 K=1,2
XNA2(K,L) = XKA2(K,L) + .1
XNA3(K,L) = XKA3(K,L) + .1

```

```
55 CONTINUE
```

```
54 CONTINUE
```

```
      CYCLE CHECK FOR AZIMUTH AND LONGITUDE
```

```

DO 75 K=1,LSEC
IF(Q(65,K).GT..5) GO TO 76

```

```
75 CONTINUE
```

```

DO 19 K=4,6,2
LQ = 25

```

```
80 LQ = LQ - 1
```

```

IF(V(K).LT.0.) GO TO 20
IF(V(K).LT.360.) GO TO 19
V(K) = V(K) - 360.

```



```

      GO TO 81
20  V(K) = V(K) + 360.
81  IF(LQ.GT.0) GO TO 80
19  CONTINUE
C    INITIAL ALTITUDE AND RADIUS MAGNITUDE
76  ARG5 = V(5)/V(45)
     SIN5 = SIN(ARG5)
     COS5 = COS(ARG5)
     IF(V(50).GT.V(49).OR.V(50).LT.1.) GO TO 21
     IF(ABS(V(5)).LT..00005) GO TO 21
     Z(14) = 1./SQRT((COS5/V(49))**2 + (SIN5/V(50))**2)
     GO TO 22
21  Z(14) = V(49)
22  IF(V(71).GT..5) GO TO 12
     V(1) = V(8) + Z(14)
     GO TO 13
12  V(8) = V(1) - Z(14)
C    INITIAL MODE INERTIAL OR RELATIVE VELOCITY, GAMMA, AND AZIMUTH
13  IF(V(77).LT..5) GO TO 40
     ARG3 = V(3)/V(45)
     SIN3 = SIN(ARG3)
     COS3 = COS(ARG3)
     IF(V(77).LT.1.5) GO TO 41
     IF(ABS(COS5).LT..000001) GO TO 42
     SIN4 = COS(V(4)/V(45))/COS5
     IF(ABS(SIN4).GT..999990) GO TO 43
     ARG4 = ASIN(SIN4)
     IF(V(77).GT.2.5) ARG4 = V(44) - ARG4
     IF(V(4).LT.0.) ARG4 = ARG4 + 2.*V(44)
     COS4 = COS(ARG4)
     GO TO 44
42  SIN4 = 1.
     COS4 = 0.
     IF(V(77).GT.2.5) SIN4 = -1.
     GO TO 44
43  COS4 = 0.
     IF(SIN4.LT.0.) GO TO 45
     SIN4 = 1.
     GO TO 44
45  SIN4 = -1.
     GO TO 44
41  ARG4 = V(4)/V(45)
     SIN4 = SIN(ARG4)
     COS4 = COS(ARG4)
44  VRX = V(2)*COS3*SIN4 - V(1)*V(37)*COS5/V(45)
     VRY = V(2)*COS3*COS4
     VRZ = V(2)*SIN3
     VARG = SQRT(VRX*VRX + VRY*VRY + VRZ*VRZ)
     IF(ABS(VARG).LT..000001) GO TO 46
     SARG = V(2)*SIN3/VARG
     IF(ABS(SARG).GT..999990) GO TO 47
     V(3) = V(45)*ASIN(SARG)

```

```

      CARG = COS(V(3)/V(45))
      GO TO 48
46  V(2) = VARG
      GO TO 40
47  V(2) = VARG
      V(4) = 0.
      IF(SARG.LT.0.) GO TO 49
      V(3) = 90.
      GO TO 40
49  V(3) = -90.
      GO TO 40
48  ARG = VARG*CARG
      IF(ABS(ARG).LT..000001) GO TO 51
      ARG = V(2)*COS3*COS4/ARG
      V(2) = VARG
      IF(ABS(ARG).GT..999990) GO TO 52
      V(4) = V(45)*ACOS(ARG)
      IF(VPX.LT.0.) V(4) = 360. - V(4)
      GO TO 40
51  V(2) = VARG
      V(4) = 0.
      GO TO 40
52  IF(ARG.LT.0.) GO TO 53
      V(4) = 0.
      GO TO 40
53  V(4) = 180.
      INITIAL PITCH ATTITUDE AND PITCH ANGLE OF ATTACK
40  IF(V(67).GT..5) GO TO 10
      V(60) = 90. - V(62) - V(3)
      GO TO 11
10  V(62) = 90. - V(60) - V(3)
      DOWN RANGE AND CROSS RANGE INITIAL SET-UP
11  Z(49) = 0.
      Z(50) = 0.
      Z(51) = 0.
      W(20) = SIN5
      W(21) = COS5
      ARG = V(6)/V(45)
      W(22) = SIN(ARG)
      W(23) = COS(ARG)
      DO 14 K=1,4
      IF(VQ(32,K).LT..5) GO TO 14
      **** TARGET RANGE ****
      ARG = VQ(33,K)/V(45)
      WQ(1,K) = SIN(ARG)
      WQ(2,K) = COS(ARG)
      WQ(10,K) = VQ(35,K) + 1./SQRT((WQ(2,K)/V(49))**2 + (WQ(1,K)/V(50))
1**2)
      ARG = VQ(34,K)/V(45)
      WQ(3,K) = SIN(ARG)
      WQ(4,K) = COS(ARG)
      ARG1 = W(21)*W(23)

```

```

ARG2 = W(21)*W(22)
ARG3 = WQ(2,K)*WQ(4,K)
ARG4 = WQ(2,K)*WQ(3,K)
WQ(5,K) = ARG1*ARG3 + ARG2*ARG4 + W(20)*WQ(1,K)
IF(ABS(WQ(5,K)).GT..9999999) GO TO 15
WQ(7,K) = V(45)*ACOS(WQ(5,K))
LTZ = ARG1*ARG4 - ARG3*ARG2
DUM = V(4) - 180.
IF(DUM*LTZ.GT.0.) WQ(7,K) = 360. - WQ(7,K)
WQ(6,K) = SIN(WQ(7,K)/V(45))
GO TO 26
15 IF(WQ(5,K)) 17,17,18
17 WQ(7,K) = 180.
GO TO 16
18 WQ(7,K) = 0.
16 WQ(6,K) = 0.
LTZ = 0.
26 IF(VQ(32,K).GT.3.5) GO TO 27
IVAR = IFIX(VQ(32,K))
VQ(IVAR+24,K) = WQ(7,K)
27 LTX = ARG2*WQ(1,K) - ARG4*W(20)
LTY = ARG3*W(20) - ARG1*WQ(1,K)
LTW = LTX*W(22) - LTY*W(23)
XLT = SQRT(LTX*LTX + LTY*LTY + LTZ*LTZ)
IF(XLT.LT..0000001) GO TO 28
ARG = LTW/XLT
IF(ARG.GT..9999999) GO TO 34
WQ(8,K) = V(45)*ACOS(ARG)
WQ(9,K) = WQ(8,K)
IF(LTY.LT.0.) WQ(9,K) = 360. - WQ(9,K)
GO TO 14
28 WQ(8,K) = V(4)
WQ(9,K) = V(4)
GO TO 14
34 IF(LTW.LT.0.) GO TO 35
WQ(8,K) = 0.
GO TO 36
35 WQ(8,K) = 180.
36 WQ(9,K) = WQ(8,K)
14 CONTINUE
IF(V(72).LT..5) GO TO 23
C ***** AUTOMATIC INITIAL AZIMUTH *****
LQL = IFIX(V(72))
LQK = LQL
IF(V(72).GT.4.5) LQK = LQK - 4
ARG = (VQ(34,LQK) + V(37)*V(73))/V(45)
SINT = SIN(ARG)
COST = COS(ARG)
ARG3 = WQ(2,LQK)*COST
ARG4 = WQ(2,LQK)*SINT
LTX = ARG2*WQ(1,LQK) - ARG4*W(21)
LTY = ARG3*W(20) - ARG1*WQ(1,LQK)

```

```

LTZ = ARG1*ARG4 - ARG3*ARG2
LTW = LTX*W(22) - LTY*W(23)
XLT = SQRT(LTX*LTX + LTY*LTY + LTZ*LTZ)
V(4) = V(45)*ACOS(LTW/XLT)
IF(V(72).GT.4.5) GO TO 24
DC = 1.
GO TO 25
24 DC = -1.
25 IF(DC*LTZ.LT.0.) V(4) = 360. - V(4)
23 IF(V(56).LT..5) GO TO 30
C INERTIAL COORDINATE SYSTEM ORIENTATION ANGLES FOR GEMBET AND TS
IF(V(76).LT.900.) GO TO 31
IF(ABS(V(5)).GT.99.995) GO TO 32
ARG = V(49)/V(50)
ARG = ARG*ARG*TAN(V(5)/V(45))
V(76) = V(45)*ATAN(ARG) - V(5)
31 ARG = (90. - V(5) - V(76))/V(45)
W(25) = SIN(ARG)
W(26) = COS(ARG)
GO TO 33
32 W(25) = 0.
W(26) = SIGN(1.,V(5))
33 W(20) = V(75)
C **** WRITE INITIAL CONDITIONS ****
30 IF(SQ(1,1).EQ.0.) GO TO 67
WRITE(6,1002) A, AGQ, ALP, ATG, ATF, ALTQ, ALTO,
1 AZM, AZMF, AZMT, B, CX, CNV, CNV2, CNV3, CNV4,
2 D2, D3, D4, GAM, GTIP, GO, LAN, LAT, LNG,
3 MULT, PLO, PRNT, PSI, PSIF, PC, RAD, PFCN, SEC,
4 STG, STPF, TWE, TO, VFL, VELO, VGAI, WE, WGH,
5 YOUT, ZOUT
WRITE(6,1010) (CRGF(K), RDALT(K), TLAT(K), TLNG(K), K=1,4)
IOUT = 1
DO 69 L=1,4
IF(V(28,L).LT..5) GO TO 69
IF(IOUT.EQ.2) WRITE(6,1009)
WRITE(6,1021) L, GIS(L), CVC1(L), CVC2(L), CVC3(L),
1 CVL1(L), CVL2(L), CVL3(L), CVM1(L), CVM2(L), CVM3(L),
2 CVN1(L), CVN2(L), CVN3(L), DCV1(L), DCV2(L), DCV3(L),
3 EC1(L), EC2(L), EC3(L), ECC1(L), ECC2(L), ECC3(L),
4 ECN1(L), ECN2(L), ECN3(L), ECT1(L), ECT2(L), ECT3(L),
5 ITT1(L), ITT2(L), ITT3(L)
IOUT = IOUT + 1
69 CONTINUE
C **** WRITE SECTION DATA ****
WRITE(6,1003)
DO 92 L=1,LSEC
IF(MOD(L,4).EQ.0) WRITE(6,1009)
92 WRITE(6,1004) L, AC1(L), AC2(L), AC3(L), ALPC(L),
1 ALPT(L), AREF1(L), AREF2(L), AREF3(L), ATMC(L), BCKT(L),
2 BCKTT(L), GDOT(L), GMC1(L), GMC2(L), GMC3(L), GMC4(L),
3 HCOEF(L), HMIN(L), HTL1(L), HTL2(L), HTL3(L), HTL4(L),

```

```

4 HTL5(L), HTL6(L), HTL7(L), HTL8(L), JETW(L), LANC(L),
5 LAMNT(L), ORB(L), PITX(L), PSIDT(L), PRIF(L), PSEC(L),
6 PTME(L), SALP(L), SIGC(L), SIGDT(L), SLAM(L), SPSI(L),
7 SSIG(L), STEP(L), STGC(L), S,GN(L), STGT(L), STGV(L),
8 TOL1(L), TOL2(L), TOL3(L), TOL4(L), TOL5(L), TOL6(L),
9 TOL7(L), TOL8(L), TWC(L), TWD1(L), TWD2(L), TWD3(L),
1 TWD4(L), TWD6(L), TWD10(L), TWD11(L), TWD12(L), TWD13(L),
2 TWD14(L), VINC(L), VSET(L), WOUT(L), XOUT(L)
ISEND = 1 + IFIX(V(78))
GO TO (63,64,65,66,67,68,67), ISEND
64 IF(XMULT-.5) 63, 63, 66
65 IF(XMULT-.5) 63, 63, 67
67 IF(XNA0.LT..5) GO TO 97
C **** WRITE ATMOSPHERE TABLE ****
  ISTOP = IFIX(XNA0)
  DO 89 L=1,LSEC
    IF(Q(27,L).LT..5) GO TO 89
    GO TO 88
89 CONTINUE
  GO TO 97
88 WRITE(6,1006)
  WRITE(6,1005) (ALT(I),I=1,ISTOP)
  WRITE(6,1007)
  WRITE(6,1005) (PRS(I),I=1,ISTOP)
  WRITE(6,1008)
  WRITE(6,1005) (VLS(I),I=1,ISTOP)
  WRITE(6,1009)
  GO TO 66
68 IF(XMULT-.5) 66, 66, 67
66 IF(XNA0.LT..5) GO TO 97
C **** WRITE AERO1 TABLE ****
  DO 96 L=1,1
    IF(XNA1(1,L).LT..5) GO TO 96
    ISTOP = IFIX(XNA1(1,L))
    KSTOP = IFIX(XNA1(2,L))
    WRITE(6,1011) L
    WRITE(6,1005) (XM1(I,L),I=1,ISTOP)
    WRITE(6,1012)
    WRITE(6,1005) (A1(K,L),K=1,KSTOP)
    WRITE(6,1013)
    DO 87 K=1,KSTOP
      WRITE(6,1005) (C1(I,K,L),I=1,ISTOP)
87 WRITE(6,1017)
96 CONTINUE
C **** WRITE AERO2 TABLE ****
  DO 95 L=1,5
    IF(XNA2(1,L).LT..5) GO TO 95
    ISTOP = IFIX(XNA2(1,L))
    KSTOP = IFIX(XNA2(2,L))
    WRITE(6,1014) L
    WRITE(6,1005) (XM2(I,L),I=1,ISTOP)
    WRITE(6,1015)

```

```

WRITE(6,1005) (A2(K,L),K=1,KSTOP)
WRITE(6,1013)
DO 86 K=1,KSTOP
WRITE(6,1005) (C2(I,K,L),I=1,ISTOP)
86 WRITE(6,1017)
95 CONTINUE
C ***** WRITE AERO3 TABLE *****
DO 94 L=1,5
IF(XNA3(1,L).LT..5) GO TO 94
ISTOP = IFIX(XNA3(1,L))
KSTOP = IFIX(XNA3(2,L))
WRITE(6,1016) L
WRITE(6,1005) (XM3(I,L),I=1,ISTOP)
WRITE(6,1015)
WRITE(6,1005) (A3(K,L),K=1,KSTOP)
WRITE(6,1013)
DO 85 K=1,KSTOP
WRITE(6,1005) (C3(I,K,L),I=1,ISTOP)
85 WRITE(6,1017)
94 CONTINUE
C ***** WRITE PRPSN TABLE *****
97 DO 70 L=1,2
IF(XNP(L).LT..5) GO TO 70
ISTOP = IFIX(XNP(L))
WRITE(6,1018) L
WRITE(6,1005) (TWD7(I,L),I=1,ISTOP)
WRITE(6,1019)
WRITE(6,1005) (TWD8(I,L),J=1,ISTOP)
IF(TWD9(1,L).LT.-.0000.) GO TO 70
WRITE(6,1020)
WRITE(6,1005) (TWD9(I,L),I=1,ISTOP)
70 CONTINUE
C ***** WRITE PITCH RATE TABLE *****
DO 72 L=1,4
IF(XNPT(L).LT.1.5) GO TO 72
DO 73 K=1,LSEC
ISEND = IFIX(Q(73,K))
GO TO (73,73,73,73,73,73,73,74,74,74,74,74,74,74,74,74,74,74,73,73,73,
1,73,77,77,77,77,77,77,77,77,77,77), ISEND
74 IISEND = (ISEND-5)/2
IF(IISEND-L) 73, 78, 73
78 ISTOP = IFIX(XNPT(L))
KSTOP = ISTOP - 1
WRITE(6,1022) L
WRITE(6,1005) (PT(I,L),I=1,ISTOP)
WRITE(6,1023)
WRITE(6,1005) (POT(I,L),I=1,KSTOP)
GO TO 72
77 IISEND = (ISEND-20)/2
IF(IISEND-L) 73, 79, 73
79 ISTOP = IFIX(XNPT(L))
WRITE(6,1025) L

```

GDC-08870-002

```
WRITE(6,1005) (PT(I,L), I=1,ISTOP)
WRITE(6,1026)
WRITE(6,1005) (PDT(I,L), I=1,ISTOP)
GO TO 72
73 CONTINUE
72 CONTINUE
67 MULT = 0.
  XMULT = 1.
  W(37) = XMULT
  RETURN
END
```

PRECEDING PAGE BLANK NOT FILMED

Appendix IV-10 Subroutine TRAJC

GDC-09870-002

SUBROUTINE TRAJC

REAL

```

1 LAT, LNG, ITT1, ITT2, ITT3, LAM, MULT, JETW, LAMC,
2 LAMDT
COMMON NSEC,T,TP,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
DIMENSION
1 TWD11(15), TWD12(15), TWD13(15), AREF1(15), AREF2(15), AREF3(15),
2 STEP(15), TOL1(15), TOL2(15), TOL3(15), TOL4(15), TOL5(15),
3 TOL6(15), TOL7(15), HTL1(15), HTL2(15), HTL3(15), HTL4(15),
4 HTL5(15), HTL6(15), HTL7(15), STGC(15), STGV(15), STGT(15),
5 STGD(15), JETW(15), ATMC(15), AC1(15), AC2(15), AC3(15),
6 TWC(15), STGC(15), ALPC(15), LAMC(15), PSIDT(15), SIGDT(15),
7 ALPDT(15), LAMDT(15), HMIN(15), HCOEF(15), BCKT(15), BCKTT(15),
8 SPSI(15), SSTG(15), SALP(15), SLAM(15), XKA1(1,1), XKA2(2,5),
9 XKA3(2,5), TWD1(15), TWD2(15), TWD3(15), TWD4(15), TWD5(2),
1 TWD6(15), TWD10(15), GDOT(15), XOUT(15), CVC1(4), CVC2(4),
2 CVC3(4), CVN1(4), CVN2(4), CVN3(4), ECC1(4), ECC2(4),
3 ECC3(4), ECN1(4), ECN2(4), ECN3(4), DCV1(4), DCV2(4),
4 DCV3(4), CVM1(4), CVM2(4), CVM3(4), CVL1(4), CVL2(4),
5 CVL3(4), ECT1(4), ECT2(4), ECT3(4), EC1(4), EC2(4),
6 EC3(4), GIS(4), ITT1(4), ITT2(4), ITT3(4), CRGF(4),
7 TLAT(4), TLNG(4), TOLA(15), HTL8(15), ORR(15), GMC1(15),
8 GMC2(15), GMC3(15), GMC4(15), XKPT(4), PITX(15), PSEC(15),
9 PTME(15), PRIF(15), TWD14(15), VSET(15), VINC(15), WOUT(15),
1 RDALT(4)
DIMENSION IVACO(7),ISLO(7),TFCTRO(7),TFCTRB(4),IVACB(4),ISLB(4)
1 ,PRNTX(3)
COMMON/DATA1/ HANM,HPNM,TRANOM ,
1 RAD, VEL, GAM, AZM, LAT, LNG, WGH, ALTO, SEC,
2 PLD, CVC1, CVC2, CVC3, CVN1, CVN2, CVN3, ECC1, ECC2,
3 ECC3, ECN1, ECN2, ECN3, DCV1, DCV2, DCV3, CVM1, CVM2,
4 CVM3, CVL1, CVL2, CVL3, ECT1, ECT2, ECT3, GO, CK,
5 WF, D2, D3, D4, AGQ, ATQ, PO, CNV1, CNV2,
6 CNV3, ALTQ, STPF, A, R, RDALT, TO, EC1, EC2, EC3,
7 GIS, ZOUT, ITT1, ITT2, ITT3, PSI, STG, ALP, LAM,
8 MULT, VELO, PRNT, PSIF, CRGF, TLAT, TLNG, ALTF, AZMF,
9 TWE, TWD11, TWD12, TWD13, AREF1, AREF2, AREF3, STEP, TOL1,
1 TOL2, TOL3, TOL4, TOL5, TOL6, TOL7, HTL1, HTL2, HTL3,
2 HTL4, HTL5, HTL6, HTL7, STGC, STGV, STGT, STGD, JETW,
3 ATMC, AC1, AC2, AC3, TWC, SIGC, ALPC, LAMC, PSIDT,
4 SIGDT, ALPDT, LAMDT, HMIN, HCOFF, BCKT, BCKTT, SPSI, SSIG,
5 SALP, SLAM, YOUT, CNV4, XKA0, XKA1, VSET, VINC, WOUT,
6 XKA2, XKA3, PFCN, AZMI, GTIP, TWD1, TWD2, TWD3, TWD4,
7 TWD5, TWD6, TWD14, PRIF, PTME, TWD10, VGAI, GDOT, XOUT,
8 TOL8, HTL8, ORR, GMC1, GMC2, GMC3, GMC4, XKPT, PITX,
9 PSEC
COMMON/DATA2X/ ALD,FBPAP,IOVEL,ISLB,ISLO,IVACB,IVACO,PERISP,QMAX,

```


GDC-08870-002

1 QMXS,SFC,SLVOUT,COPIES,SYNIT,TFCTPB,TFCTRO,TOLMU,TOLTW,
2 TPATIO,TWLO,TWLOI,WTOUT,FIRE,ROOTW,VCRUSE
3 ,NXFOB,PRNTX,FSEC
4 ,SOLID,AS,RS,SISP,SINERT,SAE,TSBO
6 ,FRFUEL,CA,CB,WFLYX,PT,R1,R3,SFC1,SFC2,SFC3,ALD1,ALD2,ALD3,VFLY1,
7 VFLY2,VFLY3

RFCN = 7.

CVC1(1)=62.

CVN1(1)= 201.

OCV1(1)=.01

CVN1(2)= 5.

ECC1(2)=8.

ECC2(2)=42.

ECN1(1)= 3.

ECN1(2)= 7.

ECN2(2)=7.

GIS (1)=1.

GTS (2)=4.

IF(SEC.LT.7.) SEC = 7.

ALPC(1)=8.

ALPC(2)=8.

ALPC(3)=1.

ALPC(4)=1.

SALP(7) = 0.

TWC (1)=6.

TWC(2)=6.

TWC (3)=0.

TWC(4)=0.

TWC (5)=6.

TWC(6)=6.

TWC(7)=6.

STGC(2)=7.

STGC(3)=0.

STGC(4)=0.

STGC(5)=0.

STGC(7)=41.

STGT(2)=1.

STGT(3)=.001

STGT(4)=.001

STGT(5)=.001

STGT(7)=1.

STGD(2)=-1.

STGD(3)=1.

STGD(4)=1.

STGD(5)=1.

STGD(7)=1.

PLD = 0.

SQ(3,1)= 0.

OPTION FOR TARGETING ORBITER TO APOGEE,PERIGEE, TRUE ANOMALY

IF(HANM.LE.0.) GO TO 1

```

IF(ORB(7).LT.1.) ORB(7)=1.
ECC1(2)=101.
ECC2(2)=102.
STGC(7)=103.
EC1(2)=0.
EC2(2)=0.
STGV(7)=0.
SQ(3,1) = HANM * CNV3
SQ(3,2) = HPNM * CNV3
SQ(3,3) = TRANOM/CNV2
SQ(3,4) = COS(SQ(3,3))
SQ(3,5) = SIN(SQ(3,3))

```

```

1 CONTINUE

```

```

      OPTION FOR TARGETING ORBITER TO APOGEE, PERIGEE, TRUE ANOMALY

```

```

C
C
C
C

```

```

FIXED SOLID STRAPON SYNTHESIS OPTION *****
IF(SOLID.LE.0.) RETURN
STGV(1) = TSBO
STGC(1) = 0.
STGD(1) = 1.
STGT(1) = .001
JETW(1)= SINERT * SOLID

```

```

C
C

```

```

FIXED SOLID STRAPON SYNTHESIS OPTION *****

```

```

RETURN
END

```

PRECEDING PAGE BLANK NOT FILMED

GDC-D8870-002

Appendix IV-11 Subroutine ITER8

```

SUBROUTINE ITER8
  DIMENSION SK0(30), SC0(300), T04(6), T020(10), T027(6), T034(6),
1  T048(10), T049(10), T050(10), T057(6)
  DIMENSION SKB(30), SCB(300), TB4(6), TB20(10), TB27(6), TB34(6),
1  TB48(10), TB49(10), TB50(10), TB57(6)
  COMMON NSEC, T, TR, TS(15), V(85), Q(73,15), W(50), Z(120), EC(3), CV(3),
1  VQ(38,4), WO(13,4), XM1(1,1), A1(1,1), C1(1,1,1), XM2(25,5), A2(12,5),
2  C2(25,12,5), XM3(25,5), A3(12,5), C3(25,12,5), XNA0, XNA1(1,1),
3  XNA2(2,5), XNA3(2,5), XNP(2), PST(25), TWD7(25,2), TWD8(25,2), SW(50),
4  TWD9(25,2), PT(25,4), PDT(25,4), XNPT(4), SV(50), SQ(40,5), SE(40),
5  PS(15)
  COMMON/ORRINX/
1  T01, T02, T03, T04, T05, T06, T07, T08, T09, T010, T011, T012, T013, T014,
2  T015, T016, T017, T018, T019, T020, T021, T022, T023, T024, T025, T026, T027,
3  T028, T029, T030, T031, T032, T033, T034, T035, T036, T037, T038, T039, T040,
4  T041, T042, T043, T044, T045, T046, T047, T048, T049, T050, T051, T052, T053,
5  T054, T055, T056, T057, SK0, SC0
  COMMON/ORBINY/
1  TB1, TB2, TB3, TB4, TB5, TB6, TB7, TB8, TB9, TB10, TB11, TB12, TB13, TB14,
2  TB15, TB16, TB17, TB18, TB19, TB20, TB21, TB22, TB23, TB24, TB25, TB26, TB27,
3  TB28, TB29, TB30, TB31, TB32, TB33, TB34, TB35, TB36, TB37, TB38, TB39, TB40,
4  TB41, TB42, TB43, TB44, TB45, TB46, TB47, TB48, TB49, TB50, TB51, TB52, TB53,
5  TB54, TB55, TB56, TB57, SKB, SCB
C  SW(2) = 2 FOR CONVERGED RUN
C  SW(2)=1. FOR LAST PASS
C  SW(2) =.5 FOR ITERATION
  IF(SE(40) .NE. 0.) SV(6) = (SV(7) + SE(40)) / SV(4)
  TOLER = ABS( T034(3) - SV(6) ) / SV(6)
  WRITE (6,1005) SV(6), T034(3), SW(3), SW(5), TOLER
  IF(ABS(SW(1)-3.1).LT..5) GO TO 500
  IF(SW(2).EQ.1.) GO TO 400
  SW(2) = .5
  IF(SW(3).LE.SW(4)) GO TO 70
  SW(2) = 1.
  SV(43) = 2.1
  WRITE (6,1020) SW(3)
70 IF( TOLER .LT. SW(5) .AND. SW(3) .GE. 2.) GO TO 10
  GO TO 11
10 SW(2) = 1.
  SV(43) = 2.1
11 T034(3) = SV(6)
  CALL RANGE
  RETURN
400 SW(2) = 2.
  IF(ABS(SQ(19,5)-5.) .GE. .5) GO TO 500
  SV(43) = 3.1
  CALL RANGE
  RETURN
500 IF( TOLER .GT. SW(5) ) WRITE(6,1000) SV(6), T034(3), SW(5), TOLER
  IF(SW(3).GT.SW(4)) WRITE(6,1021) SW(3), SW(5), TOLER
  CALL SUMOUT
  RETURN

```

GDC-DB870-002

1000 FORMAT (/26H FAILED TO CONVERGE - MU = 2E16.8,
1 5X,13H INPUT TOL = F8.5,5X,7H TOL = F9.6)
1005 FORMAT (/5X,10HGTSM MU = F10.6,5X,11HWTVOL MU = F10.6,F4.0,
1 5X,13H INPUT TOL = F8.5,5X,7H TOL = F9.6)
1020 FORMAT (/31H ITERATION COUNTER EXCEEDED = F4.0//)
1021 FORMAT (/31H ITERATION COUNTER EXCEEDED = F4.0,
1 5X,13H INPUT TOL = F8.5,5X,7H TOL = F9.6)
END

Appendix IV-12 Subroutine RANGE

SUBROUTINE RANGE

```

C
DIMENSION VSTBL(7), HSTBL(5), HANTBL(7,5), HAATBL(7,5),
2 VAATBL(7,5), VABTBL(7,5), RAATBL(7,5), PABTBL(7,5),
3 VATBL(7), HATBL(7), PSATBL(7,7),
4 HARTBL(7), PSITBL(4), HRATBL(7,4), HRBTPL(7,4)

C
COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)

C
DATA HSTBL/ 150000., 175000., 200000., 225000., 250000./

C
DATA VSTBL/ 7000., 8000., 9000., 10000., 11000., 12000., 13000./

C
DATA HANTBL/ 2.175, 2.240, 2.320, 2.410, 2.505, 2.6075, 2.7125,
2 2.115, 2.145, 2.190, 2.250, 2.325, 2.405, 2.485,
3 2.055, 2.073, 2.100, 2.125, 2.160, 2.200, 2.240,
4 2.035, 2.038, 2.042, 2.050, 2.060, 2.080, 2.105,
5 2.015, 2.020, 2.022, 2.025, 2.030, 2.032, 2.035/

C
DATA HAATBL/ 195., 200., 200., 197., 193., 187., 175.,
2 150., 183., 203., 215., 220., 220., 220.,
3 200., 250., 303., 353., 405., 460., 510.,
4 220., 290., 375., 470., 570., 670., 780.,
5 235., 320., 415., 520., 640., 790., 880./

C
DATA VAATBL/ 3.65, 6.00, 8.25, 10.60, 12.90, 15.40, 18.25,
2 1.70, 2.30, 3.20, 4.40, 5.90, 7.55, 9.30,
3 0.70, 1.15, 1.60, 2.35, 3.25, 4.45, 6.00,
4 -1.00, -1.40, -1.00, 0.02, 1.70, 2.70, 2.70,
5 -1.30, -1.80, -1.70, -1.25, -0.70, -0.45, -0.70/

C
DATA VABTBL/ -122.0, -166.0, -212.0, -260.0, -309.0, -360.0, -416.0,
2 -62.0, -70.0, -103.0, -133.0, -167.0, -202.0, -240.0,
3 -38.0, -47.0, -58.0, -73.0, -92.5, -119.0, -153.0,
4 -6.0, -10.0, -15.0, -22.0, -32.0, -46.0, -67.0,
5 5.0, 2.5, 0., -2.5, -6.0, -12.5, -22.5/

C
DATA RAATBL/ 0.05, 0.065, 0.10, 0.18, 0.33, 0.385, 0.40,
2 0.02, 0.08, 0.16, 0.26, 0.39, 0.54, 0.73,
3 0.07, 0.07, 0.07, 0.11, 0.15, 0.25, 0.52,
4 -0.05, 0.01, 0.04, 0.055, 0.06, 0.098, 0.201,
5 0.04, 0.03, 0.01, -0.018, -0.04, -0.05, 0.03/

C
DATA RABTBL/ 2.3, 2.8, 3.1, 2.8, 2.2, 2.5, 3.4,
2 3.7, 4.1, 4.2, 4.9, 5.2, 5.55, 5.8,
3 3.45, 4.85, 6.45, 8.3, 9.8, 10.9, 11.6,
4 5.2, 5.2, 6.3, 9.0, 11.8, 14.25, 16.8,

```

5 4.1, 5.9, 8.0, 10.5, 13.2, 16.1, 19.4/

DATA VATBL/ 7000., 8000., 9000., 10000., 11000., 12000., 13000./

DATA HATBL/ 180000., 200000., 220000., 240000., 260000., 280000.,
1 300000./

DATA PSATBL/ 90.0, 90.0, 89.0, 85.0, 83.0, 81.0, 79.5,
2 87.0, 80.5, 76.0, 73.5, 72.0, 71.0, 71.0,
3 81.5, 66.5, 63.0, 62.0, 61.5, 61.5, 62.0,
4 65.0, 58.5, 55.5, 54.0, 54.5, 56.0, 58.5,
5 35.0, 32.5, 32.0, 33.0, 35.0, 38.0, 41.5,
6 17.0, 16.0, 17.0, 19.0, 22.0, 25.5, 30.0,
7 0.0, 0.0, 1.5, 4.0, 8.0, 13.0, 19.0/

DATA PSITBL/ 0., 45., 60., 75./

DATA HAPTBL/150000., 175000., 200000., 225000.,
1 250000., 275000., 300000./

DATA HRATBL/ 7*16.0E-07,
1 7.985E-07, 7.950E-07, 7.915E-07, 7.955E-07,
1 7.994E-07, 8.000E-07, 8.040E-07,
2 3.204E-07, 4.100E-07, 4.636E-07, 3.900E-07,
2 5.176E-07, 5.350E-07, 5.541E-07,
3 0.297E-07, 1.150E-07, 1.969E-07, 2.600E-07,
3 3.131E-07, 3.550E-07, 3.958E-07/

DATA HRBTBL/ 0., 0.250E-02, 0.475E-02, 0.650E-02,
4 0.800E-02, 0.950E-02, 1.100E-02,
1 0.577E-02, 0.600E-02, 0.992E-02, 1.150E-02,
1 1.276E-02, 1.400E-02, 1.512E-02,
2 0.830E-02, 1.000E-02, 1.152E-02, 1.275E-02,
2 1.425E-02, 1.525E-02, 1.652E-02,
3 0.931E-02, 1.120E-02, 1.271E-02, 1.400E-02,
3 1.522E-02, 1.630E-02, 1.726E-02/

BANK ANGLE DURING ENTRY = PSI = SQ(10,2)
SQ(10,2) = 0.

ISND = IFIX(SQ(19,5)+.1)
GO TO (100,200,300,400,500), ISND

100 CONTINUE
DETAILED FLYBACK RANGE CALCULATIONS
VS = SV(8)
HS = SV(9)
IF(HS.GT.250000.) WRITE(6,1000)
IF(HS.GT.250000.) HS = 250000.
GAMS = SV(10)
IF(GAMS.GT.14.) WRITE(6,2000)
IF(GAMS.GT.14.) GAMS = 14.

```

C   DRS = SW(10)
C   CALL TRL2D(VS,HS,VSTBL,HSTBL,HANTBL,HAN)
C   CALL TRL2D(VS,HS,VSTBL,HSTBL,HAATBL,HAA)
C   HA = HAA*GAMS**HAN + HS
C   CALL TRL2D(VS,HS,VSTBL,HSTBL,VAATBL,VAA)
C   CALL TRL2D(VS,HS,VSTBL,HSTBL,VARTBL,VAB)
C   VA = VAA*GAMS**2. + VAB*GAMS + VS
C   CALL TRL2D(VS,HS,VSTBL,HSTBL,RAATBL,RAA)
C   CALL TRL2D(VS,HS,VSTBL,HSTBL,PARTBL,RAB)
C   RA = RAA*GAMS**2. + RAB*GAMS
C   IF(HA.GT.300000.) WRITE(6,3000)
C   IF(HA.GT.300000.) HA = 300000.
C   CALL TRL2D(VA,HA,VATBL,HATBL,PSATBL,PSI)
C   IF(PSI.GE.75.0) PSI = 75.0
C   SQ(10,2) = PSI
C   CALL TRL2D(HA,PSI,HARTBL,PSITBL,HRATBL,HRA)
C   CALL TRL2D(HA,PSI,HARTBL,PSITBL,HRBTBL,HRB)
C   RR = HRA*VA**2. + HRB*VA
C   R = 60.*DRS + PA + RP + SQ(10,3)
C   SQ(10,3) IS A FLYBACK ADDITIVE RANGE FACTOR TO ACCOUNT FOR THE
C   FINITE TIME REQUIRED TO ACCOMPLISH VARIOUS REENTRY MANEUVERS.
C   DRNG IS THE NUMONIC INPUT IN DATA2.
C   IF(SW(9).NE.1.) GO TO 10
C   SW(15) = R
C   RETURN
C   CORRECT FG RANGE PREDICTION FOR V6 CONFIGURATION
10  SW(15) = R*(0.7230/SW(9))**0.25
C   RETURN
C 200 CONTINUE
C   FLYBACK RANGE = FUNCTION OF STAGING Q ONLY
C   SPECIAL FLYBACK RANGE COMPUTATION FOR MSC
C   VQ(25,1) = LOG(10) Q STAGE
C   RNG = -55.915*VQ(25,1) + 430.0
C   SQ(10,3) IS A FLYBACK ADDITIVE RANGE FACTOR TO ACCOUNT FOR THE
C   FINITE TIME REQUIRED TO ACCOMPLISH VARIOUS REENTRY MANEUVERS.
C   DRNG IS THE MNEMONIC INPUT IN DATA2.
C   SW(15) = FLYBACK RANGE
C   SW(15) = RNG + SQ(10,3)

```

```

C
C
C      RETURN
C
C 300 CONTINUE
C      FLYBACK RANGE = CONSTANT = SQ(10,3) INPUT IN DATA?
C      SW(15) = SQ(10,3)
C      RETURN
C
C 400 CONTINUE
C      FLYBACK RANGE = IIP
C      SW(10) = CENTRAL ANGLE FROM LIFT OFF TO STAGING
C      SQ(12,1) = IIP CENTRAL ANGLE FROM STAGING TO IMPACT = Z(89)
C      SQ(10,3) = ADDITIVE RANGE FACTOR (DATA? INPUT)
C      SW(15) = 60.*(SQ(12,1) + SW(10)) + SQ(10,3)
C      RETURN
C
C 500 CONTINUE
C      **** FLYBACK TRAJECTORY TO BE NUMERICALLY INTEGRATED ****
C      SV(35) = 2.1
C      RETURN
C
C 1000 FORMAT(74H STAGING ALTITUDE GREATER THAN 250000 FT - FLYBACK RANGE
C          1 EQUATIONS INVALID)
C 2000 FORMAT(68H STAGING GAMMA GREATER THAN 14 DEG - FLYBACK RANGE EQUAT
C          1IONS INVALID)
C 3000 FORMAT(73H APOGEE ALTITUDE GREATER THAN 300000 FT - FLYBACK RANGE
C          1EQUATIONS INVALID)
C      END

```


Appendix IV-13 Subroutine TBL2D

```

C      SUBROUTINE TBL2D(X,Y,XTBL,YTBL,ZTBL,Z)
C
C      2 DIMENSIONAL TABLE LOOK-UP
C
      DIMENSION XTBL(7), YTBL(7), ZTBL(7,7)
      K = 0
      DO 1 II=2,7
      I=II
      IF(X-XTBL(I)) 3,2,1
1 CONTINUE
      GO TO 3
2 K = 1
      GO TO 4
3 SCAL1 = (X-XTBL(I-1))/(XTBL(I)-XTBL(I-1))
4 DO 5 JJ=2,7
      J=JJ
      IF(Y-YTBL(J)) 7,6,5
5 CONTINUE
      GO TO 7
6 SCAL2 = 1.
      GO TO 8
7 SCAL2 = (Y-YTBL(J-1))/(YTBL(J)-YTBL(J-1))
8 IF(K-1) 9,10,9
9 Z1 = ZTBL(I-1,J-1) + SCAL1*(ZTBL(I,J-1) - ZTBL(I-1,J-1))
  Z2 = ZTBL(I-1,J) + SCAL1*(ZTBL(I,J) - ZTBL(I-1,J))
  Z = Z1 + SCAL2*(Z2 - Z1)
  RETURN
10 Z = ZTBL(I,J-1) + SCAL2*(ZTBL(I,J) - ZTBL(I,J-1))
C
      RETURN
      END

```

PRECEDING PAGE BLANK NOT FILMED

GDC-06870-002

Appendix IV-14 Subroutine SUMOUT

```
SUBROUTINE SUMOUT
REAL LBODYO,LBODYR
DIMENSION SKO(30),SCO(300),TO4(6),TO20(10),TO27(6),TO34(6),
1 TO48(10),TO49(10),TO50(10),TO57(6)
DIMENSION SKR(30),SCR(300),TB4(6),TB20(10),TB27(6),TB34(6),
1 TB48(10),TB49(10),TB50(10),TB57(6)
COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),POT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
COMMON/ORRINX/
1 TO1,TO2,TO3,TO4,TO5,TO6,TO7,TO8,TO9,TO10,TO11,TO12,TO13,TO14,
2 TO15,TO16,TO17,TO18,TO19,TO20,TO21,TO22,TO23,TO24,TO25,TO26,TO27,
3 TO28,TO29,TO30,TO31,TO32,TO33,TO34,TO35,TO36,TO37,TO38,TO39,TO40,
4 TO41,TO42,TO43,TO44,TO45,TO46,TO47,TO48,TO49,TO50,TO51,TO52,TO53,
5 TO54,TO55,TO56,TO57,SKO,SCO
COMMON/ORRINY/
1 TB1,TB2,TB3,TB4,TB5,TB6,TB7,TB8,TB9,TB10,TB11,TB12,TB13,TB14,
2 TB15,TB16,TB17,TB18,TB19,TB20,TB21,TB22,TB23,TB24,TB25,TB26,TB27,
3 TB28,TB29,TB30,TB31,TB32,TB33,TB34,TB35,TB36,TB37,TB38,TB39,TB40,
4 TB41,TB42,TB43,TB44,TB45,TB46,TB47,TB48,TB49,TB50,TB51,TB52,TB53,
5 TB54,TB55,TB56,TB57,SKR,SCR
COMMON/TAMP/WGROSO,OTTOT,WFOUXO,TBTO,BTTOT
COMMON/SUMVW/WOP, WABFUB,WFOUXB,WDRYB,WGROSB,WOTHB,
1 VFUTKB,VOXTKB,VOTHB,VRODYB,VABFUB,
2 LBODYR,SBODYB,SPLANR,WOVRSB,
1 WPAYLO,WDRYO,WOTHO, WABFUB,
2 VFUTKO,VOXTKO,VCARO,VOTHO,VRODYO,
3 LBODYO,SBODYO,SPLANO,WOVRSO
4 ,WORATO,WORBTB,WRTRNO,WRTRNA,WENTRO,WENTRB,WLANDB,WLANDO
5 ,WCONTO,WCONTR
IFIRE = SE(5)
SQ(14,3) = SQ(14,2) - SQ(14,1)
FRANK = WFOUXO - WOP
TWORB = OTTOT/WGROSO
WORYT = WDRYB + WDRYO
WGROSV = WGROSB + WGROSO
VPROP0 = VFUTKO + VOXTKO
VPROPR = VFUTKB + VOXTKB
70 TVANO = OTTOT/TO36
TVAUO = TVANO * SE(38)
TVANR = BTTOT/TB36
TVAUR = TVANR * SE(37)
TSLNO = TVANO * SE(32)/SE(30)
TSLNR = TVANR * SE(31)/SE(29)
TSLUO = TVAUO * SE(36)/SE(34)
TSLUR = TVAUR * SE(35)/SE(33)
X = 1.
IF(SE(21).EQ.0..OR. IFIRE.EQ.2) X = 0.
TVAN = OTTOT * X + BTTOT
```

```

WDOTN = OTTOT * X / SE(30) + BTTOT / SE(29)
SPVAN = TVAN / WDOTN
TSLN = TSLNO * T036 * X + TSLNB * TB36
SPSLN = TSLN / WDOTN
TVAU = OTTOT * X * SE(34) + BTTOT * SE(37)
WDOTU = TVAU * X * T036 / SE(34) + TVAUB * TB36 / SE(33)
SPVAU = TVAU / WDOTU
TSLU = TSLUO * T036 * X + TSLUR * TB36
SPSLU = TSLU / WDOTU
WOPT = FRANK + SQ(37,4)
WOPTX = WFUOXO + SQ(37,4)
WOPTX = WFUOXB + SQ(37,5)
NCOPIE = SV(29) + .1
DO 900 I=1, NCOPIE
WRITE (6,1000)
CALL DATE
WRITE (6,1005)
WRITE (6,1010)
WRITE (6,1011) WFUOXB, FRANK
WRITE (6,1012) SQ(37,5), SQ(37,4)
WRITE (6,1015) WOPTX, WOPT
WRITE (6,1020) WOP
WRITE (6,1025) WOPTX, WOPTX
WRITE (6,1022) WAFUB, WAFUO
WRITE (6,1030) WPAYLO
WRITE (6,1035) WDRYB, WDRYO, WDRYT
WRITE (6,1041) WCONTB, WCONTO
WRITE (6,1040) WOTHB, WOTHO
WRITE (6,1045) WGROSB, WGROSO, WGROSV
WRITE (6,1170) WORRTO
WRITE (6,1175) WTRNB, WTRNO
WRITE (6,1180) WENTRB, WENTRO
WRITE (6,1185) WLANDB, WLANDO
WRITE (6,1050)
WRITE (6,1055) VFUTKB, VFUTKO
WRITE (6,1060) VOXTKB, VOXTKO
WRITE (6,1065) VPROPB, VPPOPO
WRITE (6,1070) VCARO
WRITE (6,1075) VOTHR, VOTHO
WRITE (6,1077) VRODYB, VBODYO
WRITE (6,1080)
WRITE (6,1085) LPODYB, LPODYO
WRITE (6,1090) SPODYB, SBODYO
WRITE (6,1095) SPLANB, SPLANO
WRITE (6,1096) TR45, T045
WRITE (6,1100) WOVPSB, WOVRSO
WRITE (6,1105)
WRITE (6,1110) TWORB, SV(13)
WRITE (6,1111) T036, T036
IF( SF(21) .EQ.0. .OR. IFIRE.EQ.2) GO TO 200
WRITE (6,1112) TSLNB, TSLUR, TSLNO, TSLUO, TSLN, TSLU
WRITE (6,1113) TVANB, TVAUB, TVANO, TVAUO, TVAN, TVAU

```

```

WRITE (6,1115) SE(31) , SE(35) , SE(32) , SE(36) , SPSLN, SPSLU
WRITE (6,1120) SE(29) , SE(33) , SE(30) , SE(34) , SPVAN,SPVAU
GO TO 220
C NO CROSSFEED
200 WRITE (6,2112) TSLNB,TSLUB,TSLNO,TSLN,TSLU
WRITE (6,2113) TVANB,TVAUB,TVANO,TVAN,TVAU
WRITE (6,2115) SE(31) , SE(35) , SE(32) , SPSLN,SPSLU
WRITE (6,2120) SE(29) , SE(33) , SE(30) , SPVAN,SPVAU
220 WRITE (6,1125)
WRITE (6,1126) TB34(3),SV(6)
WRITE (6,1128) SQ(14,1),SQ(14,3),SQ(14,2)
WRITE (6,1127) SE(9)
WRITE (6,1130) SV(12)
WRITE (6,1135) SV(8)
WRITE (6,1140) SV(9)
WRITE (6,1145) SV(10)
WRITE (6,1150) SQ(14,4)
WRITE (6,1155) SQ(14,5)
WRITE (6,1160) SQ(15,3)
WRITE (6,1165) SQ(15,4)
WRITE (6,1195) SW(15)
C
C FIXED SOLID STRAPON SYNTHESIS OPTION *****
IF(SQ(20,1).LE.0.) GO TO 900
WRITE (6,1200) SQ(20,1),SQ(22,1),SQ(20,5),SQ(21,5),SQ(21,1),
1 SQ(21,2) ,SQ(20,4),SQ(20,2),SQ(20,3),SQ(22,2)
C FIXED SOLID STRAPON SYNTHESIS OPTION *****
C
900 CONTINUE
WRITE (6,1000)
CALL DATE
WRITE (6,3000) SQ(36,1)
WRITE (6,3010) SQ(36,2)
WRITE (6,3015) SQ(36,3)
WRITE (6,3020) SQ(36,4)
WRITE (6,3025) SQ(36,5)
WRITE (6,3030) Q(62,1)
WRITE (6,3035) Q(45,5)
WRITE (6,3040) Q(35,5)
WRITE (6,3045) SQ(37,1)
WRITE (6,3046) SQ(37,2)
WRITE (6,3050)
IFLY = SQ(19,5)
GO TO (910,920,930,940,950),IFLY
910 WRITE (6,3055) SW(9),SQ(10,2)
GO TO 960
920 WRITE (6,3060) SV(12)
GO TO 960
930 WRITE (6,3065) SQ(10,3)
GO TO 960
940 WRITE (6,3070) SW(10),SQ(12,1)
GO TO 960

```

```

950 WRITE (6,3075) SV(42)
    WRITE (6,3080) V7(33,3)
    WRITE (6,3085) V7(34,3)
    WRITE (6,3090) 7(95)
    WRITE (6,3095) 7(8)
    WRITE (6,3100) 7(5)
    WRITE (6,3105) 7(6)
    WRITE (6,3110) 7(94)
960 IFUL = SQ(32,1)
    GO TO (965,970,975),IFUL
965 WRITE (6,3115) SW(15)
    WRITE (6,3120) SW(12)
    WRITE (6,3125) SW(11)
    WRITE (6,3130) SW(14)
    WRITE (6,3135) SQ(10,3)
    WRITE (6,3140) SQ(32,4)
    W2 = WABFUB - SQ(32,4)
    WRITE (6,3145) W2
    WRITE (6,3150) WABFUB
    WRITE (6,3155) SCR(214)
    GO TO 980
970 WRITE (6,3160) SW(15)
    WRITE (6,3175) SQ(32,5)
    WRITE (6,3185) SQ(32,2)
    WRITE (6,3180) SQ(33,1)
    WRITE (6,3215) SQ(35,2)
    WRITE (6,3190) SQ(33,4)
    WRITE (6,3195) SQ(34,2)
    WRITE (6,3200) SQ(34,5)
    WRITE (6,3205) SQ(35,5)
    WRITE (6,3220) SQ(35,3)
    WRITE (6,3170) SQ(32,3)
    WRITE (6,3185) SQ(33,2)
    WRITE (6,3225) SQ(35,4)
    WRITE (6,3135) SQ(10,3)
    WRITE (6,3210) SQ(32,4)
    WRITE (6,3150) WABFUB
    WRITE (6,3155) SCR(214)
    GO TO 980
975 WRITE (6,3230) SW(15)
    WRITE (6,3175) SQ(32,5)
    WRITE (6,3235) SQ(33,3)
    WRITE (6,3245) SQ(34,1)
    WRITE (6,3255) SQ(34,4)
    WRITE (6,3180) SQ(33,1)
    WRITE (6,3215) SQ(35,2)
    WRITE (6,3190) SQ(33,4)
    WRITE (6,3195) SQ(34,2)
    WRITE (6,3200) SQ(34,5)
    WRITE (6,3205) SQ(35,5)
    WRITE (6,3220) SQ(35,3)
    WRITE (6,3240) SQ(33,5)

```

```

WRITE (6,3250) SQ(34,3)
WRITE (6,3260) SQ(35,1)
WRITE (6,3185) SQ(33,2)
WRITE (6,3225) SQ(35,4)
WRITE (6,3135) SQ(10,3)
WRITE (6,3210) SQ(32,4)
WRITE (6,3150) WABFUB
WRITE (6,3155) SCB(214)

```

```

9A0 CONTINUE
RETURN

```

C

```

1000 FORMAT(1H1,25X,31HSPACE SHUTTLE SYNTHESIS SUMMARY)
1005 FORMAT (37X,7HROOSTER11X,7HORBITER11X,7HVEHICLE/)
1010 FORMAT (12H WEIGHT (LB))
1011 FORMAT(32H      PROPELLANT, ASCENT LESS FPRF12.0, F18.0)
1012 FORMAT (27H      PROPELLANT, ASCENT FPR, F17.0, F18.0)
1015 FORMAT(30H      PROPELLANT, ASCENT TOTAL F14.0, 2F18.0)
1020 FORMAT (36H      PROPELLANT, ORBIT MANEUVER      8X, 2F18.0)
1022 FORMAT (26H      FLYBACK FUEL                      3F18.0)
1025 FORMAT (26H      PROPELLANT, TOTAL                  3F18.0)
1030 FORMAT (26H      PAYLOAD                          18X, 2F18.0)
1035 FORMAT(26H      STRUCTURE                          3F18.0)
1040 FORMAT (26H      OTHER                              3F18.0)
1041 FORMAT (26H      CONTINGENCY                      3F18.0)
1045 FORMAT (26H      TOTAL                            3F18.0)
1050 FORMAT (13H0 VOLUME (FT3))
1055 FORMAT (26H      FUEL TANK                        3F18.0)
1060 FORMAT (26H      OXIDIZER TANK                    3F18.0)
1065 FORMAT (26H      PROPELLANT TANKS                  3F18.0)
1070 FORMAT (26H      PAYLOAD                          18X, 2F18.0)
1075 FORMAT (26H      OTHER                              3F18.0)
1077 FORMAT (26H      TOTAL                            3F18.0)
1080 FORMAT (9H GEOMETRY)
1085 FORMAT (26H      LENGTH (FT)                      3F18.1)
1090 FORMAT (27H      BODY WETTED AREA (FT2)           F17.1, 2F18.1)
1095 FORMAT (29H      BODY PLANFORM AREA (FT2)         F15.1, 2F18.1)
1096 FORMAT (33H      THEORETICAL WING AREA (FT2) F11.1, F18.1)
1100 FORMAT (23H      WING LOADING (PSF) F21.1, F18.1)
1105 FORMAT (11H PROPULSION)
1110 FORMAT (26H      THRUST-TO-WEIGHT                18X, 2F18.5)
1111 FORMAT (26H      NO. OF ENGINES                  2F18.0)
1112 FORMAT (26H      SL THRUST/ENG NOM/UR , 2(F10.0, 1H/, F7.0) ,
1 F9.0, 1H/, F8.0)
1113 FORMAT (26H      VAC THRUST/ENG NOM/UR, 2(F10.0, 1H/, F7.0) ,
1 F9.0, 1H/, F8.0)
1115 FORMAT (26H      SL ISP NOM/UR                    , 3(F12.1, 1H/, F5.1) )
1120 FORMAT (26H      VAC ISP NOM/UR                    , 3(F12.1, 1H/, F5.1) )
1125 FORMAT (11H0 TRAJECTORY )
1126 FORMAT (26H      MASS RATIO                      2F18.5)
1127 FORMAT (35H      MAXIMUM DYNAMIC PRESSURE (PSF) 27X, F18.1)
1128 FORMAT(34H      CHARACTERISTIC VELOCITY (FPS) 3X, F7.0, 2F18.0)
1130 FORMAT (35H      STAGING DYNAMIC PRESSURE (PSF) 27X, 2F18.0)

```

1135 FORMAT (38H STAGING VELOCITY (RELATIVE) (FPS)24X,F18.0)
 1140 FORMAT (35H STAGING ALTITUDE (FT) 27X,F18.0)
 1145 FORMAT (47H STAGING FLIGHT PATH ANGLE (RELATIVE) (DEG)15X,
 1F18.3)
 1150 FORMAT (40H INJECTION VELOCITY (INERTIAL) (FPS)4X,F18.0)
 1155 FORMAT (35H INJECTION ALTITUDE (FT) 9X,2F18.0)
 1160 FORMAT (49H INJECTION FLIGHT PATH ANGLE (INERTIAL) (DEG)F13.3)
 1165 FORMAT (35H INJECTION INCLINATION (DEG) 9X,2F18.2)
 1170 FORMAT (26H0 IN ORBIT 18X,2F18.0)
 1175 FORMAT (26H RETURN CONDITION 2F18.0)
 1180 FORMAT (26H ENTRY 2F18.0)
 1185 FORMAT (26H LANDING 2F18.0)
 1195 FORMAT (26H0FLYBACK RANGE (N MI) F18.1)
 1200 FORMAT (22H0SOLID AUGMENTATION ,F4.0,7H SOLIDS,2X,15HPROPELLANT
 1WT.= F8.0,2X,10HINERT WT.= F8.0/
 2 12X,11HTOTAL WT. =F9.0,5X,11HEXIT A
 3REA = F6.0,5X,11HBURN TIME =F5.0/
 4 12X,5HISP =F7.2, 5X,16HVAC THRUST/
 5ENG= F8.0,4H + (,F7.0,9H * TIME) / 12X,15HSL THRUST/ENG = F9.0)
 2112 FORMAT (26H SL THRUST/ENG NOM/UR ,
 1 F10.0,1H/,F7.0,F10.0,1H/,7X, F9.0,1H/,F8.0)
 2113 FORMAT (26H VAC THRUST/ENG NOM/UR ,
 1 F10.0,1H/,F7.0,F10.0,1H/,7X, F9.0,1H/,F8.0)
 2115 FORMAT (26H SL ISP NOM/UR
 1 F12.1,1H/,F5.1,F12.1,1H/,5X,F12.1,1H/,F5.1)
 2120 FORMAT (26H VAC ISP NOM/UR
 1 F12.1,1H/,F5.1,F12.1,1H/,5X,F12.1,1H/,F5.1)
 2000 FORMAT (34H0DRAG LOSS AT STAGING = F10.2)
 2010 FORMAT (34H GRAVITY LOSS AT STAGING = F10.2)
 2015 FORMAT (34H DRAG LOSS AT INJECTION = F10.2)
 2020 FORMAT (34H GRAVITY LOSS AT INJECTION = F10.2)
 2025 FORMAT (34H MISALIGNMENT LOSS AT INJECTION = F10.2)
 2030 FORMAT (11H0PITX = E16.7)
 2035 FORMAT (11H SALP(5) = E16.7)
 2040 FORMAT (11H PSIOT(5) = E16.7)
 2045 FORMAT (30H BOOSTER BURN TIME (SEC. 2) = F7.2)
 2046 FORMAT (33H ORBITER BURN TIME (SEC. 5,6,7) = F7.2)
 2050 FORMAT (9H0FLYBACK)
 2055 FORMAT (24H DETAILED RANGE, CLVG = E16.7,5X,12HBANK ANGLE = F7.2)
 2060 FORMAT (27H F(Q STAGE), Q AT STAGING = F10.3)
 2065 FORMAT (17H CONSTANT, DRNG = F10.3)
 2070 FORMAT (22H IIP, CENTRAL ANGLE = F8.2,5X,5H IIP= F8.2)
 2075 FORMAT (38H NUMERICAL INTEGRATION, MAX. LOADING = F7.3)
 2080 FORMAT (5X,6HTLAT = F8.3)
 2085 FORMAT (5X,6HTLNG = F8.3)
 2090 FORMAT (5X15HSURFACE RANGE = F9.3)
 2095 FORMAT (5X10HALTITUDE = F10.1)
 2100 FORMAT (5X 10HLATITUDE = F8.3)
 2105 FORMAT (5X,11HLONGITUDE = F8.3)
 2110 FORMAT (5X,15HAZIMUTH DIFF. = F8.3)
 2115 FORMAT (16H0CRUISE, RANGE = F8.2)
 2120 FORMAT (5X,5HSFC = F9.4)

3125 FORMAT (5X,5HL/D = F9.4)
3130 FORMAT (5X,17HCRUISE VELOCITY = F9.2)
3135 FORMAT (5X,6HWRNG = F9.2)
3140 FORMAT (5X,7HWFLYX = F9.2)
3145 FORMAT (5X,13HCRUISE FUEL = F10.2)
3150 FORMAT (5X,14HFLYBACK FUEL = F10.2)
3155 FORMAT (5X,10H C(214) = F9.5)
3160 FORMAT (27H0PERCENTAGE WEIGHT, RANGE = F8.2)
3165 FORMAT (5X,5HCA = F9.4)
3170 FORMAT (5X,5HCB = F9.4)
3175 FORMAT (5X,5HPT = F9.2)
3180 FORMAT (5X,5HR1 = F9.2)
3185 FORMAT (5X,5HR3 = F9.2)
3190 FORMAT (5X,5HSFC2=F9.4)
3195 FORMAT (5X,5HALD2=F9.4)
3200 FORMAT (5X,6HVFLY2=F9.4)
3205 FORMAT (5X,5HR2 = F9.2)
3210 FORMAT (5X,6HWFLYX= F9.2)
3215 FORMAT (5X20HIDLE DESCENT FUEL = F9.2)
3220 FORMAT (5X20HCRUISE FUEL WEIGHT = F10.2)
3225 FORMAT (5X,27HFINAL DESCENT FUEL WEIGHT = F9.2)
3230 FORMAT (17H0BREGUET, RANGE = F9.2)
3235 FORMAT (5X,7HSFC1 = F9.4)
3240 FORMAT (5X,7HSFC3 = F9.4)
3245 FORMAT (5X,7HALD1 = F9.4)
3250 FORMAT (5X,7HALD3 = F9.4)
3255 FORMAT (5X,7HVFLY1= F9.4)
3260 FORMAT (5X,7HVFLY3= F9.4)
END

PRECEDING PAGE BLANK NOT FILMED

GOC-08870-002

Appendix IV-15 Program GTSM

```

C   OVERLAY(TTSS,2,0)
C   PROGRAM GTSM
      SUBROUTINE GTSM
      DIMENSION ICV(3)
      COMMON NSFC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1     VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2     C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3     XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4     TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5     PS(15)
1001  FORMAT(53H0 *** VEHICLE ATTEMPTED SUBTERRANEAN FLIGHT *****)
1002  FORMAT(36H0 ***** TABLE ARGUMENT ERROR *****)
      IF(SV(35).GT.1.5) GO TO 14
      KA = 1
      KI1 = 1
      KI2 = 2
      LSFC = IFIX(SV(36))
      GO TO 15
14    KA = IFIX(SV(36)) + 1
      KI1 = 3
      KI2 = 4
      LSEC = IFIX(V(0))
15    W(36) = .1
      V(55) = 0.
      V(68) = 0.
      DO 10 KI=KI1,KI2
      W( ) = .1
      V( ) = .6
      DO 43 I=9,12
43    W(I) = .1
      IF(VQ(28,KI).GT..5) GO TO 40
      IF(VQ(32,KI)-.5) 10, 10, 41
40    DO 35 K=1,24
35    V(K+10) = VQ(K,KI)
      DO 36 K=25,28
36    V(K+27) = VQ(K,KI)
      DO 37 K=29,31
37    V(K+28) = VQ(K,KI)
41    DO 38 K=32,34
38    V(K+36) = VQ(K,KI)
      V(80) = VQ(35,KI)
      DO 39 K=1,7
39    W(K+12) = WQ(K,KI)
      W(24) = WQ(8,KI)
      W(34) = WQ(9,KI)
      W(35) = WQ(10,KI)
      ISLCT = IFIX(VQ(28,KI))
      IF(ISLCT.LT.1) GO TO 10
      DO 44 K=1,3
      IISC = IFIX(V(K+13))
      JJSC = IISC/100
C   **** ICV(K) IS THE K-TH PRIMARY CONTROL VARIABLE SECTION NUMBER ***

```

```

44 ICV(K) = IISC - JJSC*100
   GO TO (11,12,13,12,13,12,12,12), ISLCT
11 KARG = ICV(1)
   GO TO 18
12 KARG = MIN0(ICV(1),ICV(2))
   GO TO 18
13 KARG = MIN0(ICV(1),ICV(2),ICV(3))
18 IF(KARG.GT.LSEC) GO TO 997
   KB = KARG - 1
   TF(KB.LT.KA) GO TO 19
   W(7) = 1.1
   W(8) = .1
   T998 = 1
56 CALL TRAJR(JFLAG,KA,KB,KI)
   IF(JFLAG.LE.0.AND.T998.LE.1) GO TO 54
   GO TO (55,998), T998
55 T998 = 2
   SW2 = SW(2)
   SW(2) = 1.
   GO TO 56
54 KA = KB + 1
   IF(KA.GT.LSEC) GO TO 999
   W(9) = .1
19 W(7) = .1
   GO TO (21,32,33,32,33,33,33,33), ISLCT
21 W(8) = 1.1
   ARG = V(20)
   KP = IFIX(ARG)
   IF(KB.GT.LSEC) GO TO 997
   CALL XNEW1(JFLAG,1,KA,KB,KI)
   GO TO 28
32 W(8) = 2.1
   ARG = AMAX1(V(20),V(21))
   GO TO 31
33 W(8) = 3.1
   ARG = AMAX1(V(20),V(21),V(22))
31 KB = IFIX(ARG)
   TF(KB.GT.LSEC) GO TO 997
   GO TO (21,21,21,24,24,24,24,24), ISLCT
   GO TO (21,22,23,24,25,26,27,27), ISLCT
C 22 CALL XNEW2(JFLAG,1,2,KA,KB,KI)
C   GO TO 28
C 23 CALL XNEW3(JFLAG,1,2,3,KA,KB,KI)
C   GO TO 28
C 24 CALL XNEW4(JFLAG,1,2,KA,KB,KI)
C   GO TO 28
C 25 CALL XNEW5(JFLAG,1,2,3,KA,KB,KI)
C   GO TO 28
C 26 CALL XNEW6(JFLAG,1,2,3,KA,KB,KI)
C   GO TO 28
C 27 CALL XNEW7(JFLAG,1,2,3,KA,KB,KI)
28 CONTINUE

```

```

      IF(W(70).GT..5) W(36)=1.1
      W(7)=1.1
      ITRIP = 1
      IF(JFLAG.LE.0.AND.W(36).LT..5) GO TO 50
      ITRIP = 2
      SW2 = SW(2)
      SW(2) = 1.
50  CALL TRAJB(JFLAG,KA,K8,KI)
      GO TO (57,58), ITRIP
58  ITRIP = 1
      SW(2) = SW2
57  KA = K8 + 1
      IF(JFLAG.GT.0) GO TO 998
      IF(KA.GT.LSEC) GO TO 999
10  CONTINUE
997 W(7) = 1.1
      W(8) = .1
      IF(KA.GT.LSEC) GO TO 999
      W(9) = .1
      I998 = 1
51  CALL TRAJB(JFLAG,KA,LSEC,KI)
      IF(JFLAG.LE.0.AND.I998.LE.1) GO TO 999
      GO TO (52,998), I998
52  I998 = 2
      SW2 = SW(2)
      SW(2) = 1.
      GO TO 51
998 I998 = 1
      SW(2) = SW2
      W(36) = 1.1
      IF(Z(8).LE.-10000.) WRITE(6,1001)
      IF(Z(8).GT.-10000.) WRITE(6,1002)
999 RETURN
      END

```

PRECEDING PAGE BLANK NOT FILMED

Appendix IV-16 Subroutine CNTVR

600-25870-002

```
SUBROUTINE CNTVR(ITYPE,LCV)
  DIMENSION SAVE(3)
  COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1  VO(38,4),WO(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2  Q2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3  XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4  TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5  PS(15)
  ISEC = IFIX(V(LCV+13))
  IPAR = IFIX(V(LCV+10))
  MSEC = 0
  LSEC = 0
  KSEC = 0
  JSEC = 0
  IF(ISEC.LT.100) GO TO 51
  MSEC = ISEC/100000000
  IARG = ISEC - MSEC*100000000
  LSFC = IARG/1000000
  IARG = IARG - LSEC*1000000
  KSEC = IARG/10000
  IARG = IARG - KSEC*10000
  JSEC = IARG/100
  ISEC = IARG - JSEC*100
51 GO TO (11,12,13,11), ITYPE
11 IF(ISEC.GT.0) GO TO 20
  CV(LCV) = V(IPAR)
  GO TO 21
20 CV(LCV) = Q(IPAR,ISEC)
  IF(MSEC.GT.0.AND.ITYPE.LT.2) DM = Q(IPAR,MSEC) - Q(IPAR,ISEC)
  IF(LSEC.GT.0.AND.ITYPE.LT.2) DL = Q(IPAR,LSEC) - Q(IPAR,ISEC)
  IF(KSEC.GT.0.AND.ITYPE.LT.2) DK = Q(IPAR,KSEC) - Q(IPAR,ISEC)
  IF(JSEC.GT.0.AND.ITYPE.LT.2) DJ = Q(IPAR,JSEC) - Q(IPAR,ISEC)
21 IF(ITYPE.GT.3) GO TO 999
  SAVE(LCV) = CV(LCV)
  GO TO 999
12 IF(ISEC.GT.0) GO TO 30
  V(IPAR) = CV(LCV)
  GO TO 999
30 Q(IPAR,ISEC) = CV(LCV)
  GO TO 50
13 IF(ISEC.GT.0) GO TO 40
  V(IPAR) = SAVE(LCV)
  GO TO 999
40 Q(IPAR,ISEC) = SAVE(LCV)
50 IF(MSEC.GT.0) Q(IPAR,MSEC) = Q(IPAR,ISEC) + DM
  IF(LSEC.GT.0) Q(IPAR,LSEC) = Q(IPAR,ISEC) + DL
  IF(KSEC.GT.0) Q(IPAR,KSEC) = Q(IPAR,ISEC) + DK
  IF(JSEC.GT.0) Q(IPAR,JSEC) = Q(IPAR,ISEC) + DJ
999 RETURN
  END
```

Appendix IV-17 Subroutine XNEW1

```

C      SUBROUTINE XNEW1(JFLAG,ICV,KA,KB,KI)
          GENERAL (1X1) NEWTON-RAPHSON ITERATION SCHEME
          COMMON NSEC,T,PP,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1         VO(38,4),WO(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2         C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3         XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4         TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5         PS(15)
1001  FORMAT(51H1      ***** SUCCESSFUL CONVERGENCE IN XNEW1 *****
1002  FORMAT(66H1      ***** EXCEEDED MAXIMUM NUMBER OF ITERATIONS IN XNE
1H1      *****
1003  FORMAT(60H1      ***** CONTROL VARIABLE EXCEEDED LIMIT IN XNEW1 *
1*****//11X,18H CONTROL VARIABLE ,I1,3H =,E17.8,4X,10H LIMITS =,
2E17.8,4H TO,E17.8)
1004  FORMAT(70H1      ***** END CONDITIONS ARE INVARIANT TO CONTROL PARA
1METERS *****
          MCYC = IFIX(V(ICV+56))
          ICYC = 1
          ARG=SV(24)
          JFLAG = 0
          W(10) = .1
          W(11) = W(11) + 1.
          CALL CNTVR(1,ICV)
          CALL TRAJR(KFLAG,KA,KB,KI)
          X1 = CV(ICV)
          Y1 = EC(ICV)
          ARG1 = Y1 - V(ICV+51)
          IF(W(30).GT..5) GO TO 999
          IF(KFLAG.GT.0) GO TO 11
          IF(ABS(ARG1).LE.V(ICV+31)) GO TO 12
          X2=X1-ARG1/SV(24)
          CV(ICV) = X2
          CALL CNTVR(2,ICV)
          CALL TRAJR(KFLAG,KA,KB,KI)
          Y2 = EC(ICV)
          ARG2 = Y2 - V(ICV+51)
          IF(W(30).GT..5) GO TO 999
          IF(KFLAG.GT.0) GO TO 11
          ARG=(Y2-Y1)/(X2-X1)
          IF(ABS(ARG2).LE.V(ICV+31)) GO TO 12
          PREDICT NEW VALUE FOR THE CONTROL VARIABLE
14  CONTINUE
          ARG=(Y2-Y1)/(X2-X1)
          IF(ABS(ARG).LT..0000001) GO TO 997
          X3=X1+(V(ICV+51)-Y1)/ARG
          IF(X3.GT.V(ICV+25).OR.X3.LT.V(ICV+28)) GO TO 15
          CV(ICV) = X3
          CALL CNTVR(2,ICV)
          CALL TRAJR(KFLAG,KA,KB,KI)
          Y3 = EC(ICV)
          ARG3=Y3-V(ICV+51)
          IF(W(30).GT..5) GO TO 999

```

```
IF(KFLAG.GT.0) GO TO 11
ARG=(Y3-Y1)/(X3-X1)
IF(ABS(ARG3).LT.V(ICV+31)) GO TO 12
IF(ICYC.GE.MCYC) GO TO 13
ICYC = ICYC + 1
IF(ABS(ARG1).LT.ABS(ARG2)) GO TO 16
X1 = X2
Y1 = Y2
ARG1=ARG2
16 X2 = X3
Y2 = Y3
ARG2=ARG3
GO TO 14
11 JFLAG = 1
GO TO 999
C      SUCCESSFUL CONVERGENCE
12 IF( SW(2).EQ.1. .OR. SW(16).NE.0. )
1WRITE(6,1001)
SV(24)=ARG
GO TO 999
C      EXCEEDED MAXIMUM NUMBER OF ITERATIONS
13 WRITE(6,1002)
GO TO 998
C      CONTROL VARIABLE EXCEEDED LIMIT
15 WRITE(6,1003) ICV, X3, V(ICV+28), V(ICV+25)
GO TO 998
C      END CONDITIONS ARE INVARIANT TO CONTROL PARAMETERS
997 WRITE(6,1004)
998 W(30) = 1.1
999 RETURN
END
```

Appendix IV-18 Subroutine XNEW4

```

SUBROUTINE XNEW4(JFLAG,ICV,JCV,KA,KB,KI)
  GENERAL (2X2) ITERATION SCHEME
  COMMON NSEC,T,TP,TS(15),V(85),Q(7,15),W(50),Z(120),EC(3),CV(3),
  1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
  2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
  3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
  4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
  5 PS(15)
1001 FORMAT(51H1 ***** SUCCESSFUL CONVERGENCE IN XNEW4 ***** )
1002 FORMAT(66H1 ***** EXCEEDED MAXIMUM NUMBER OF ITERATIONS IN XNE
  1W4 ***** )
1003 FORMAT(60H1 ***** CONTROL VARIABLE EXCEEDED LIMIT IN XNEW4 *
  1*****///11X,18H CONTROL VARIABLE ,I1,3H =,E17.8,4X,10H LIMITS =,
  2E17.8,4H TO,E17.8//11X,18H CONTROL VARIABLE ,I1,3H =,E17.8,4X,10
  3H LIMITS =,E17.8,4H TO,E17.8)
1004 FORMAT(70H1 ***** END CONDITIONS ARE INVARIANT TO CONTROL PARA
  1METERS ***** )
  MCYC = IFIX(V(ICV+56))
  ICYC = 1
  JFLAG = 0
  IPRTL = 0
  W(10) = .1
  W(11) = W(11) + 1.
  W(12) = 1.1
  CALL CNTVP(1,ICV)
  CALL CNTVR(1,JCV)
10 CALL TRAJB(KFLAG,KA,KB,KI)
  IF(W(30).GT..5) GO TO 999
  IF(KFLAG.GT.0) GO TO 11
  IF(ABS(EC(ICV)-V(ICV+51)).LE.V(ICV+31).AND.ABS(EC(JCV)-V(JCV+51)).
  1LE.V(JCV+31)) GO TO 12
  IF(ICYC.GT.MCYC) GO TO 13
  X10 = CV(ICV)
  X20 = CV(JCV)
  Y10 = EC(ICV)
  Y20 = EC(JCV)
  DY1 = V(ICV+51) - Y10
  DY2 = V(JCV+51) - Y20
  IF(IPRTL.GT.0) GO TO 16
  EVALUATE PARTIAL DERIVATIVES
  IPRTL = 1
  X11 = X10 + V(ICV+22)
  X21 = X20 + V(JCV+22)
  H1 = V(ICV+22)**2
  H2 = V(JCV+22)**2
  CV(ICV) = X11
  CALL CNTVR(2,ICV)
  CALL TRAJB(KFLAG,KA,KB,KI)
  IF(W(30).GT..5) GO TO 999
  IF(KFLAG.GT.0) GO TO 11
  Y11 = EC(ICV)
  Y21 = EC(JCV)

```

```

CALL CNTVR(3,ICV)
CALL CNTVR(4,ICV)
CV(JCV) = X21
CALL CNTVR(2,JCV)
CALL TRAJB(KFLAG,KA,KB,KI)
IF(W(30).GT..5) GO TO 999
IF(KFLAG.GT.0) GO TO 11
Y12 = EC(ICV)
Y22 = EC(JCV)
CALL CNTVR(3,JCV)
CALL CNTVR(4,JCV)
C11 = Y11 - Y10
C21 = Y21 - Y20
C12 = Y12 - Y10
C22 = Y22 - Y20
D1 = X11 - X10
D2 = X21 - X20
A11 = C11/D1
A21 = C21/D1
A12 = C12/D2
A22 = C22/D2
C PREDICT NEW VALUES FOR CONTROL VARIABLES
14 ICYC = ICYC + 1
DA = A11*A22 - A21*A12
IF(ABS(DA).LT..0000001) GO TO 997
D1 = A22*DY1 - A12*DY2
D2 = A11*DY2 - A21*DY1
DX1 = D1/DA
DX2 = D2/DA
CV(ICV) = X10 + DX1
CV(JCV) = X20 + DX2
IF(CV(ICV).GT.V(ICV+25).OR.CV(JCV).GT.V(JCV+25)) GO TO 15
IF(CV(ICV).LT.V(ICV+28).OR.CV(JCV).LT.V(JCV+28)) GO TO 15
CALL CNTVR(2,ICV)
CALL CNTVR(2,JCV)
GO TO 10
16 CONTINUE
C ***** ADJUST PARTIAL DERIVATIVES *****
SIG = DX1**2/H1 + DX2**2/H2
A11 = A11 - DY1*DX1/SIG/H1
A12 = A12 - DY1*DX2/SIG/H2
A21 = A21 - DY2*DX1/SIG/H1
A22 = A22 - DY2*DX2/SIG/H2
GO TO 14
11 JFLAG = 1
GO TO 999
C SUCCESSFUL CONVERGENCE
12 IF( SW(2).EQ.1. .OR. SW(16).NE.0. )
1WRITE(6,1001)
GO TO 999
C EXCEEDED MAXIMUM NUMBER OF ITERATIONS
13 WRITE(6,1002)

```



```

C      GO TO 998
      CONTROL VARIABLE EXCEEDED LIMIT
15 WRITE(6,1003) (IJK, CV(IJK), V(IJK+28), V(IJK+25), IJK=ICV,JCV)
      GO TO 998
C      END CONDITIONS ARE INVARIANT TO CONTROL PARAMETERS
997 WRITE(6,1004)
998 W(30) = 1.1
999 RETURN
      END

```

PRECEDING PAGE BLANK NOT FILMED

GOC-09P70-002

Appendix IV-19 Subroutine TRAJB

```
SUBROUTINE TRAJB(IFLAG,KA,K8,KI)
  DIMENSION XW3(5), XW4(5)
  DIMENSION Y(12), IEC(3), NUM(4), ZS(100)
  COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
  1 VQ(38,4),WO(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
  2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
  3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
  4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
  5 PS(15)
  COMMON/AEROX/XW3,XW4
  COMMON/LTS/WW(5)
1001 FORMAT(1H0//5X9H ABS TIME,11X,9H REL TIME,11X,9H ALTITUDE,11X,9H L
  1ATITUDE,11X,10H LONGITUDE,10X,7H HEIGHT//5X,13H REL VELOCITY,7X,10
  2H REL GAMMA,10X,12H REL AZIMUTH,8X,15H INRTL VELOCITY, 5X,12H IN
  3RTL GAMMA,8X,14H INRTL AZIMUTH/5X,12H HEATING PAR,8X,10H HEAT FLUX
  4,10X,7H RADIUS,13X,12H TARGET DIST,8X,12H CROSS RANGE,8X,11H DOWN
  5RANGE/5X,11H ROLL ANGLE,9X,6H ALPHA,14X,10H YAW ANGLE,10X,12H PITC
  6H ANGLE,8X,11H WEIGHT DOT,9X,4H ISP/5X,8H C (YAW),12X,10H C (AXIAL
  7),10X,11H C (NORMAL),9X,12H YAW F(AERO),8X,14H AXIAL F(AERO),6X,13
  8H NORM F(AERO)//5X,7H THRUST,13X,12H THRUST LOAD,8X,12H PITCH RATE
  9 ,8X,14H FORCE (AZMTH),6X,12H FORCE (VEL),8X,14H FORCE (GAMMA)/5X,
  112H YAW AERO LD,8X,14H AXIAL AERO LD,6X,13H NORM AERO LD,7X,11H AX
  2IAL LOAD,9X,14H TOTAL AERO LD,6X,9H TOTAL LD/5X,10H DYN PRESS,10X,
  310H ATM PRESS,10X,10H VEL SOUND,10X,12H MACH NUMBER,8X,15H MERIDIA
  4N GRVTY,5X,13H RADIAL GRVTY/5X,12H TOTAL GRVTY,8X,12H SURF RADIUS,
  58X,13H GRAVITY LOSS,7X,10H DRAG LOSS,10X,14H MISALIGN LOSS,6X,10H
  6IDEAL VFL/)
1002 FORMAT(20H0 BURNOUT WEIGHT =,E16.8,21H, JETTISON WEIGHT =,E16.
  18,23H, SURSEQUENT WEIGHT =,F15.8)
1003 FORMAT(23H FIRST CONTROL VAR =,E16.8)
1004 FORMAT(23H FIRST CONTROL VAR =,E16.8,23H, SECOND CONTROL VAR =
  1,E16.8)
1005 FORMAT(23H FIRST CONTROL VAR =,E16.8,23H, SECOND CONTROL VAR =
  1,E16.8,22H, THIRD CONTROL VAR =,E16.8)
1006 FORMAT(42H0 NUMBER OF TRAJECTORIES FOR THIS CASE =,I4,12X,36H TR
  1AJECTORIES PER INSIDE ITERATION =,I4,750H INSIDE ITERATIONS PE
  2R INTERMEDIATE ITERATION =,I4,4X,48H INTERMEDIATE ITERATIONS PER 0
  3UTSIDE ITERATION =,I4,72H SIMULATION SECTION,I3,33X,10H PAYL
  4OAD =,F16.8)
1007 FORMAT(22H1 SIMULATION SECTION,I3)
1008 FORMAT(20H0 FIRST END COND =,E16.8///)
1009 FORMAT(20H0 FIRST END COND =,E16.8,21H, SECOND END COND =,E16.
  18///)
1010 FORMAT(20H0 FIRST END COND =,E16.8,21H, SECOND END COND =,E16.
  18,20H, THIRD END COND =,E16.8///)
1011 FORMAT(22H1 SIMULATION SECTION,I3,33X,10H PAYLOAD =,E16.8)
1012 FORMAT(23H TARGET AZIMUTH =,E16.8,23H, TARGET DOWN RANGE =
  1,E16.8///)
1013 FORMAT(5X,10H RADIUS(1),10X,10H RADIUS(2),10X,10H RADIUS(3),10X,12
  1H VELOCITY(1),8X,12H VELOCITY(2),8X,12H VELOCITY(3)/5X,9H ACCEL(1)
  2,11X,9H ACCEL(2),11X,9H ACCEL(3)///)
1014 FORMAT(5X,10H RADIUS(1),10X,10H RADIUS(2),10X,10H RADIUS(3),10X,12
```

```

1H VELOCITY(1),8X,12H VELOCITY(2),8X,12H VELOCITY(3)/5X,10H ACCEL(A
21),10X,10H ACCEL(A2),10X,10H ACCEL(A3),10X,10H ACCEL(G1),10X,10H A
3ACCEL(G2),10X,10H ACCEL(G3)///)
1015 FORMAT(1H0)
1016 FORMAT(5X,12H INCLINATION,8X,11H NODAL LONG,9X,12H PERIGEE ARG,
18X,12H ORBITAL PAR,8X,14H ENERGY (FPS)2,6X,14H ENERGY (KPS)2/5X,13
2H CIRCULAR VEL,7X,11H APOGEE RAD,9X,11H APOGEE VEL,9X,13H TRUE ANO
3MALY,7X,7H PERIOD,13X,13H ECCENTRICITY/5X,12H PERIGEE,RAD,8X,12H P
4ERIGEE LAT,8X,11H APOGEE LAT,9X,12H PERIGEE ALT,8X,11H APOGEE ALT)
1017 FORMAT(22H0 SIMULATION SECTION,I3)
1018 FORMAT(20H0 IDEAL VELOCITY =,E16.8,21H, VELOCITY RESET =,E16.
18,25H, SUBSEQUENT VELOCITY =,E16.8)
1019 FORMAT(5X,12H SLANT RANGE,8X,11H ELEV ANGLE,9X,8H AZIMUTH/)
1020 FORMAT(5X,12H SLANT RANGE,8X,11H ELEV ANGLE,9X,8H AZIMUTH,12X,15H
1SLANT RANGE(T),5X,14H ELEV ANGLE(T),6X,11H AZIMUTH(T)/)
1021 FORMAT(5X,11H IMPACT LAT,9X,12H IMPACT LONG,8X,14H IMPACT RADIUS,6
1X,15H TIME TO IMPACT,5X,15H INST IMPCT LNG,5X,15H ANGLE TO IMPCT//
2)
1022 FORMAT(5X,12H SLANT RANGE,8X,11H ELEV ANGLE,9X,8H AZIMUTH,12X,15H
1SLANT RANGE(T),5X,14H ELEV ANGLE(T),6X,11H AZIMUTH(T)/5X,14H SURF
2RANGE(V),6X,14H AZIMUTH DIFF ,6X,11H AZIMUTH(V)/)
1023 FORMAT(50H1 INITIATION OF BOOSTER RETURN FLIGHT TRAJECTORY//22H
1 SIMULATION SECTION,I3)
1024 FORMAT(50H1 INITIATION OF BOOSTER RETURN FLIGHT TRAJECTORY//22H
1 SIMULATION SECTION,I3,33X,10H PAYLOAD =,E16.8)
1025 FORMAT(1H0/44H **** MAXIMUM LOADING DURING FLYBACK = ,F8.5,9H
1 G ****/)
ISV36 = IFIX(SV(36)) + 1
LSEC = IFIX(V(9))
L = IFIX(W(8))
N = 12
IFLAG = 0
W(9) = W(9) + 1.
W(10) = W(10) + 1.
IF(KA.LE.1) GO TO 40
IF(KA.EQ.ISV36) GO TO 73
T = ST
C **** INITIALIZATION FOR SUBSEQUENT TRAJECTORY SEGMENTS ****
DO 41 K=1,100
41 Z(K) = ZS(K)
DO 43 K=1,7
43 Y(K) = Z(K)
DO 44 K=1,4
44 Y(K+7) = Z(K+16)
Y(12) = Z(39)
GO TO 42
C **** INITIALIZATION FOR THE FIRST TRAJECTORY SEGMENT ****
40 I = V(51)
SE(9) = 0.
DO 11 K=1,6
11 Y(K) = V(K)
Y(7) = V(7) + V(10)

```

```

      DO 10 K=1,5
10  Y(K+7) = 0.
      Z(43) = V(60)
      Z(11) = V(61)
      Z(12) = V(62)
      Z(13) = V(63)
      GO TO 42
C  ****  INITIALIZATION FOR THE BOOSTER RETURN FLIGHT  ****
73  T = SW(20)
      SV(42) = 0.
      Z(1) = SV(37)
      Z(2) = SV(8)
      Z(3) = SV(10)
      Z(4) = SV(31)
      Z(5) = SV(32)
      Z(6) = SV(33)
      Z(7) = SV(34)
      Z(8) = SV(9)
      Z(11) = SV(38)
      Z(12) = SV(39)
      Z(13) = SV(40)
      Z(43) = SV(41)
      DO 74 K=1,7
74  Y(K) = Z(K)
      DO 75 K=8,12
75  Y(K) = 0.
42  X = T
      DO 28 I=1,4
28  NUM(I) = IFIX(W(I+8))
      DO 12 NSEC=KA,KB
      DO 13 K=1,6
13  W(K) = .1
      DO 200 M=1,5
      XH3(M) = .1
200 XW4(M) = .1
C  ****  PRINT OUTPUT HEADERS  ****
      IF(NSEC.EQ.KA) GO TO 26
      IF(V(66).LT..5.AND.W(7).LT..5) GO TO 50
      IF(SW(2).NE.1..AND.SQ(1,2).EQ.0.) GO TO 22
      IF(NSEC-1SV36) 76, 77, 76
76  WRITE(6,1007) NSEC
      GO TO 22
77  WRITE(6,1023) NSEC
      GO TO 22
50  IF( SW(2).EQ.1. .OR. SQ(1,2).NE.0. )
1  WRITE(6,1017) NSEC
      GO TO 29
26  IARG = L + 1
      IF(SQ(1,2).NE.0.) GO TO 90
      IF(SW(2).NE.1.) GO TO 100
90  GO TO (27,19,20,21), IARG
19  WRITE(6,1003) CV(1)

```

```

GO TO 18
20 WRITE(6,1004) CV(1), CV(2)
GO TO 18
21 WRITE(6,1005) CV(1), CV(2), CV(3)
18 IF(V(68).GT..5) WRITE(6,1012) W(34), W(19)
WRITE(6,1006) NUM(1), NUM(2), NUM(3), NUM(4), NSEC, V(10)
100 IF(ABS(Q(33,KA)-19.5).GT.2.) GO TO 22
WW(1) = T
WW(5) = FLOAT(KA) + .1
GO TO (96,96,97,33), IARG
97 IF(W(9).GT.1.5) GO TO 22
96 WW(2) = -Q(37,KA)
IF(WW(2).LT.0.) WW(2) = WW(1) - WW(2)
GO TO 22
83 IF(Q(22,KB).LT.-.5) GO TO 84
WW(2) = WW(1)
DO 85 K=KA,KB
85 WW(2) = WW(2) + Q(23,K)
GO TO 22
84 WW(2) = Q(23,KB)
GO TO 22
27 IF(NSEC-ISO36) 78, 79, 78
78 WRITE(6,1011) NSEC, V(10)
GO TO 22
79 WRITE(6,1024) NSEC, V(10)
22 IF(V(66).LT..5.AND.W(7).LT..5) GO TO 54
IF(SW(2).NE.1..AND.SQ(1,2).EQ.0.) GO TO 54
WRITE(6,1001)
JSND = 1 + IFIX(Q(69,NSEC))
GO TO (67,68,69,70), JSND
68 WRITE(6,1019)
GO TO 67
69 WRITE(6,1020)
GO TO 67
70 WRITE(6,1022)
67 IF(Q(56,NSEC).LT..5) GO TO 71
WRITE(6,1016)
IF(Q(56,NSEC).GT.1.5) GO TO 72
WRITE(6,1015)
GO TO 71
72 WRITE(6,1021)
71 JSND = 1 + IFIX(V(56))
GO TO (51,52,52,53,53), JSND
52 WRITE(6,1013)
GO TO 32
53 WRITE(6,1014)
GO TO 32
51 IF(SW(2).EQ.1..OR.SQ(1,2).NE.0.) WRITE(6,1015)
GO TO 29
54 IF(V(56).LT..5) GO TO 51
32 IF(W(29).GT.900.) V(75) = V(4)
ARG = (180. - V(75))/V(45)

```

```

      W(27) = SIN(ARG)
      W(28) = COS(ARG)
29  TS(NSEC) = X
      PS(NSEC) = Z(43)
C  **** TRAJECTORY INTEGRATION ****
      CALL INTGRT(X,Y,N)
      IF(N.LE.0) GO TO 15
      DO 14 K=1,3
        IEC(K) = IFIX(V(K+19))
C  **** DETERMINATION OF ITERATION END CONDITIONS ****
      IF(IEC(K).EQ.NSEC) CALL ENDCN(K)
14  CONTINUE
      IF(ABS(Q(26,NSEC)).LT..00001) GO TO 60
C  **** JETTISON WEIGHT ****
      ARG1 = Y(7)
      IF(Q(26,NSEC).LT.0.) GO TO 16
      Y(7) = Y(7) - Q(26,NSEC)
      ARG2 = Q(26,NSEC)
      GO TO 17
16  Y(7) = V(10) - Q(26,NSEC)
      ARG2 = ARG1 - Y(7)
17  IF( SW(7).EQ.1. .OR. SQ(1,2).NE.0. )
    1WRITE(6,1002) ARG1, ARG2, Y(7)
      Z(7) = Y(7)
60  IF(Q(67,NSEC).LT..5) GO TO 80
C  **** RESET IDEAL VELOCITY ****
      IVEL = IFIX(Q(67,NSEC))
      GO TO (61,62,63,64,62,61), IVEL
62  Y(9) = 0.
      Y(10) = 0.
      Y(11) = 0.
      IF(IVEL.EQ.5) GO TO 63
61  Y(8) = Q(68,NSEC)
      GO TO 65
64  Y(9) = Y(9) + Q(68,NSEC)
      Y(10) = Y(10) + Q(68,NSEC)
      Y(11) = Y(11) + Q(68,NSEC)
63  Y(8) = Y(8) + Q(68,NSEC)
65  ARG = Y(8) - Z(17)
      WRITE(6,1018) Z(17), ARG, Y(8)
      DO 66 J=1,4
66  Z(J+16) = Y(J+7)
80  JEC = 3
      LEC = IFIX(SV(36))
      KEC = IFIX(V(74))
      IF(NSEC.EQ.2) SQ(37,1) = T
      IF(NSEC.EQ.4) SQ(37,3) = T
      IF(KEC.LE.0) KEC = LEC
      IF (NSEC.NE.3) GO TO 82
      SV(8) = Z(2)
      SV(9) = Z(8)
      SV(10) = Z(3)

```

```

SV(12) = Z(9)
SV(31) = Z(4)
SV(32) = Z(5)
SV(33) = Z(6)
SV(37) = Z(1)
SV(38) = Z(11)
SV(39) = Z(12)
SV(40) = Z(13)
SV(41) = Z(14)
SW(20) = T
SW(10) = 7(49)
SQ(12,1) = Z(89)
SQ(14,1) = Z(17)
SQ(36,1) = Z(19)
SQ(36,2) = Z(18)
82 IF(NSEC.NE.KEC) GO TO 81
SV(3) = Z(17)
81 IF(NSEC.NE.LEC) GO TO 12
SV(4) = 7(7)
SV(7) = -Q(26,3)
SV(5) = SV(7) - SV(4)
SV(6) = SV(7)/SV(4)
SQ(14,2) = 7(17)
SQ(14,4) = 7(41)
SQ(14,5) = Z(8)
SQ(15,3) = Z(42)
SQ(15,4) = Z(66)
SQ(36,3) = Z(19)
SQ(36,4) = Z(18)
SQ(36,5) = Z(20)
SQ(37,2) = T - SQ(37,3)
12 CONTINUE
IF(NSEC.LT.LSEC) GO TO 101
IF(V(66).LT..5.AND.W(7).LT..5) GO TO 101
IF(SW(2).NE.1..AND.SQ(1,2).EQ.0.) GO TO 101
IF(SQ(19,5).EQ.5.) WRITE(6,1025) SV(42)
101 IF(ABS(Q(33,K8)-19.5).GT.2..OR.ABS(W(8)-2.).GT..5) GO TO 95
IF(W(9).LT.Q(37,K8)+.5) GO TO 95
DO 93 K=1,2
IF(ABS(EC(K)-V(K+51)).GT.V(K+31)) GO TO 94
93 CONTINUE
GO TO 95
94 CONTINUE
C 94 WW(2) = T (DELETED HERE)
95 IF(W(7).LT..5) GO TO 45
C **** SAVE END CONDITIONS FOR SUBSEQUENT TRAJECTORY SEGMENTS ****
ST = T
DO 46 K=1,100
46 ZS(K) = Z(K)
45 IF(SW(16).NE.0.) GO TO 92
IF(SQ(1,2).EQ.1.) GO TO 92
RETURN

```

92 GO TO (999,23,24,25), IARG
15 IFLAG = 1
N = 12
GO TO 999
C **** PRINT END CONDIONS ****
23 WRITE(6,1008) EC(1)
GO TO 999
24 WRITE(6,1009) EC(1), EC(2)
GO TO 999
25 WRITE(6,1010) EC(1), EC(2), EC(3)
999 RETURN
END

PRECEDING PAGE BLANK NOT FILMED

GDC-08879-002

Appendix IV-20 Subroutine ENDCN

```
SUBROUTINE ENDCN(LCV)
COMMON NSEC,T,TR,TS(15),V(85),Q(77,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WO(13,4),XH1(1,1),A1(1,1),C1(1,1,1),XH2(25,5),A2(12,5),
2 C2(25,12,5),XH3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
ISEND = IFIX(V(LCV+16))
IF(ISEND) 22,21,20
22 EC(LCV) = T
GO TO 999
21 EC(LCV) = TR
GO TO 999
20 EC(LCV) = Z(ISEND)
999 RETURN
END
```

PRECEDING PAGE BLANK NOT FILMED

Appendix IV-21 Subroutine INTGRT

GDC-DBB70-002

```

SUBROUTINE INTGRT(X,Y,N)
  DOUBLE PRECISION Y1(12),Y2(12),Y3(12),Y4(12),Y5(12),
  *F0(12),H0,H2,H3,H6,H8,Y0(12),X0,ARGD
  DIMENSION TMAX(12),TMIN(12),F0(12),F1(12),F2(12),F3(12),F4(12),F5
  112),Y(12),ERROR(12),YS1(12),YS2(12),YS3(12),YS4(12),YS5(12)
  COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
  1 V0(38,4),W0(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
  2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
  3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
  4 TWD9(25,2),PT(25,4),PDT(25,4),XNP1(4),SV(50),SQ(40,5),SE(40),
  5 PS(15)
1001 FORMAT(71H0 **** STEPSIZE IS LESS THAN THE INPUT STEPSIZE IN SIM
  1ULATION SECTION,I3,7H ****//6X,10H STEP SIZE,6X,9H MIN STEP,7X,
  2H ERROR(1),7X,9H ERROR(2),7X,9H ERROR(3),7X,9H ERROR(4),7X,9H ERRO
  3R(5)/6X,9H ERROR(6),7X,9H ERROR(7),7X,9H ERROR(8),7X,9H ERROR(9),7
  4X,10H ERROR(10),6X,10H ERROR(11),6X,10H ERROR(12)//5X,7E16.8/5X,7E
  516.8///)
1002 FORMAT(27H0 **** TERMINATED SECTION,I3,22H ON BACKUP TIME ****
  1//7X,5H TIME,2X,10H ABS BTIME,2X,10H REL BTIME,2X,10H BTIME TOL,2X
  2,10H TERM CODE,4X,11H TERM VALUE,7X,9H TERM TOL,8X,4H DIR//3F12.3,
  3F12.6,I12,2E18.8,F7.1///)
1003 FORMAT(38H0 **** FAILED IN BACKUP TIME SECTION,I3,19H TERMINATIO
  1N ****//7X,5H TIME,2X,10H ABS BTIME,2X,10H REL BTIME,2X,10H BTIM
  2E TOL,2X,10H TERM CODE,4X,11H TERM VALUE,7X,9H TERM TOL,8X,4H DIR/
  3/3F12.3,F12.6,I12,3E18.8,F7.1///)
1004 FORMAT(33H0 **** FAILED IN NORMAL SECTION,I3,19H TERMINATION *
  1****//7X,5H TIME,2X,10H ABS BTIME,2X,10H REL BTIME,2X,10H BTIME TOL
  2,2X,10H TERM CODE,4X,11H TERM VALUE,7X,9H TERM TOL,8X,4H DIR//3F12
  3.3,F12.6,I12,2E18.8,F7.1///)

  NSECX = SQ(2,1) + .1
  H = Q(7,NSEC)
  ISTG = IFIX(Q(22,NSEC))
  STGT = Q(24,NSEC)
  STGV = Q(23,NSEC)
  COET = Q(25,NSEC)
  DO 54 I=1,7
    TMAX(I) = Q(I+7,NSEC)
  54 TMIN(I) = Q(I+14,NSEC)
  DO 61 I=8,11
    TMAX(I) = Q(9,NSEC)
  61 TMIN(I) = Q(16,NSEC)
    TMAX(12) = Q(54,NSEC)
    TMIN(12) = Q(55,NSEC)
    HMIN = Q(39,NSEC)
    SCDEF = Q(40,NSEC)
    TRCKT = Q(41,NSEC)
    BCKPT = Q(42,NSEC)
    IPRNT = 1
    JPRNT = IFIX(Q(53,NSEC))
    CALL F0TNL(X,Y,F0,H,2)
    X0=DBLE(X)

```

```

DO 10 I=1,N
Y0(I)=DBLE(Y(I))
10 F0(I)=DBLE(F0(I))
H0=DBLE(H)
H2 = H0/2.
H3 = H0/3.
H6 = H0/6.
H8 = H0/8.
JSTG = 0
KSTG = 0
ICYC = 10
KCYC = 10
LOUT = 1
IF(TBCKT) 49,49,48
48 TBCKP = TBCKT + TS(NSEC)
GO TO 50
49 TBCKP = -TBCKT
50 COEF = SIGN(1.,COET)
IF(ISTG) 11,12,13
11 STGTV = STGV
GO TO 14
12 STGTV = STGV + TS(MSEC)
GO TO 14
**** INITIALIZATION OF THE NON STATE STAGING PARAMETERS ****
13 IF(ISTG-7) 14,58,59
58 STGV = STGV + V(10)
GO TO 14
59 QQ = Z(ISTG)
14 IF(W(7).GT..5.OR.V(66).GT.1.5.OR.(V(66).GT..5.AND.NSEC.EQ.1)) CALL
1 FAROUT
IF(ISTG) 65,15,67
67 IF(ISTG.GT.7) GO TO 64
IF(COEF*(Y(ISTG)-STGV).GT.STGT) GO TO 53
GO TO 15
64 IF(COEF*(Z(ISTG)-STGV).GT.STGT) GO TO 53
GO TO 15
65 IF(X-STGV.GT.STGT) GO TO 53
15 IMAX = 0
IMIN = 0
DO 16 I=1,N
Y1(I) = Y0(I) + H3*F0(I)
16 YS1(I)=SNGL(Y1(I))
XX1 = X + H/3.
CALL EQTNL(XX1,YS1,F1,H,0)
DO 17 I=1,N
Y2(I) = Y0(I) + H6*(F0(I) + DBLE(F1(I)))
17 YS2(I)=SNGL(Y2(I))
CALL EQTNL(XX1,YS2,F2,H,0)
DO 18 I=1,N
Y3(I) = Y0(I) + H8*(F0(I) + 3.*DBLE(F2(I)))
18 YS3(I)=SNGL(Y3(I))
XX2 = X + H/2.

```

```

      CALL EQTNL (XX2,YS3,F3,H,0)
      DO 19 I=1,N
      Y4(I) = Y0(I) + H2*(FD(I) - 3.*DBLE(F2(I))+4.*DBLE(F3(I)))
19  YS4(I)=SNGL(Y4(I))
      XX3 = X + H
      CALL EQTNL (XX3,YS4,F4,H,0)
      DO 20 I=1,N
      Y5(I) = Y0(I) + H6*(FD(I) + 4.*DBLE(F3(I))+DBLE(F4(I)))
      YS5(I)=SNGL(Y5(I))
      ARGD = DBS(.2*(Y4(I) - Y5(I)))
      ERROR(I) =SNGL (ARGD)
      IF (JSTG.GT.0) GO TO 20
      IF (ERROR(I).GT.TMAX(I)) GO TO 22
      IF (ERROR(I).GT.TMIN(I)) IMIN = 1
      GO TO 20
22  ARG = H/2.
      IF (ARG.GE.HMIN .AND. NSEC.GT.NSECX) GO TO 23
      IMAX = 1
20  CONTINUE
      IF (IMAX.GE.1) GO TO 55
56  IF (ABS(YS5(5)).LT.90.) GO TO 87
C  **** CROSSING THE POLE ****
      IF (YS5(5).LT.0.) GO TO 73
      YS5(5)=180.-YS5(5)
      Y5(5) = 180. - Y5(5)
      GO TO 74
73  Y5(5) = -180. - Y5(5)
      YS5(5)=-180.-YS5(5)
74  DO 81 L=4,6,2
      Y5(L) = Y5(L) + 180.
      YS5(L)=YS5(L)+180.
81  CONTINUE
87  IF (Q(65,NSEC).GT..5) GO TO 72
C  **** CYCLE CHECK FOR AZIMUTH AND LONGITUDE ****
      DO 82 L=4,6,2
      LQ = 75
84  LQ = LQ - 1
      IF (YS5(L).LT.0.) GO TO 85
      IF (YS5(L).LT.360.) GO TO 82
      YS5(L)=YS5(L)-360.
      Y5(L) = Y5(L) - 360.
      GO TO 86
85  Y5(L) = Y5(L) + 360.
      YS5(L)=YS5(L)+360.
86  IF (LQ.GT.0) GO TO 84
82  CONTINUE
72  CALL EQTNL (XX3,YS5,F3,H,1)
      IF (W(32).GT..5) GO TO 57
C  **** STAGING LOGIC ****
      IF (ISTG.LE.0) GO TO 27
      IF (ABS(TBCKP-XX3).LE.BCKPT.AND.COEF*(Z(ISTG)-STGV).LT.0..AND.ABS(Z
1(ISTG)-STGV).GT.STGT) GO TO 29

```

```

IF (XX3.GT.TBCKP.AND.COEF*(Z(ISTG)-STGV).LT.0..AND.ABS(Z(ISTG)-STGV
1).GT.STGT) GO TO 30
IF (KSTG.GT.0) GO TO 31
IF (ISTG-8) 41,33,33
55 WRITE(6,1001) NSEC, H, HMIN, (ERROR(I), I=1,N)
IF (W(7).GT..5.OR.V(66).GT.1.5) GO TO 75
SW2 = SW(2)
SW(?) = 1.
CALL PAROUT
SW(2) = SW2
75 IF (V(48).LT..5) GO TO 56
IF (W(38) .GT. V(48)) GO TO 57
W(38)=W(38)+1.
GO TO 56
41 IF (ABS(STGV-YS5(ISTG)).LE.STGT) GO TO 28
IF (COEF*(YS5(ISTG)-STGV).GT.0.) GO TO 34
IF (JSTG.GT.0) GO TO 35
36 X0 = X0 + H0
X=SNGL(X0)
DO 37 I=1,N
Y(I) = YS5(I)
Y0(I)=Y5(I)
F0(I) = F5(I)
37 F0(I)=DBLE(F0(I))
IF (MOD(IPRNT,JPRNT).GT.0) GO TO 60
IF (V(66).GT.1.5.OR.W(7).GT..5) CALL PAROUT
60 IPRNT = IPRNT + 1
IF (7(8).LE.-10000.) GO TO 57
IF (IMIN.GT.0) GO TO 15
IF (NSEC.GT.NSECX) M = SCOE * M
GO TO 25
57 N = 0
GO TO 999
23 H = H/2.
25 H0=DBLE(H)
H2 = H0/2.
H3 = H0/3.
H6 = H0/6.
H8 = H0/8.
GO TO 15
27 IF (ABS(STGTV-XX3).LE.STGT) GO TO 28
IF (XX3.GT.STGTV) GO TO 24
IF (JSTG) 36,36,21
24 JSTG = 1
21 IF (ICYC.LE.0) GO TO 26
H = H + STGTV - XX3
38 ICYC = ICYC - 1
GO TO 25
29 LOUT = 2
GO TO 28
30 KSTG = 1
31 IF (KCYC.GT.0) GO TO 32

```

```

      LOUT = 3
      GO TO 28
34 JSTG = 1
35 IF(ICYC.LE.0) GO TO 26
      H = H - (YS5(ISTG) - STGV)/F5(ISTG)
      GO TO 38
26 LOUT = 4
      GO TO 28
32 H = H + TBCKP - XX3
      KCYC = KCYC - 1
      GO TO 25
C **** EVALUATION OF THE NON STATE STAGING PARAMETERS ****
33 PAR = Z(ISTG)
      IF(ABS(STGV-PAR).LE.STGT) GO TO 28
      IF(COEF*(PAR-STGV).GT.0.) GO TO 44
      IF(JSTG.GT.0) GO TO 45
      QQ = PAR
      GO TO 36
44 JSTG = 1
45 IF(ICYC.LE.0) GO TO 26
      FX = (PAR - QQ)/H
      H = H - (PAR - STGV)/FX
      GO TO 38
28 X = XX3
      DO 42 I=1,N
42 Y(I) = YS5(I)
      IF(V(66).GT..5.OR.W(7).GT..5) CALL PAROUT
      GO TO (999,51,52,53), LOU
51 WRITE(6,1002) NSEC, X, TBCKP, TACKT, BCKPT, ISTG, STGV, STGT, COEF
      CALL PAROUT
      GO TO 999
52 WRITE(6,1003) NSEC, X, TBCKP, TACKT, BCKPT, ISTG, STGV, STGT, COEF
      CALL PAROUT
      GO TO 999
53 WRITE(6,1004) NSEC, X, TBCKP, TACKT, BCKPT, ISTG, STGV, STGT, COEF
      CALL PAROUT
999 RETURN
      END

```

PRECEDING PAGE BLANK NOT FILMED

GDC-08870-002

Appendix IV-22 Subroutine PAROUT

```

SUBROUTINE PAROUT
COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNAC,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
COMMON/AERO/ZP21,ZP22,ZP23
1001 FORMAT(4H ***,E16.8,5E20.8//6E20.8/6E20.8/6E20.8/6E20.8//6E20.8/6
120.8/6E20.8/6E20.8/)
1002 FORMAT(1W0/)
1003 FORMAT(6E20.8/3E20.8/)
1004 FORMAT(6E20.8/6E20.8/)
1005 FORMAT(F10.3,6F11.5)
1006 FORMAT(F10.3,3F11.1,3F11.4)
1007 FORMAT(6E20.8/6E20.8/5F20.8/6E20.8/)
1008 FORMAT(3E20.8/)
1009 FORMAT(6E20.8/)
IF( SW(2).EQ.1. .OR. SQ(1,2).NE.0. )
1WRITE(6,1001) T, TR, Z(8), Z(5), Z(6), Z(7), Z(2), Z(3), Z(4), Z(
11), Z(42), Z(65), Z(39), Z(38), Z(1), Z(51), Z(50), Z(49), Z(11),
2Z(12), Z(13), Z(43), Z(47), Z(48), ZP21, ZP22, ZP23, Z(24), Z(
35), Z(26), Z(27), Z(37), Z(8), Z(44), Z(45), Z(46), Z(29), Z(30)
4 Z(31), Z(33), Z(34), Z(35), Z(9), Z(36), Z(37), Z(40), Z(16), Z(
55), Z(10), Z(14), Z(18), Z(19), Z(23), Z(17)
IF(NSEC.GT.1) GO TO 100
IF(SQ(20,1).LE.0.) GO TO 100
IF(SW(2).EQ.1..OR.SQ(1,2).NE.0.)
1WRITE(6,1010) SQ(30,2),SQ(30,1),SQ(30,5),SQ(30,3),SQ(30,4),
2 SQ(31,1)
1010 FORMAT(10X,15HMAIN THRUST = E20.8,3X,14HAUX THRUST = E20.8/
1 3X,22HUNTHROTTLED THRUST = E20.8/
1 10X,13HMAIN WDOT = 2X,E20.8,3X,12HAUX WDOT = 2X,E20.8/
2 3X,20HUNTHROTTLED WDOT = E22.8)
1011 FORMAT(6E20.8/6E20.8/5E20.8/)
100 CONTINUE
IF(SW(2).NE.1..AND.SQ(1,2).EQ.0.) GO TO 11
IF(Q(69,NSEC).LT..5) GO TO 11
JSND = 1 + IFIX(Q(69,NSEC))
GO TO (11,21,22,23), JSND
21 WRITE(6,1008) Z(79), Z(80), Z(52)
GO TO 11
22 IF(V(69).LT..5) GO TO 21
WRITE(6,1009) Z(79), Z(80), Z(52), Z(82), Z(83), Z(81)
GO TO 11
23 IF(V(69).LT..5) GO TO 21
WRITE(6,1003) Z(79), Z(80), Z(52), Z(82), Z(83), Z(81), Z(95),
1 Z(94), Z(93)
11 IF(Q(56,NSEC).LT..5) GO TO 12
IF(Q(56,NSEC).GT.1.5) GO TO 13
IF( SW(2).EQ.1. .OR. SQ(1,2).NE.0. )
1WRITE(6,1011) Z(56), Z(67), Z(91), Z(74), Z(68), Z(72), Z(69), Z(7

```

```

20), Z(71), Z(76), Z(77), Z(73), Z(96), Z(97), Z(98), Z(99), Z(100)
GO TO 12
13 IF( SW(2).EQ.1. .OR. SQ(1,2).NE.0. )
1WRITE(6,1007) Z(66), Z(67), Z(91), Z(74), Z(68), Z(72), Z(69), Z(7
20), Z(71), Z(76), Z(77), Z(73), Z(96), Z(97), Z(98), Z(99), Z(100)
3, Z(84), Z(85), Z(86), Z(87), Z(88), Z(89)
12 IF(V(56).LT..5) GO TO 999
IF(SW(2).NE.1..AND.SQ(1,2).EQ.0.) GO TO 999
IF(V(56).GT.2.5) GO TO 10
WRITE(6,1003) Z(53), Z(54), Z(55), Z(56), Z(57), Z(58), Z(59), Z(6
10), Z(61)
IF(V(56).LT.1.5) GO TO 999
C **** TAPE OUTPUT ****
NDOT = 0.
WRITE(1) T, NDOT, (Z(I+58), I=1,3), NDOT, NDOT, NDOT, (Z(J+52), J=
11,6)
GO TO 999
10 WRITE(6,1004) Z(53), Z(54), Z(55), Z(56), Z(57), Z(58), Z(59), Z(6
10), Z(61), Z(62), Z(63), Z(64)
IF(V(56).LT.3.5) GO TO 999
C **** CARD OUTPUT ****
PUNCH 1005, T, Z(59), Z(60), Z(61), Z(62), Z(63), Z(64)
PUNCH 1006, T, Z(53), Z(54), Z(55), Z(56), Z(57), Z(58)
999 IF( SW(2).EQ.1. .OR. SQ(1,2).NE.0. )
1WRITE(6,1002)
RETURN
END

```



```

SUBROUTINE EQTNL(X,Y,F,H,IFLAG)
REAL LBW, LBX, LBY, LBZ
DIMENSION Y(12), P(4), PP(4), F(12)
DIMENSION ZP(3)
COMMON NSEC,T,TR,TS(15),V(85),O(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
COMMON/AERO/ZP21,ZP22,ZP23
1001 FORMAT(92H0 ***** EXCEEDED MAXIMUM NUMBER OF ITERATION IN DE-
1 RMINATION OF IMPACT LATITUDE *****/)
1002 FORMAT(70H0 ***** IMPACT CONDITIONS ARE INVARIANT TO IMPACT L
1 ITITUDE *****/)
1003 FORMAT(61H0 ***** PREDICTED IMPACT LATITUDE EXCEEDED LIMITS
1 *****/)
1004 FORMAT(5X,14H IMPACT LAT(?),6X,14H IMPACT LAT(1),6X,13H DELTA LAT
12),7X,13H DELTA LAT(1),7X,14H IMPACT RADIUS,6X,10H ITERATION//5E2
2.8,8X,I3///)
1005 FORMAT(92H0 **** FAILURE TO PLACE AZIMUTH TO TARGET POINT WITHIN
1 180 DEG OF AZIMUTH IN EQTNL ****/)
IEO = 0
IALF = 1
JALF = 0
TEST1 = .0000001
TEST2 = .9999999
TEST3 = .00005
C ***** STATE VARIABLES ****
T = X
DO 10 I=1,7
10 Z(I) = Y(I)
DO 61 I=1,4
61 Z(I+16) = Y(I+7)
Z(39) = Y(12)
C ***** NECESSARY Z PARAMETERS ****
ARG3 = Z(3)/V(45)
SIN3 = SIN(ARG3)
COS3 = COS(ARG3)
ARG4 = Z(4)/V(45)
SIN4 = SIN(ARG4)
COS4 = COS(ARG4)
ARG5 = Z(5)/V(45)
SIN5 = SIN(ARG5)
COS5 = COS(ARG5)
TAN5 = TAN(ARG5)
ARG6 = Z(6)/V(45)
SIN6 = SIN(ARG6)
COS6 = COS(ARG6)
WE = V(37)/V(45)
WE2 = WE*WE
TR = T - TS(NSEC)

```

```

      W(32) = .1
C   **** ROLL, PITCH, AND YAW ANGLES ****
      IF(IFLAG.LT.2) GO TO 146
      IPSIF = IFIX(Q(33,NSEC))
      PSARG = Z(43)
      IF(IPSIF.EQ.7) Z(12) = Q(45,NSEC)
146 CALL ANGLE
C   **** EARTH RADIUS GEOMETRY ****
      IF(ABS(V(49)-V(50)).LT.1.) GO TO 11
      IF(ABS(Z(5)).LT.TEST1) GO TO 11
      Z(14) = 1./SQRT((COS5/V(49))**2 + (SIN5/V(50))**2)
      ISURF = 1
      GO TO 12
11  Z(14) = V(49)
      ISURF = 0
12  Z(8) = 7(1) - 7(14)
C   **** GRAVITY FOR ASPHERICAL GRAVITATIONAL FIELD ****
      7(15) = V(36)/(7(1)*Z(1))
      DO 14 I=38,40,1
      IF(ABS(V(I)).GE.TEST1) GO TO 15
14  CONTINUE
16  Z(16) = 0.
      Z(10) = Z(15)
      GO TO 18
15  IF(ABS(Z(5)).LE.TEST1) GO TO 16
      SINS5 = SIN5*SIN5
      P(2) = -4.5*SINS5 + 1.5
      P(3) = SIN5*(6. - 10.*SINS5)
      P(4) = -21.875*SINS5*SINS5 + 18.75*SINS5 - 1.875
      PP(2) = -3.*SIN5
      PP(3) = -7.5*SINS5 + 1.5
      PP(4) = SIN5*(7.5 - 17.5*SINS5)
      ARG0 = V(49)/Z(1)
      ARG1 = 0.
      ARG2 = 0.
      DO 17 I=2,4,1
      ARGN = ARG0**I
      ARG1 = ARG1 + V(I+36)*P(I)*ARGN
17  ARG2 = ARG2 + V(I+36)*PP(I)*ARGN
      Z(16) = Z(15)*ARG2*COS5
      Z(15) = Z(15)*(1. + ARG1)
      Z(10) = SQRT(Z(15)*7(15) + Z(16)*Z(16))
C   **** ATMOSPHERE ****
18  IF(Z(8).GE.V(47).OR.Q(27,NSEC).LT..5) GO TO 19
      CALL ATMOS
      IF(W(32).GT..5) GO TO 19
      Z(40) = Z(2)/7(37)
      Z(9) = 100.8*Z(36)*Z(40)*Z(40)
      IF(Z(9).GT.SE(9).AND.NSEC.LE.3) GO TO 8
      GO TO 13
8   SE(9) = Z(9)
      SQ(5,1) = T

```

```

SQ(6,1) = Z(8)
SQ(7,1) = Z(2)
SQ(8,1) = Z(3)
SQ(9,1) = Z(40)
GO TO 13
19 Z(9) = 0.
   Z(21) = 0.
   Z(22) = 0.
   Z(23) = 0.
   Z(24) = 0.
   Z(25) = 0.
   Z(26) = 0.
   Z(29) = 0.
   Z(30) = 0.
   Z(31) = 0.
   Z(34) = 0.
   Z(36) = 0.
   Z(37) = 0.
   Z(40) = 0.
GO TO 21
C **** AERODYNAMIC COMPUTATIONS ****
17 IF(NSEC.GF.4) GO TO 201
C **** COMBINED VEHICLE AERODYNAMICS ****
DO 20 I = 1,3
  L = 1
  GO TO (22,23,24), I
22 Z(21) = 0.
   GO TO 36
23 CALL AERO2(L)
   GO TO 36
24 CALL AERO3(L)
36 Z(I+23) = Z(I+20)*Q(I+3,NSEC)*7(9)
20 CONTINUE
DO 200 I = 1,3
  L = 2
  GO TO (222,223,224), I
222 Z(21) = 0.
   GO TO 236
223 CALL AERO2(L)
   GO TO 236
224 CALL AERO3(L)
236 IFQ = 1
   Z(I+23) = Z(I+23) + Z(I+20)*Z(9)*Q(I+3,NSEC+3)
   Z(I+28) = Z(I+23)/Z(7)
   IF(I.GT.2) Z(34) = SQRT(7(29)*7(29) + Z(30)*7(30) + Z(31)*Z(31))
   IF(7(9)*Q(I+3,NSEC).EQ.0.) ZP(I) = 0.
   IF(Z(9)*Q(I+3,NSEC).EQ.0.) GO TO 2.0
   ZP(I) = Z(I+23)/(Z(9)*Q(I+3,NSEC))
200 CONTINUE
   ZP21 = 7P(1)
   ZP22 = 7P(2)
   ZP23 = 7P(3)

```

```

      GO TO 21
C  **** SINGLE VEHICLE AERODYNAMICS ****
201 DO 300 I=1,3
      IF(SV(35).GT.1.5) GO TO 301
      L = 3
      GO TO 302
301 IF(Q(I+27,NSEC).GT..5) GO TO 304
      Z(I+20) = 0.
      Z(I+23) = 0.
      Z(I+28) = 0.
      GO TO 303
304 L = IFIX(Q(I+27,NSEC))
302 GO TO (322,323,324), I
322 Z(21) = 0.
      GO TO 336
323 CALL AERO2(L)
      GO TO 336
324 CALL AERO3(L)
326 IEQ = 1
      Z(I+23) = Z(I+20)*Q(I+3,NSEC)*Z(9)
      Z(I+28) = Z(I+23)/Z(7)
303 IF(I.GT.2) Z(34) = SQRT(Z(29)*Z(29) + Z(30)*Z(30) + Z(31)*Z(31))
300 CONTINUE
      ZP21 = Z(21)
      ZP22 = Z(22)
      ZP23 = Z(23)
C  **** PROPULSION AND TIME RATE OF CHANGE IN WEIGHT ****
21 IF(Q(31,NSEC).GT..5) GO TO 70
      Z(27) = 0.
      Z(32) = 0.
      Z(47) = Q(3,NSEC)
      Z(48) = 0.
      GO TO 71
70 CALL PRPSN
      Z(32) = Z(27)/Z(7)
71 Z(28) = Z(27) + Z(25)
      Z(33) = Z(29)/Z(7)
      Z(35) = SQRT(Z(29)*Z(29) + Z(33)*Z(33) + Z(31)*Z(31))
      IF(SV(35).LE.1.6) GO TO 204
      IF(Z(35).LT.SV(42)) GO TO 204
      SV(42) = Z(35)
C  **** APPLIED FORCES ****
204 IF(Q(31,NSEC).GT..5) GO TO 72
      IF(IEQ.GT.0) GO TO 72
      Z(44) = 0.
      Z(45) = 0.
      Z(46) = 0.
      GO TO 26
72 ARG8 = Z(12)/V(45)
      SIN8 = SIN(ARG8)
      COS8 = COS(ARG8)
      IF(IPSIF.EQ.7) GO TO 101

```

```

      IF(Q(32,NSEC).LT.2.5) GO TO 100
C  **** TRANSITION FLIGHT BY ROLL-OUT CONTROL  **
      IF(IEO.LT.1.OR.Z(8).GE.V(47)) GO TO 91
      ARG = SIN8*Z(28) - COS8*Z(26)
      IF(ABS(ARG).LT.TEST1) GO TO 91
      FGAM = Z(7)*(Z(16)*SIN3*COS4 + Z(15)*COS3 - 2.*WE*Z(2)*SIN4*COS5 -
1 WE2*Z(1)*COS5*(COS3*COS5 + SIN3*COS4*SIN5) + Z(2)*(Q(52,NSEC)/V(4
25) - Z(2)*COS3/Z(1)))/V(35)
      ARG = FGAM/ARG
      CALL QAPG(ARG,Z(11),2,0)
      Z(11) = Q(36,NSEC)*Z(11)
      IF(ABS(Z(11)).GT.180.-TEST1) GO TO 90
      IF(Q(36,NSEC).LT.0.) GO TO 94
      IF(Z(11).LE.Q(44,NSEC)) GO TO 91
      GO TO 90
94 IF(Z(11).LT.Q(44,NSEC)) GO TO 90
91 Z(11) = Q(44,NSEC)
90 CALL ANGLE
      GO TO 100
C  **** TRANSITION FLIGHT BY PITCH ANGLE OF ATTACK CONTROL  ****
101 IF(JALF.GT.0) GO TO 102
      IF(IALF.GT.1) GO TO 104
      FGAM = Z(7)*(Z(16)*SIN3*COS4 + Z(15)*COS3 - 2.*WE*Z(2)*SIN4*COS5 -
1 WE2*Z(1)*COS5*(COS3*COS5 + SIN3*COS4*SIN5) + Z(2)*(Q(52,NSEC)/V(4
25) - Z(2)*COS3/Z(1)))/V(35)
      ARG7 = Z(11)/V(45)
      COS7 = COS(ARG7)
104 ARG = COS7*(SIN8*Z(28) - COS8*Z(26))
      Y3 = FGAM - ARG
      IF(ABS(Y3).LE.Q(14,NSEC)) GO TO 102
      IF(IALF=7) 106,113,116
116 IF(ABS(Y1).LT.ABS(Y2)) GO TO 107
      X1 = X2
      Y1 = Y2
107 X2 = X3
113 Y2 = Y3
114 X3 = X1 - Y1*(X2 - X1)/(Y2 - Y1)
      Z(12) = X3
      GO TO 108
106 X1 = Z(12)
      Y1 = Y3
      X2 = X1 + Q(37,NSEC)
      Z(12) = X2
108 IALF = IALF + 1
      IF(Z(12).LE.Q(35,NSEC)) GO TO 109
      IF(Z(12).GE.Q(43,NSEC)) GO TO 110
      IF(IALF=15) 105,105,115
110 Z(12) = Q(43,NSEC)
      GO TO 111
115 JALF = 1
      GO TO 105
100 Z(12) = Q(35,NSEC)

```

```

111 JALF = 1
105 IF(IEQ.LT.1.OR.7(8).GT.V(47)) GO TO 72
GO TO 13
102 CALL ANGLE
GO TO 112
100 ARG7 = Z(11)/V(45)
COS7 = COS(ARG7)
112 SIN7 = SIN(ARG7)
ARG9 = Z(13)/V(45)
SIN9 = SIN(ARG9)
COS9 = COS(ARG9)
IF(IEQ.GT.0) GO TO 73
Z(44) = (COS9*SIN8*SIN7 + SIN9*COS7)*Z(27)
7(45) = COS9*COS8*Z(27)
7(46) = (-SIN9*SIN7 + COS9*SIN8*COS7)*7(27)
GO TO 26
73 Z(44) = (COS9*COS7 - SIN9*SIN8*SIN7)*Z(24) + (COS9*SIN8*SIN7 + SIN
19*COS7)*7(28) - COS8*SIN7*Z(26)
7(45) = -SIN9*COS8*Z(24) + COS9*COS8*Z(28) + SIN8*7(26)
Z(46) = (-SIN9*SIN8*COS7 - COS9*SIN7)*Z(24) + (-SIN9*SIN7 + COS9*S
1IN8*COS7)*Z(28) - COS8*COS7*Z(26)
C **** DIFFERENTIAL EQUATIONS (STATE EQUATIONS) ****
26 F(1) = 7(2)*SIN3
F(2) = Z(16)*COS3*COS4 - 7(15)*SIN3 + Z(45)*V(35)/Z(7) + WE2*Z(1)*
1COS5*(SIN3*COS5 - COS3*COS4*SIN5)
IF(ABS(7(2)).LT.V(65)) GO TO 76
7ARG = 7(2)
GO TO 77
76 ZARG = V(65)
77 F(3) = COS3*7(2)/Z(1) + (Z(46)*V(35)/Z(7) + 2.*WE*7(2)*SIN4*COS5 +
1 WE2*Z(1)*COS5*(COS3*COS5 + SIN3*COS4*SIN5) - Z(16)*SIN3*COS4 - Z(
215)*COS3)/ZARG
F(3) = F(3)*V(45)
IF(ABS(ABS(Z(3)) - 90.).LT.TEST3) GO TO 29
IF(ABS(ABS(7(5)) - 90.).LT.TEST3) GO TO 30
IF(ABS(7(3)).GT.V(41).OR.T.LT.V(42)) GO TO 33
F(4) = (Z(2)*Z(2)*COS3*COS3*SIN4*TAN5/Z(1) - Z(16)*SIN4 + Z(44)*V(
135)/Z(7) + 2.*WE*7(2)*(COS3*SIN5 - SIN3*COS4*COS5) + WE2*Z(1)*SIN4
2*SIN5*COS5)/(7ARG*COS3)
F(4) = F(4)*V(45)
34 F(5) = 7(2)*COS3*COS4/Z(1)
F(5) = F(5)*V(45)
F(6) = Z(2)*COS3*SIN4/(Z(1)*COS5)
F(6) = F(6)*V(45)
GO TO 28
29 F(4) = 0.
F(5) = 0.
F(6) = 0.
GO TO 28
33 F(4) = 0.
GO TO 34
30 F(4) = 0.

```



```

43 DRG = Z(41)*COS(Z(42)/V(45))
   IF(ABS(DRG).GE.TEST1) GO TO 121
   Z(65) = 0.
   GO TO 120
121 COSBI = VIY/DRG
   CALL QARG(COSBI,7(65),2,0)
   IF(VIX.LT.0.) Z(65) = 360. - Z(65)
120 IF(Q(56,NSEC).LT..5) GO TO 160
C  **** ORBITAL ELEMENTS ****
C  **** CONIC ORBITAL PARAMETERS ****
   Z(75) = Z(1)*SQRT(VIX*VIX + VIY*VIY)
   Z(68) = Z(41)*Z(41) - 2.*V(36)/Z(1)
   Z(72) = Z(68)*V(79)*V(79)
   Z(74) = (Z(75)/V(36))*Z(75)
   ARG = 1. + Z(74)*Z(68)/V(36)
   IF(ARG) 127, 127, 128
128 Z(73) = SQRT(ARG)
   GO TO 129
127 Z(73) = 0.
   Z(69) = Z(41)
   Z(76) = 0.
   GO TO 125
129 Z(69) = SQRT(V(36)/Z(1))
   ARG = (Z(74)/Z(1) - 1.)/Z(73)
   CALL QARG(ARG,Z(76),2,0)
   IF(Z(42).LT.0.) Z(76) = 360. - Z(76)
125 IF(Z(68).GT.-.000001) GO TO 130
   XAX = -V(36)/Z(68)
   Z(70) = XAX*(1. + Z(73))
   ARG = V(36)*(1. - Z(73))/Z(70)
   IF(ARG.LT.TEST1) GO TO 131
   Z(71) = SQRT(ARG)
   Z(77) = 2.*V(44)*XAX*SQRT(XAX/V(36))
   GO TO 132
130 Z(70) = 0.
131 Z(71) = 0.
   Z(77) = 0.
C  **** ORBITAL INCLINATION AND NODAL LONGITUDE ****
132 IF(ABS(DRG).LT.TEST1) GO TO 41
   IF(ABS(COSBI).GT.TEST2) GO TO 42
   ARG = COS5*SIN(Z(65)/V(45))
   IF(ABS(ARG).GT.TEST2) GO TO 44
   Z(66) = V(45)*ACOS(ARG)
   ARG = COSBI/SIN(Z(66)/V(45))
   CALL QARG(ARG,ARG,2,0)
   IF(Z(66).GT.90.) GO TO 46
   IF(Z(5).LT.0.) GO TO 47
   Z(67) = Z(6) - ARG
   GO TO 50
47 Z(67) = Z(6) + ARG
   GO TO 50
46 IF(Z(5).LT.0.) GO TO 48

```



```

      Z(67) = Z(6) + ARG
      GO TO 50
48  Z(67) = Z(6) - ARG
50  IF(Z(67).LT.0.) Z(67) = Z(67) + 360.
      IF(Z(67).GT.360.) Z(67) = Z(67) - 360.
      GO TO 149
44  IF(ARG) 51,52,52
51  Z(66) = 180.
      GO TO 53
52  Z(66) = 0.
53  Z(67) = Z(6)
      GO TO 149
41  Z(66) = Z(5)
      Z(67) = Z(6)
      GO TO 149
42  IF(COSBI) 57,63,63
57  Z(67) = Z(6) + 180.
      GO TO 64
63  Z(67) = Z(6)
64  Z(66) = 90.
      GO TO 50
149 IF(ABS(COSBI).LE.TEST2) GO TO 133
      IF(COSBI.LT.0.) GO TO 134
      Z(91) = Z(5) - Z(76)
      GO TO 135
134 Z(91) = 180. - Z(5) - Z(76)
      GO TO 135
133 ARG = Z(67) - Z(6)
      CALL QUAD(ARG,ARG,1,10)
      ARG = ARG/V(45)
      ARG = COS5*COS(ARG)
      CALL QARG(ARG,ARG,2,0)
      IF(Z(5).GE.0.) ARG = 360. - ARG
      Z(91) = 360. - ARG - Z(76)
135 IF(Q(65,NSEC).LE..5) CALL QUAD(Z(91),Z(91),1,10)
      Z(96) = 7(74)/(1. + Z(73))
      ARG = SIN(Z(66)/V(45))*SIN(Z(91)/V(45))
      CALL QARG(ARG,Z(97),1,0)
      Z(98) = -Z(97)
      ARG2 = ARG*ARG
      ARG1 = 1. - ARG2
      REA = 1./SQRT(ARG1/(V(49)*V(49)) + ARG2/(V(50)*V(50)))
      Z(99) = Z(96) - REA
      IF(Z(68).GT.-.000001) GO TO 203
      Z(100) = Z(70) - REA
      GO TO 78
203 Z(100) = 0.
      78 CONTINUE

```

C
C
C
C

OPTION FOR TARGETING ORBITER TO APOGEE, PERIGEE, TRUE ANOMALY

```

      IF(SQ(3,1).LT.0.) GO TO 88
      RAT=REA + SQ(3,1)
      AT = REA + .5 * (SQ(3,1) + SQ(3,2) )
      SQ(3,1) IS APOGEE ALT, SQ(3,2) IS PERIGEE ALT
      ET = RAT/AT -1.
      RT = RAT*(1.-ET)/(1.+ET*SQ(3,4))
      HT = RT - Z(14)
      SQRTK= SQRT(V(36))
      VT = SQRTK*SQRT(2./RT-1./AT)
      VAT= SQRTK*SQRT(2./RAT-1./AT)
      GAMT=ACOS (VAT*RAT/VT/RT ) *V(45)
      IF(SQ(3,5).LT.0.) GAMT=-GAMT
      Z(101)= Z( 8) - HT
      Z(102)= Z(42) - GAMT
      Z(103)= Z(41) - VT
88 CONTINUE
      THE VARIABLES HT,GAMT,VT ARE TARGET VALUES DERIVED FROM THE
      INPUTS HANM, HPNM, TRANOM, AND THE CURRENT VALUE OF RE=Z(14).
      OPTION FOR TARGETING ORBITER TO APOGEE,PERIGEE, TRUE ANOMALY

      IF(Q(56,NSEC).LT.1.5) GO TO 160
      **** INSTANTANEOUS IMPACT POINT PREDICTION ****
      IF(Z(73).LT..0001.OR.Z(73).GT..9999.OR.Z(70).LT.V(50)) GO TO 161
      XRP = XAX*(1. - Z(73))
      IF(XRP.GT..999*V(50)) GO TO 161
      Z6 = Z(6)
      IF(Q(65,NSEC).GT..5) CALL QUAD(Z6,Z6,1,10)
      IF(ISURF.LT.1) GO TO 164
      IF(Q(56,NSEC).GT.2.5) GO TO 199
      **** ITERATION INITIALIZATION FOR OBLATE EARTH IMPACT LATITUDE ****
      JERR = 0
      LCYC = 10
      KCYC = 0
      IF(IFLAG.EQ.2) Z(84) = SIGN(30.,Z(5))
      Z84 = Z(84)
162 ARG1 = COS(Z(84)/V(45))
      ARG1 = ARG1*ARG1
      ARG2 = 1. - ARG1
      Z(86) = 1./SQRT(ARG1/(V(49)*V(49)) + ARG2/(V(50)*V(50)))
      KCYC = KCYC + 1
      GO TO 163
      **** SPHERICAL EARTH ****
199 Z(86) = V(1)
      GO TO 163
164 Z(86) = V(49)
      **** INITIAL CONIC ANALYSIS ****
163 ARG = Z(74)/Z(86)
      ARG = (ARG - 1.)/Z(73)
      CALL OARG(ARG,Z(90),2,0)
      Z(90) = 360. - Z(90)
      Z(89) = Z(90) - Z(76)

```

```

      IF(ABS(Z(66)-90.).LT.TEST1.OR.ABS(Z(5)).GT.90.-TEST1) GO TO 166
C   **** NON-POLAR INCLINATIONS ****
      IF(Z(5).LT.0.) GO TO 167
C   **** POSITIVE CURRENT LATITUDE ****
      XKF2 = 1.
      XPA = Z(67) - 180.
      CALL QUAD(XPA,XPA,1,10)
172  DPH = XPA - 76
      IF(Z(65).GE.180.) DPH = -DPH
      CALL QUAD(DPH,DPH,1,10)
      GO TO 169
C   **** NEGATIVE CURRENT LATITUDE ****
167  XKF2 = -1.
      XPA = Z(67)
      GO TO 172
169  ARG = COS5*COS(DPH/V(45))
      CALL QAR8(ARG,XCC,2,0)
      ARG = XCC - Z(89)
      IF(ARG.GE.0.) GO TO 174
      IF(ARG.LE.-180) GO TO 175
      XDD = -ARG
      XKF1 = -1.
      XKF3 = 1.
      GO TO 176
174  XDD = ARG
      XKF3 = 0.
      GO TO 178
175  XDD = -ARG - 180.
      XKF3 = 2.
170  XKF1 = 1.
176  ARG = XKF1*XKF2*SIN(Z(66)/V(45))*SIN(XDD/V(45))
      CALL QAR8(ARG,Z(84),1,0)
      GO TO 178
C   **** POLAR INCLINATION ****
166  IF(ABS(Z(65)-180.).LT.90.) GO TO 179
C   **** INERTIAL AZIMUTH IS 0 OR 360 DEG ****
      XKF1 = 1.
      ARG1 = Z(5) + Z(89)
      GO TO 180
C   **** INERTIAL AZIMUTH IS 180 DEG ****
179  XKF1 = -1.
      ARG1 = Z(5) - Z(89)
180  AARG1 = ABS(ARG1)
      IF(AARG1.LT.90.) GO TO 181
      IF(AARG1.GE.270.) GO TO 182
      Z(84) = XKF1*180. - ARG1
      GO TO 178
181  Z(84) = ARG1
      GO TO 178
182  Z(84) = ARG1 - XKF1*360.
178  IF(ISURF.LT.1.OR.Q(56,NSEC).GT.2.5) GO TO 177
C   **** NEWTON-RAPHSON ITERATION FOR OBLATE EARTH IMPACT LATITUDE ****

```

```

D7B84 = Z(84) - Z84
IF(ABS(DZB84).LE.0(19,NSEC)) GO TO 177
IF(KCYC.GT.LCYC) GO TO 183
IF(KCYC-2) 187,187,184
186 JERR = 1
187 Z84 = Z(84)
DZA84 = DZB84
GO TO 162
184 ARG = DZB84 - DZA84
IF(ABS(ARG).GE.TEST1) GO TO 188
IF(JERR) 186,186,185
188 ARG = Z(84) - D7B84*DZB84/ARG
Z84 = Z(84)
DZA84 = DZB84
Z(84) = ARG
IF(ABS(Z(84)).GT.100.) GO TO 189
GO TO 162
C **** EXCEEDED MAXIMUM NUMBER OF ITERATIONS ****
183 WRITE(6,1001)
GO TO 190
C **** IMPACT CONDITIONS ARE INVARIANT TO IMPACT LATITUDE ****
185 WRITE(6,1002)
GO TO 190
C **** PREDICTED IMPACT LATITUDE EXCEEDED LIMITS ****
189 WRITE(6,1003)
190 WRITE(6,1004) Z(84), Z84, D7B84, DZA84, Z(86), KCYC
C **** DETERMINATION OF IMPACT LONGITUDE FOR A NON-ROTATING CENTRAL BO
177 IF(ABS(Z(66)-90.).LT.TEST1.OR.ABS(Z(5)).GT.90.-TEST1) GO TO 191
C **** NON-POLAR INCLINATIONS ****
ARG1 = COS(XDD/V(45))
ARG2 = COS(Z(84)/V(45))
ARG = ARG1/ARG2
CALL QARG(ARG,ARG,2,0)
IF(XKF3.LT..5) GO TO 192
IF(XKF3.GT.1.5) GO TO 193
GO TO 194
192 ARG = -ARG
GO TO 194
193 ARG = ARG + 180.
194 IF(Z(65).GT.180.) ARG = -ARG
Z(88) = XPA + ARG
GO TO 195
C **** POLAR INCLINATION ****
191 IF(AARG1.LT.90..OR.AARG1.GE.270.) GO TO 196
Z(88) = Z6 + 180.
GO TO 195
196 Z(88) = Z6
GO TO 197
195 CALL QUAD(Z(88),Z(88),1,10)
C **** DETERMINATION OF TIME TO IMPACT ****
197 ARG = SQRT((1.-Z(73))/(1.+Z(73)))
ARG1 = .5/V(45)

```

```

ARG2 = ARG*TAN(ARG1*Z(76))
CALL DARG(ARG2,XE1,3,1)
IF(XE1.LT.0.) XE1 = V(44) + XE1
XE1 = 2.*XE1
ARG2 = ARG*TAN(ARG1*Z(90))
CALL DARG(ARG2,XE2,3,1)
IF(XE2.LT.0.) XE2 = V(44) + XE2
XE2 = 2.*XE2
SINE1 = SIN(XE1)
SINE2 = SIN(XE2)
Z(87) = .5*(XE2 - XE1 - Z(73)*(SINE2 - SINE1))*Z(77)/V(44)
C **** DETERMINATION OF IMPACT LONGITUDE FOR A ROTATING CENTRAL BODY
Z(85) = Z(88) - V(37)*Z(87)
CALL DQAD(Z(85),Z(85),1,10)
GO TO 160
161 DO 198 I=1,7
198 Z(I+83) = 0.
160 IF(V(56).LT..5) GO TO 60
IF(V(56).GT.2.5) GO TO 40
C **** TRANSFORMATIONS FOR INPUT COMPATIBILITY WITH GENBET AND TSOP PR
CALL TRAN1(0.,Z(2),0.,SIN3,COS3,SIN4,COS4,V1,V2,V3)
ARG = V(35)/Z(7)
AX = ARG*Z(44)
AY = ARG*Z(45)
AZ = ARG*Z(46)
CALL TRAN1(AX,AY,AZ,SIN3,COS3,SIN4,COS4,AU,AV,AW)
AV = AV + Z(16)
AW = AW - Z(15)
CALL TRAN2(0.,0.,Z(1),SIN5,COS5,SIN6,COS6,Z(53),Z(54),Z(55))
CALL TRAN2(V1,V2,V3,SIN5,COS5,SIN6,COS6,Z(56),Z(57),Z(58))
CALL TRAN2(AU,AV,AW,SIN5,COS5,SIN6,COS6,AX,AY,AZ)
AAA = V(6) - V(37)*T
ARGA = AAA/V(45)
SINA = SIN(ARGA)
COSA = COS(ARGA)
CALL TRAN3(AX,AY,AZ,SINA,COSA,W(25),W(26),W(27),W(28),Z(59),Z(60),
1Z(61))
GO TO 60
C **** TRANSFORMATIONS FOR INERTIAL COMPONENTS OF R, V, A(F), A(G) **
40 ARG = Z(42)/V(45)
SINGI = SIN(ARG)
COSGI = COS(ARG)
ARG = Z(65)/V(45)
SINBI = SIN(ARG)
COSBI = COS(ARG)
CALL TRAN1(0.,Z(41),0.,SINGI,COSGI,SINBI,COSBI,VX,VY,VZ)
ARG = V(35)/Z(7)
AX = ARG*Z(44)
AY = ARG*Z(45)
AZ = ARG*Z(46)
CALL TRAN1(AX,AY,AZ,SIN3,COS3,SIN4,COS4,AU,AV,AW)
CALL TRAN2(0.,0.,Z(1),SIN5,COS5,SIN6,COS6,R1,R2,R3)

```

```

CALL TRAN2(VX,VY,VZ,SIN5,COS5,SIN6,COS6,V1,V2,V3)
CALL TRAN2(AU,AV,AW,SIN5,COS5,SIN6,COS6,AX,AY,AZ)
CALL TRAN2(0.,Z(16),-Z(15),SIN5,COS5,SIN6,COS6,B1,B2,B3)
AAA = V(6) - V(37)*T
ARGA = AAA/V(45)
SINA = SIN(ARGA)
COSA = COS(ARGA)
CALL TRAN3(R1,R2,R3,SINA,COSA,W(25),W(26),W(27),W(28),Z(53),Z(54),
1Z(55))
CALL TRAN3(V1,V2,V3,SINA,COSA,W(25),W(26),W(27),W(28),Z(56),Z(57),
17(58))
CALL TRAN3(AX,AY,AZ,SINA,COSA,W(25),W(26),W(27),W(28),Z(59),Z(60),
1Z(61))
CALL TRAN3(B1,B2,B3,SINA,COSA,W(25),W(26),W(27),W(28),Z(62),Z(63),
17(64))

```

C **** DOWN RANGE COMPUTATION ****

```

60 ARK1 = W(21)*W(23)
   ARK2 = W(21)*W(22)
   ARK5 = COS5*COS6
   ARK6 = COS5*SIN6
   COS49 = ARK1*ARK5 + ARK2*ARK6 + W(20)*SIN5
   CALL QARG(COS49,Z(49),2,0)
   ARG = ARK1*ARK6 - ARK5*ARK2
   DUM = V(4) - 180.
   IF(DUM*ARG.GT.0.) Z(49) = 360. - Z(49)
   IF(V(68).LT..5.AND.Q(69,NSEC).LT..5) GO TO 85

```

C **** AZIMUTH FROM INITIAL POINT ****

```

   LBX = ARK2*SIN5 - ARK6*W(20)
   LBY = ARK5*W(20) - ARK1*SIN5
   LBZ = ARK1*ARK6 - ARK5*ARK2
   LBW = LBX*W(22) - LBY*W(23)
   XLB = SQRT(LBX*LBX + LBY*LBY + LBZ*LBZ)
   IF(XLB.LT.TEST1) GO TO 84
   Z(52) = V(45)*ACOS(LBW/XLB)
   Z52 = Z(52)
   IF(LBZ.LT.0.) Z(52) = 360. - Z(52)
   GO TO 85

```

```

84 Z(52) = V(4)
   Z52 = Z(52)

```

C **** SLANT RANGE AND ELEVATION ANGLE FROM INITIAL POINT ****

```

95 IF(Q(69,NSEC).LT..5) GO TO 151
   XL1 = Z(1)*ARK5 - V(1)*ARK1
   XL2 = Z(1)*ARK6 - V(1)*ARK2
   XL3 = Z(1)*SIN5 - V(1)*W(20)
   Z(79) = SQRT(XL1*XL1 + XL2*XL2 + XL3*XL3)
   IF(Z(79).LT.TEST1) GO TO 157
   ARG = ARK1*XL1 + ARK2*XL2 + W(20)*XL3
   ARG = ARG/Z(79)
   CALL QARG(ARG,ARG,2,0)
   Z(80) = 90. - ARG
   GO TO 151

```

```

157 Z(80) = 0.

```

```

151 IF(V(68).LT..5) GO TO 999
C **** CROSS RANGE COMPUTATION ****
  Z(50) = Z52 - W(24)
C **** TARGET MISS ANGLE COMPUTATION ****
  ARK3 = W(14)*W(16)
  ARK4 = W(14)*W(15)
  BDT = ARK5*ARK3 + ARK6*ARK4 + SIN5*W(13)
  IF(ABS(BDT).GT..95) GO TO 86
  Z(51) = V(45)*ACOS(BDT)
  GO TO 150
86 BTX = ARK6*W(13) - ARK4*SIN5
  BTY = ARK3*SIN5 - ARK5*W(13)
  BTZ = ARK5*ARK4 - ARK3*ARK6
  S751 = SQRT(BTX*BTX + BTY*BTY + BTZ*BTZ)
  CALL QARG(SZ51,Z(51),1,0)
  IF(BDT.LT.0.) Z(51) = 180. - Z(51)
150 IF(Q(69,NSEC).LT.1.5) GO TO 999
C **** AZIMUTH, SLANT RANGE, AND ELEVATION ANGLE FROM TARGET POINT *
  BTX = ARK4*SIN5 - W(13)*ARK6
  BTY = W(13)*ARK5 - ARK3*SIN5
  BTZ = ARK3*ARK6 - ARK4*ARK5
  BTW = BTX*W(15) - BTY*W(16)
  BDT = SQRT(BTX*BTX + BTY*BTY + BTZ*BTZ)
  IF(BDT.LT.TEST1) GO TO 154
  Z(81) = V(45)*ACOS(BTW/BDT)
  GO TO 158
154 Z(81) = 0.
158 IF(BT7.LT.0.) Z(81) = 360. - Z(81)
  XL1 = Z(1)*ARK5 - W(35)*ARK3
  XL2 = Z(1)*ARK6 - W(35)*ARK4
  XL3 = Z(1)*SIN5 - W(35)*W(13)
  Z(82) = SQRT(XL1*XL1 + XL2*XL2 + XL3*XL3)
  IF(Z(82).LT.TEST1) GO TO 159
  ARG = ARK3*XL1 + ARK4*XL2 + W(13)*XL3
  ARG = ARG/Z(82)
  CALL QARG(ARG,ARG,2,0)
  Z(83) = 90. - ARG
  GO TO 145
159 Z(83) = 0.
145 IF(Q(69,NSEC).LT.2.5) GO TO 999
C **** AZIMUTH AND ELEVATION ANGLE OF THE TARGET POINT FROM THE VEHICLE
  Z(95) = (Z(51)/V(45))*((W(35) - V(80))/V(46))
  BTW = BTY*COS6 - BTX*SIN6
  IF(BDT.LT.TEST1) GO TO 138
  Z(93) = V(45)*ACOS(BTW/BDT)
  GO TO 139
138 Z(93) = 0.
139 IF(BTZ.GT.0.) Z(93) = 360. - Z(93)
  IF(Q(65,NSEC).LT..5) GO TO 147
  AQD = Z(4) - 360.
  BQD = Z(4) + 360.
  NQ = 1

```

```
137 IF(Z(93).LT.A00) GO TO 142
    IF(Z(93).LT.B00) GO TO 147
    Z(93) = Z(93) - 360.
    GO TO 136
142 Z(93) = Z(93) + 360.
136 IF(NQ.GT.10) GO TO 998
    NQ = NQ + 1
    GO TO 137
998 WRITE(6,1005)
147 Z(94) = Z(93) - Z(4)
    IF(F(4).LT.0..AND.Z(94).GT. 60.) Z(94)=Z(94)-360.
    IF(F(4).GT.0..AND.Z(94).LT.-60.) Z(94)=Z(94)+360.
999 RETURN
END
```


Appendix IV-24 Subroutine ANGLE

```

SUBROUTINE ANGLE
COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VO(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 Q2(25,12,5),XH3(25,5),A3(12,5),Q3(25,12,5),XNA9,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
COMMON/LTS/WH(5)
1001 FORMAT(83H0 **** FAILURE IN THE TABLE LOOK-UP OF SUBROUTINE ANGL
1E DURING SIMULATION SECTION,I3,7H ****///)
W(31) = .1
TEST1 = .0000001
TEST2 = .9999999
IF(W(6).GT..5) GO TO 45
C **** INITIALIZATION ****
ISEND = IFIX(Q(33,NSEC))
KSEND = IFIX(Q(32,NSEC))
LSEND = IFIX(Q(34,NSEC))
C **** ROLL RATE CONTROL ****
GO TO (41,42,44,44,44), KSEND
45 GO TO (46,46,44,44,44), KSEND
41 SIGO = Q(44,NSEC)
GO TO 46
42 SIGO = Z(11) + Q(44,NSEC)
46 Z(11) = Q(36,NSEC)*TR + SIGO
C **** YAW ANGLE OF ATTACK RATE CONTROL ****
44 IF(W(6).GT..5) GO TO 49
GO TO (47,48,83,84,47), LSEND
47 XLM0 = Q(46,NSEC)
GO TO 49
48 XLM0 = Z(13) + Q(46,NSEC)
49 Z(13) = Q(38,NSEC)*TR + XLM0
GO TO 82
C **** SPECIFIED ROLL AXIS AZIMUTH ****
83 XDLO = Q(46,NSEC)
GO TO 82
84 IF(ABS(Z(13)).LT.TEST1) GO TO 87
PSIP = 90. - Z(12) - Z(3)
IF(ABS(PSIP).LT.TEST1.OR.ABS(180.-PSIP).LT.TEST1) GO TO 83
ARG = 90. - ABS(Z(13))
IF(ABS(ARG).LT.TEST1) GO TO 88
ARG = TAN(Z(13)/V(45))/SIN(PSIP/V(45))
CALL QARG(ARG,XLMH,3,0)
CALL QUAD(Z(13),XLM0,2,10)
IF(ABS(XLM0).LE.90.) GO TO 85
XLMH = 180. - XLMH
CALL QUAD(XLMH,XLMH,2,10)
GO TO 85
88 XLMH = Z(13)
85 XDLO = Z(4) + XLMH + Q(46,NSEC)
C **** PITCH ANGLE OF ATTACK RATE CONTROL ****
82 IF(W(6).GT..5) GO TO 52

```

```

      GO TO (50,51,54,50,51,53,30,61,67,61,67,61,57,51,67,61,67,90,90,
190,90,100,101,100,161,100,101,100,101,100,101), ISEND
52 GO TO (29,29,26,29,29,26,30,62,62,62,62,62,62,62,62,62,91,91,
191,91,110,110,110,110,110,110,110,110,110,110), ISEND
50 ALP0 = Q(45,NSEC)
      GO TO 29
51 ALP0 = Z(12) + Q(45,NSEC)
29 Z(12) = Q(37,NSEC)*TR + ALP0
      GO TO 80
100 ALP0 = Q(45,NSEC)
      GO TO 102
101 ALP0 = Z(12) + Q(45,NSEC)
102 ITBL = (ISEND-20)/2
      ITRIP = 2
      JMAX = IFIX(XNPT(ITBL))
      IF(NSEC-1) 103, 103, 110
103 Z(8) = V(8)
      CALL ATMOS
      IF(Z(37)-TEST1) 104, 104, 105
104 Z(40) = 0.
      GO TO 110
105 Z(40) = Z(2)/Z(37)
110 IF(Z(40).GT.PT(ITRIP,ITBL)) GO TO 111
      JSTOP = ITRIP - 1
      DO 114 J=1,JSTOP
      I = ITRIP - J + 1
      IF(Z(40).GE.PT(I-1,ITBL)) GO TO 113
114 CONTINUE
      GO TO 997
111 JSTOP = JMAX - ITRIP
      IF(JSTOP.LE.0) GO TO 997
      DO 112 J=1,JSTOP
      I = ITRIP + J
      IF(Z(40).LE.PT(I,ITBL)) GO TO 113
112 CONTINUE
997 ITRIP = 2
      Z(12) = ALP0
      WRITE(6,1001) NSEC
      W(32) = 1.1
      CALL PAROUT
      W(32) = 1.1
      GO TO 999
113 ITRIP = 1
      ARG = (Z(40) - PT(I-1,ITBL))/(PT(I,ITBL) - PT(I-1,ITBL))
      Z(12) = ALP0 + Q(37,NSEC)*TR + PDT(I-1,ITBL) + ARG*(PDT(I,ITBL) -
1 PDT(I-1,ITBL))
      GO TO 80
90 KA = IFIX(WW(5))
      IF(NSEC.GT.KA) GO TO 91
      KARG = ISEND - 17
      GO TO (92,93,94,95), KARG
92 ARG = Q(43,KA)

```

```

GO TO 96
93 ARG = 90. - Z(3)
   GO TO 96
94 ARG = 7(43)
96 ARG = (ARG + Q(45,KAI))/V(45)
   IF (ABS(ARG).LT.TEST1) GO TO 74
   WW(3) = 1./TAN(ARG)
   GO TO 75
74 WW(3) = V(44)*.5
75 IF (ARS(Q(35,KAI)).LT.TEST1) GO TO 97
   ARG = Q(35,KAI)/V(45)
   ARG = 1./TAN(ARG)
   GO TO 98
97 ARG = V(44)*.5
98 WW(4) = (ARG - WW(3))/(WW(2) - WW(1))
   Q(37,KAI) = WW(2) - WW(1)
   GO TO 91
95 WW(3) = Q(43,KAI)
   WW(4) = Q(45,KAI)
91 ARG = WW(3) + WW(4)*(T-WW(1))
   IF (ABS(ARG).LT.TEST1) GO TO 76
   ARG = 1./ARG
   CALL QARG(ARG,7(43),3,0)
   IF (7(43).LT.0.) Z(43) = Z(43) + 180.
   GO TO 77
76 Z(43) = 90.
77 Z(12) = 90. - 7(3) - Z(43)
80 IF (LSEND.LT.3) GO TO 69
C **** SPECIFIED ROLL AXIS AZIMUTH ****
   XLMH = Q(38,NSEC)*TR + XDLD - Z(4)
   CALL QUAD(XLMH,XLMH,2,10)
   PSIP = 90 - Z(12) - 7(3)
   IF (ARS(PSIP).LT.TEST1.OR.ABS(180.-PSIP).LT.TEST1) GO TO 86
   ARG = 90. - ABS(XLMH)
   IF (ABS(ARG).LT.TEST1) GO TO 86
   ARG = SIN(PSIP/V(45))*TAN(XLMH/V(45))
   CALL QARG(ARG,Z(13),3,0)
   IF (ABS(XLMH).LE.90.) GO TO 87
   Z(13) = 180. - 7(13)
   CALL QUAD(Z(13),Z(13),2,10)
   GO TO 87
86 Z(13) = XLMH
87 Z(11) = 0.
69 GO TO (30,30,999,999,999,999,30,30,30,30,30,30,30,30,30,30,30,30,70,
      1 70,70,70,30,30,30,30,30,30,30,30,30), ISEND
C **** COMPUTATION OF PSI (PITCH ATTITUDE) ****
30 IF (ABS(Z(11)).LT.TEST1) GO TO 12
C **** NON-ZERO ROLL ANGLE ****
   ARG6 = (90. - 7(3))/V(45)
   ARG1 = Z(11)/V(45)
   ARG2 = Z(12)/V(45)
   COSI = COS(ARG1)

```

```

SIN2 = SIN(ARG2)
COS2 = COS(ARG2)
SIN6 = SIN(ARG6)
COS6 = COS(ARG6)
IF(ABS(Z(12)).LT.TEST1) GO TO 60
ARGT = COS6*COS2 + SIN6*SIN2*COS1
CALL QARG(ARGT,ARGS,2,0)
IF(ABS(Z(13)).LT.TEST1) GO TO 19
C **** NON-ZERO ROLL ANGLE, NON-ZERO YAW ANGLE ****
ARG3 = Z(13)/V(45)
COS3 = COS(ARG3)
ARG7 = ARGS/V(45)
SIN7 = SIN(ARG7)
COS7 = ARG7
ARG0 = SIN7*SIN2
ARGN = COS6 - COS7*COS2
C **** TRANSCENDENTAL FUNCTION ARGUMENT CHECK ****
IF(ABS(ARG0).LE.ABS(ARGN)-TEST1) GO TO 21
SIN3 = SIN(ARG3)
COS5 = ARGN/ARG0
ARG5 = ACOS(COS5)
SIN5 = SIN(ARG5)
APGT = COS7*COS3 - SIN7*SIN3*SIN5
GO TO 20
21 ARGT = COS7*COS3
GO TO 20
C **** NON-ZERO ROLL ANGLE, ZERO PITCH ANGLE OF ATTACK ****
60 SIN1 = SIN(ARG1)
ARG3 = Z(13)/V(45)
SIN3 = SIN(ARG3)
COS3 = COS(ARG3)
ARG = COS6*COS3 - SIN6*SIN3*SIN1
Z(43) = V(45)*ACOS(ARG)
GO TO 999
C **** NON-ZERO ROLL ANGLE, ZERO YAW ANGLE ****
19 Z(43) = ARGS
GO TO 999
C **** ZERO ROLL ANGLE ****
12 IF(ABS(Z(13)).LT.TEST1) GO TO 13
C **** ZERO ROLL ANGLE, NON-ZERO YAW ANGLE ****
PAR8 = 90. - Z(3) - Z(12)
COS3 = COS(Z(13)/V(45))
COS6 = COS(PAR8/V(45))
ARGT = COS6*COS3
20 CALL QARG(ARGT,Z(43),2,0)
GO TO 999
C **** ZERO ROLL ANGLE, ZERO YAW ANGLE ****
13 Z(43) = 90. - Z(3) - Z(12)
GO TO 999
C **** PSI-DOT CONTROL, ZERO ROLL ANGLE ****
54 PSID = Z(43) + Q(43,NSEC)
GO TO 26

```

```

53 PSIO = Q(43,NSEC)
26 Z(43) = Q(35,NSEC)*Q(62,NSEC)*TR + PSIO
   Z(78) = Q(35,NSEC)*Q(62,NSEC)
   GO TO 70
C **** PITCH RATE TABLE ****
61 PSFLG = 1.
   GO TO 68
67 PSFLG = 0.
68 KSEC = IFIX(Q(63,NSEC))
   PST(1) = Q(43,NSEC)
   IF(PSFLG.GT..5) PST(1) = PST(1) + PS(KSEC)
   ITBL = (ISEND - 6)/2
   ITRIP = 2
   JMAX = IFIX(XNPT(ITBL))
   IF(KSEC.LT.1) KSEC = 1
   DO 73 L=2,JMAX
73 PST(L) = Q(62,NSEC)*PDT(L-1,ITBL)*(PT(L,ITBL) - PT(L-1,ITBL)) +
   1 PST(L-1)
62 ARG = T - TS(KSEC) - Q(64,NSEC)
   IF(ARG.LT.PT(1,ITBL).OR.ARG.GT.PT(JMAX,ITBL)) GO TO 71
   IF(ARG.GT.PT(ITRIP,ITBL)) GO TO 72
   JSTOP = ITRIP - 1
   DO 65 J=1,JSTOP
   I = ITRIP - J + 1
   IF(ARG.GE.PT(I,ITBL)) GO TO 64
65 CONTINUE
998 WRITE(6,1001) NSEC
   W(32) = 1.
71 Z(12) = Q(45,NSEC)
   W(31) = 1.1
   GO TO 80
72 JSTOP = JMAX - ITRIP
   IF(JSTOP.LE.0) GO TO 998
   DO 63 J=1,JSTOP
   I = ITRIP + J
   IF(ARG.LE.PT(I,ITBL)) GO TO 64
63 CONTINUE
   GO TO 998
64 ITRIP = I
   Z(43) = PST(I-1) + Q(62,NSEC)*PDT(I-1,ITBL)*(ARG - PT(I-1,ITBL))
   Z(78) = PDT(I-1,ITBL)*Q(62,NSEC)
70 Z(11) = 0.
   IF(ISEND.LT.3) GO TO 81
C **** SPECIFIED ROLL AXIS AZIMUTH ****
   XLMH = Q(38,NSEC)*TR + XDLO - Z(4)
   CALL QUAD(XLMH,XLMH,2,10)
   IF(ABS(XLMH).LT.TEST1.OR.ABS(XLMH).GT.180.-TEST1) GO TO 89
   ARG = 90. - ABS(XLMH)
   IF(ABS(ARG).LT.TEST1) GO TO 89
   ARG = SIN(XLMH/V(45))*SIN(Z(43)/V(45))
   CALL QARG(ARG,Z(13),1,0)
   IF(ABS(XLMH).LE.90.) GO TO 81

```

```

      Z(13) = 180. - Z(13)
      CALL QUAD(Z(13),7(13),2,10)
      GO TO 81
89 Z(13) = XLMM
81 IF(ABS(7(13)).GT.TEST1) GO TO 25
C **** ZERO ROLL ANGLE, ZERO YAW ANGLE ****
      7(12) = 90. - 7(3) - Z(43)
      GO TO 999
C **** ZERO ROLL ANGLE, NON-ZERO YAW ANGLE ****
25 ARG3 = 7(13)/V(45)
      COS3 = COS(ARG3)
      ARG8 = 7(43)/V(45)
      COS8 = COS(ARG8)
C **** TRANSCENDENTAL FUNCTION ARGUMENT CHECK ****
      IF(ABS(COS8).LE.ABS(COS3)-TEST1) GO TO 22
      IF(ABS(COS8).GT.ABS(COS3)+TEST1) GO TO 999
      IF(SIGN(COS8,COS3).LT.0.) GO TO 23
      Z(12) = 90. - Z(3)
      GO TO 999
23 Z(12) = -90. - Z(3)
      GO TO 999
22 7(12) = 90. - Z(3) - V(45)*ACOS(COS8/COS3)
C **** EXIT ****
999 W(6) = 1.1
      RETURN
      END

```

Appendix IV-25 Subroutine ATMOS

```

SUBROUTINE ATMOS
COMMON/APV/ALT(200),PRS(200),VLS(200)
COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
1001 FORMAT(83H0 **** FAILURE IN THE TABLE LOOK-UP OF SUBROUTINE ATM
1S DURING SIMULATION SECTION,I3,7H ****///)
IF(W(1).GT..5) GO TO 10
W(1) = 1.1
ITRIP = 2
JMAX = IFIX(XNA0)
10 IF(Z(8).GT.ALT(ITRIP)) GO TO 11
JSTOP = ITRIP - 1
DO 14 J=1,JSTOP
I = ITRIP - J + 1
IF(Z(8).GE.ALT(I-1)) GO TO 13
14 CONTINUE
GO TO 998
11 JSTOP = JMAX - ITRIP
IF(JSTOP.LE.0) GO TO 998
DO 12 J=1,JSTOP
I = ITRIP + J
IF(Z(8).LE.ALT(I)) GO TO 13
12 CONTINUE
998 W(1) = .1
Z(36) = 0.
Z(37) = 0.
WRITE(6,1001) NSEC
CALL PAROUT
W(32) = 1.1
GO TO 999
13 ITRIP = I
ARG = (Z(8) - ALT(I-1))/(ALT(I) - ALT(I-1))
Z(36) = PRS(I-1) + ARG*(PRS(I) - PRS(I-1))
Z(37) = VLS(I-1) + ARG*(VLS(I) - VLS(I-1))
999 RETURN
END

```

GDC-08870-002

Appendix IV-26 Subroutine AERO1

C SUBROUTINE AERO1
 DUMMY SUBROUTINE FOR YAW AERODYNAMICS
 RETURN
 END


```

SUBROUTINE AERO2(M)
  DIMENSION D(12)
  DIMENSION ITRP(5),KTRP(5),ISND(5),JMX(5),LMX(5),XW3(5),XW4(5)
  COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),H(50),Z(120),EC(3),CV(3),
  1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
  2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNAG,XNA1(1,1),
  3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
  4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
  5 PS(15)
  COMMON/AEROX/XW3,XW4
1001 FORMAT(83H0 **** FAILURE IN THE TABLE LOOK-UP OF SUBROUTINE AERO
  12 DURING SIMULATION SECTION,I3,7H ****//)
  IF(XW3(M).GT..5) GO TO 10
  XW3(M) = 1.1
  ITRP(M) = 2
  KTRP(M) = 2
  ISND(M) = M
  JMX(M) = XNA2(1,M)
  LMX(M) = XNA2(2,M)
  10 ITRIP = ITRP(M)
  KTRIP = KTRP(M)
  ISEND = ISND(M)
  JMAX = JMX(M)
  LMAX = LMX(M)
  IF(Z(40).GT.XM2(ITRIP,ISEND)) GO TO 11
  JSTOP = ITRIP - 1
  DO 14 J=1,JSTOP
  I = ITRIP - J + 1
  IF(Z(40).GE.XM2(I-1,ISEND)) GO TO 13
  14 CONTINUE
  GO TO 998
  11 JSTOP = JMAX - ITRIP
  IF(JSTOP.LE.0) GO TO 998
  DO 12 J=1,JSTOP
  I = ITRIP + J
  IF(Z(40).LE.XM2(I,ISEND)) GO TO 13
  12 CONTINUE
  GO TO 998
  13 ITRIP = I
  IF(Z(12).GT.A2(KTRIP,ISEND)) GO TO 15
  LSTOP = KTRIP - 1
  DO 18 L=1,LSTOP
  K = KTRIP - L + 1
  IF(Z(12).GE.A2(K-1,ISEND)) GO TO 17
  18 CONTINUE
  GO TO 998
  15 LSTOP = LMAX - KTRIP
  IF(LSTOP.LE.0) GO TO 998
  DO 16 L=1,LSTOP
  K = KTRIP + L
  IF(Z(12).LE.A2(K,ISEND)) GO TO 17
  16 CONTINUE

```

```

998 W(3) = .1
   Z(22) = 0.
   WRITE(6,1001) NSEC
   CALL PAROUT
   W(32) = 1.1
   GO TO 999
17  KTRIP = K
   ARG1 = (Z(40) - XM2(I-1,ISEND))/(XM2(I,ISEND) - XM2(I-1,ISEND))
   D(K) = C2(I-1,K,ISEND) + ARG1*(C2(I,K,ISEND) - C2(I-1,K,ISEND))
   D(K-1) = C2(I-1,K-1,ISEND) + ARG1*(C2(I,K-1,ISEND) - C2(I-1,K-1,ISEND))
   ARG2 = (Z(12) - A2(K-1,ISEND))/(A2(K,ISEND) - A2(K-1,ISEND))
   Z(22) = D(K-1) + ARG2*(D(K) - D(K-1))
999 ITRP(M) = ITRIP
   KTRP(M) = KTRIP
   RETURN
   END

```

```

SUBROUTINE AERO3(M)
  DIMENSION D(12)
  DIMENSION ITRP(5), KTRP(5), ISND(5), JMX(5), LMX(5), XW3(5), XW4(5)
  COMMON NSEC, T, FR, TS(15), V(85), Q(73,15), W(50), Z(120), EC(3), CV(3),
  1 VQ(38,4), WQ(13,4), XM1(1,1), A1(1,1), C1(1,1,1), XM2(25,5), A2(12,5),
  2 C2(25,12,5), XM3(25,5), A3(12,5), C3(25,12,5), XNA0, XNA1(1,1),
  3 XNA2(2,5), XNA3(2,5), XNP(2), PST(25), TWD7(25,2), TWD8(25,2), SW(50),
  4 TWD9(25,2), PT(25,4), PDT(25,4), XWPT(4), SV(50), SQ(40,5), SE(40),
  5 PS(15)
  COMMON/AEROX/XW3,XW4
1001 FORMAT(88H0 **** FAILURE IN THE TABLE LOOK-UP OF SUBROUTINE AERO
13 DURING SIMULATION SECTION,I3,7H ****//)
  IF(XW4(M).GT..5) GO TO 13
  XW4(M) = 1.1
  ITRP(M) = 2
  KTRP(M) = 2
  ISND(M) = M
  JMX(M) = XNA2(1,M)
  LMX(M) = XNA2(2,M)
10 ITRIP = ITRP(M)
  KTRIP = KTRP(M)
  ISEND = ISND(M)
  JMAX = JMX(M)
  LMAX = LMX(M)
  IF(Z(40).GT.XM3(ITRIP,ISEND)) GO TO 11
  JSTOP = ITRIP - 1
  DO 14 J=1,JSTOP
    I = ITRIP - J + 1
    IF(Z(40).GE.XM3(I-1,ISEND)) GO TO 13
14 CONTINUE
  GO TO 998
11 JSTOP = JMAX - ITRIP
  IF(JSTOP.LE.0) GO TO 998
  DO 12 J=1,JSTOP
    I = ITRIP + J
    IF(Z(40).LE.XM3(I,ISEND)) GO TO 13
12 CONTINUE
  GO TO 998
13 ITRIP = I
  IF(Z(12).GT.A3PKTRIP,ISEND)) GO TO 15
  LSTOP = KTRIP - 1
  DO 18 L=1,LSTOP
    K = KTRIP - L + 1
    IF(Z(12).GE.A3PK-1,ISEND)) GO TO 17
18 CONTINUE
  GO TO 998
15 LSTOP = LMAX - KTRIP
  IF(LSTOP.LE.0) GO TO 998
  DO 16 L=1,LSTOP
    K = KTRIP + L
    IF(Z(12).LE.A3(K,ISEND)) GO TO 17
16 CONTINUE

```

```

998 W(4) = .1
   Z(23) = 0.
   WRITE(6,1081) NSEC
   CALL PAROUT
   W(32) = 1.1
   GO TO 999
17 KTRIP = K
   ARG1 = (Z(40) - XM3(I-1,ISEND))/(XM3(I,ISEND) - XM3(I-1,ISEND))
   D(K) = C3(I-1,K,ISEND) + ARG1*(C3(I,K,ISEND) - C3(I-1,K,ISEND))
   D(K-1) = C3(I-1,K-1,ISEND) + ARG1*(C3(I,K-1,ISEND) - C3(I-1,K-1,ISEND))
1ENDI)
   ARG2 = (Z(12) - A3(K-1,ISEND))/(A3(K,ISEND) - A3(K-1,ISEND))
   Z(23) = D(K-1) + ARG2*(D(K) - D(K-1))
999 ITRP(M) = ITRIP
   KTRP(M) = KTRIP
   RETURN
   END

```

Appendix IV-29 Subroutine PRPSN

```

SUBROUTINE PRPSN
COMMON NSEC,T,TR,TS(15),V(65),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
1001 FORMAT(83H0 **** FAILURE IN THE TABLE LOOK-UP OF SUBROUTINE PRPS
IN DURING SIMULATION SECTION,I3,7H ****//)
1002 FORMAT(64H0 **** WEIGHT OF THE VEHICLE IS ZERO DURING SIMULATION
1 SECTION,I3,7H ****//)
1003 FORMAT(62H0 **** INPUT SPECIFIC IMPULSE IS ZERO FOR SIMULATION S
1 SECTION,I3,7H ****//)
IF(W(5).GT..5) GO TO 17
C **** ISEND INITIALLY DEFINES THE TOTAL COMBINED PROPULSION OPTIONS
ISEND = IFIX(Q(31,NSEC))
W(5) = 1.1
17 IF(ISEND.GT.10) GO TO 24
C **** ISEND DEFINES THE SIMPO OPTION WHEN THERE ARE NO TABLES ****
GO TO (18,19,13,40,41,43,42,53,52,18), ISEND
C **** CONSTANT WDOT, VARIABLE THRUST AND SPECIFIC IMPULSE ****
18 IF(ABS(Q(2,NSEC)).LT..000001) GO TO 11
Z(27) = Q(1,NSEC) - (Q(1,NSEC) - Q(2,NSEC))*Z(36)/V(43)
GO TO 12
11 Z(27) = Q(1,NSEC)
12 Z(47) = Q(3,NSEC)
45 IF(ABS(7(47)).LT..000001) GO TO 10
Z(48) = Z(27)/Z(47)
GO TO 995
10 Z(48) = 0.
GO TO 995
C **** CONSTANT WDOT, VARIABLE THRUST AND SPECIFIC IMPULSE ****
19 IF(ABS(Q(3,NSEC)).LT..000001) GO TO 20
Z(47) = Q(1,NSEC)/Q(2,NSEC)
Z(48) = Q(2,NSEC) - (Q(2,NSEC) - Q(3,NSEC))*Z(36)/V(43)
GO TO 21
20 Z(48) = Q(2,NSEC)
7(47) = Q(1,NSEC)/Q(2,NSEC)
21 IF(ABS(Q(2,NSEC)).LT..000001) GO TO 22
GO TO 23
22 Z(47) = 0.
23 Z(27) = Z(47)*Z(48)
GO TO 995
C **** CONSTANT THRUST, VARIABLE WDOT AND SPECIFIC IMPULSE ****
13 IF(ABS(Q(3,NSEC)).LT..000001) GO TO 14
Z(48) = Q(2,NSEC) - (Q(2,NSEC) - Q(3,NSEC))*Z(36)/V(43)
GO TO 15
14 Z(48) = Q(2,NSEC)
15 Z(27) = Q(1,NSEC)
46 IF(ABS(7(48)).LT..000001) GO TO 16
Z(47) = Z(27)/Z(48)

```

C FIXED SOLID STRAPON SYNTHESIS OPTION *****

C
C
IF(NSEC.GT.1) GO TO 101
IF(SQ(20,1).LE.0.) GO TO 101
SQ(30,2) = Z(27)-SQ(30,1)
SQ(30,3) = Z(47) - SQ(30,4)
IF(SQ(30,3).GE.0.) GO TO 101
SQ(30,3) = 0.
SQ(30,2) = 0.
Z(27) = SQ(30,1)
Z(47) = SQ(30,4)
Z(48) = Z(27)/Z(47)

101 CONTINUE

C
C
GO TO 995

16 Z(47) = 0.

GO TO 995

C **** VACUUM THRUST, CONSTANT WDOT, AND EXIT AREA ****

40 Z(27) = Q(1,NSEC) - Z(36)*Q(3,NSEC)

GO TO 44

C **** SEA LEVEL THRUST, CONSTANT WDOT, AND EXIT AREA ****

41 Z(27) = Q(1,NSEC) + (V(43) - Z(36))*Q(3,NSEC)

44 Z(47) = Q(2,NSEC)

GO TO 45

C **** VAC THRUST - MAX (T-FA)/W, VACUUM AND SEA LEVEL SPECIFIC IMPULS

43 LSEND = 2

GO TO 47

C **** VAC THRUST - MAX T/W, VACUUM AND SEA LEVEL SPECIFIC IMPULSE **

53 LSEND = 4

47 IF(ABS(Q(2,NSEC)).LT..000001) GO TO 992

IF(ABS(Q(3,NSEC)).LT..000001) Q(3,NSEC) = Q(2,NSEC)

WDREF = Q(1,NSEC)/Q(2,NSEC)

GO TO 49

C **** S.L. THRUST - MAX (T-FA)/W, VACUUM AND SEA LEVEL SPECIFIC IMPUL

42 LSEND = 1

GO TO 48

C **** S.L. THRUST - MAX T/W, VACUUM AND SEA LEVEL SPECIFIC IMPULSE *

52 LSEND = 3

48 IF(ABS(Q(3,NSEC)).LT..000001) GO TO 992

IF(ABS(Q(2,NSEC)).LT..000001) Q(2,NSEC) = Q(3,NSEC)

WDREF = Q(1,NSEC)/Q(3,NSEC)

49 IF(ABS(Z(7)).LT..000001) GO TO 994

Z(48) = Q(2,NSEC) - (Q(2,NSEC) - Q(3,NSEC))*Z(36)/V(43)

ARG = WDREF*Z(48)

C FIXED SOLID STRAPON SYNTHESIS OPTION *****

C
C
IF(NSEC.GT.1) GO TO 100
IF(SQ(20,1).LE.0.) GO TO 100
TVS = SQ(20,1)*(SQ(20,2) + SQ(20,3)*T)
SQ(30,4) = TVS/SQ(20,4)

```

SQ(30,1) = TVS - SQ(20,1)*Z(36)*SQ(21,1)
SQ(30,2) = ARG
SQ(30,5) = ARG
ARG = ARG + SQ(30,1)
SQ(30,3) = WOREF
SQ(31,1) = WOREF
WOREF = WOREF + SQ(30,4)
Z(48) = ARG/WOREF
100 CONTINUE
C
C
GO TO (50,50,51,51), LSEND
50 IF((ARG+Z(25))/Z(7).LE.Q(66,NSEC)) GO TO 54
Z(27) = Z(7)*Q(66,NSEC) - Z(25)
GO TO 46
51 IF(ARG/Z(7).LE.Q(66,NSEC)) GO TO 54
Z(27) = Z(27)*Q(66,NSEC)
GO TO 46
54 Z(27) = ARG
Z(47) = WOREF
GO TO 995
992 W(5) = .1
WRITE(6,1003) NSEC
GO TO 993
994 W(5) = .1
WRITE(6,1002) NSEC
GO TO 993
C **** THRUST TABLE ****
24 IF(W(5).GT.1.5) GO TO 29
W(5) = 2.1
IARG = 10
C **** KSEND DEFINES WHICH TABLE IS TO BE USED ****
KSEND = MOD(ISEND,IARG)
C **** JSEND DEFINES THE SIMPO OPTION TO BE USED WITH THE TABLE ****
JSEND = (ISEND - KSEND)/10
JMAX = IFIX(XNP(KSEND))
CANT = COS(Q(50,NSEC)/V(45))
KSEC = IFIX(Q(47,NSEC))
ITRIP = 2
IF(KSEC.LT.1) KSEC = 1
29 ARG7 = T - TS(KSEC) - Q(48,NSEC)
IF(ARG7.LT.TWO7(1,KSEND)) GO TO 997
IF(ARG7.GT.TWO7(ITRIP,KSEND)) GO TO 25
JSTOP = ITRIP - 1
DO 27 J=1,JSTOP
I = ITRIP - J + 1
IF(ARG7.GE.TWO7(I-1,KSEND)) GO TO 28
27 CONTINUE
GO TO 998
25 JSTOP = JMAX - ITRIP
IF(JSTOP.LE.0) GO TO 998
DO 26 J=1,JSTOP

```

```

      I = ITRIP + J
      IF(ARGT.LE.TWD7(I,KSEND)) GO TO 28
26  CONTINUE
998  W(5) = .1
      WRITE(6,1001) NSEC
      CALL PAROUT
      W(32) = 1.1
      GO TO 997
993  CALL PAROUT
      W(32) = 1.1
      Z(27) = 0.
      Z(47) = 0.
      Z(48) = 0.
      GO TO 999
28  ITRIP = I
      ARG = (ARGT - TWD7(I-1,KSEND))/(TWD7(I,KSEND) - TWD7(I-1,KSEND))
      Z(27) = TWD8(I-1,KSEND) + ARG*(TWD8(I,KSEND) - TWD8(I-1,KSEND))
      IF(ABS(Q(51,NSEC)).GT..000001) GO TO 30
C  **** WDOT TABLE ****
      Z(47) = TWD9(I-1,KSEND) + ARG*(TWD9(I,KSEND) - TWD9(I-1,KSEND))
      GO TO 31
C  **** WDOT VIA EFFECTIVE SPECIFIC IMPULSE ****
30  Z(47) = Z(27)/Q(51,NSEC)
31  Z(27) = Q(49,NSEC)*CANT*(Z(27) - Z(36)*Q(61,NSEC))
      Z(47) = Q(49,NSEC)*Z(47)
      Z(48) = Z(27)/Z(47)
      GO TO 996
995  IF(ISEND.LT.11) GO TO 999
      Z(27) = Z(27) + Z27
      Z(47) = Z(47) + Z47
      Z(48) = Z(27)/Z(47)
      GO TO 999
997  Z(27) = 0.
      Z(47) = 0.
      Z(48) = 0.
996  IF(JSEND.GT.9) GO TO 991
      Z27 = Z(27)
      Z47 = Z(47)
      GO TO (18,19,13,40,41,43,42,53,52,18), JSEND
991  Z(47) = Z(47) + Q(3,NSEC)
999  RETURN
      END

```


Appendix IV-30 Subroutine TRAN1

GDC-DRB70-002

SUBROUTINE TRAN1(ARG1,ARG2,ARG3,SIN3,COS3,SIN4,COS4,ARK1,ARK2,AR
1)
ARK1 = ARG1*COS4 + ARG2*SIN4*COS3 - ARG3*SIN4*SIN3
ARK2 = -ARG1*SIN4 + ARG2*COS4*COS3 - ARG3*COS4*SIN3
ARK3 = ARG2*SIN3 + ARG3*COS3
RETURN
END

PRECEDING PAGE BLANK NOT FILMED

Appendix IV-30 Subroutine TRAN2

GDC-D8870-002

C
SUBROUTINE TRAN2(ARG1,ARG2,ARG3,SIN5,COS5,SIN6,COS6,ARK1,ARK2,AR
1)
ARK1 = -ARG1*SIN6 - ARG2*COS6*SIN5 + ARG3*COS6*COS5
ARK2 = ARG1*COS6 - ARG2*SIN6*SIN5 + ARG3*SIN6*COS5
ARK3 = ARG2*COS5 + ARG3*SIN5
RETURN
END

PRECEDING PAGE BLANK NOT FILMED

GDC-DBB70-002

Appendix IV-30 Subroutine TRAN3

SUBROUTINE TRAN3(ARG1,ARG2,ARG3,SINA,COSA,SINB,COSB,SINC,COSC,AR
1,ARK2,ARK3)
ARK1 = ARG1*(COSA*COSB*COSC - SINA*SINC) + ARG2*(SINA*COSB*COSC
1 COSA*SINC) - ARG3*SINB*COSC
ARK2 = -ARG1*(COSA*COSB*SINC + SINA*COSC) - ARG2*(SINA*COSB*SINC
1 COSA*COSC) + ARG3*SINB*SINC
ARK3 = ARG1*COSA*SINB + ARG2*SINA*SINB + ARG3*COSB
RETURN
END

PRECEDING PAGE BLANK NOT FILMED

GOC-DBR70-002

Appendix IV-31 Subroutine QUAD

```
      SUBROUTINE QUAD(X,Y,I,M)
C  **** X IS THE ENTRY ARGUMENT TO BE OPERATED ON ****
C  **** Y IS THE EXIT ARGUMENT WHICH LIES WITHIN THE REQUIRED BOUNDS
C  **** I DEFINES THE PRINCIPAL CYCLE BOUNDS ****
C      I = 1.  DEFINES CYCLE BOUNDS OF 0 AND +360 DEGREES
C      I = 2.  DEFINES CYCLE BOUNDS OF -180 AND +180 DEGREES
C      I = 3.  DEFINES CYCLE BOUNDS OF 0 AND +2*PI RADIANS
C      I = 4.  DEFINES CYCLE BOUNDS OF -PI AND +PI RADIANS
C  **** M DEFINES THE MAXIMUM NUMBER OF CYCLE TEST SETS ****
1001 FORMAT(71H0 **** FAILURE TO PLACE ANGULAR ARGUMENT IN THE PRINC.
      1PAL CYCLE ****//5X,10H ENTRY ARG,10X,9H EXIT ARG,11X,10H LOW BOUN
      2D,10X,11H HIGH BOUND,9X,13H CYCLE LENGTH,7X,2H N//5E20.8,4X,I3//)
      IF(M.LT.0) M = 10
      GO TO (10,10,20,20), I
10  C = 360.
      GO TO (11,12), I
11  A = 0.
      B = 360.
      GO TO 50
12  A = -180.
      B = 180.
      GO TO 50
20  PI = 3.14159265
      C = 2.*PI
      J = I - 2
      GO TO (21,22), I
21  A = 0.
      B = C
      GO TO 50
22  A = -PI
      B = PI
50  N = 1
      Y = X
51  IF(Y.LT.A) GO TO 52
      IF(Y.LT.B) GO TO 999
      Y = Y - C
      GO TO 53
52  Y = Y + C
53  IF(N.GT.M) GO TO 998
      N = N + 1
      GO TO 51
998 WRITE(6,1001) X, Y, A, B, C, N
999 RETURN
      END
```

PRECEDING PAGE BLANK NOT FILMED

G7C-08870-002

Appendix IV-32 Subroutine QARG

```
      SUBROUTINE QARG(Y,Z,I,J)
C  **** Y IS THE ENTRY ARGUMENT WHICH IS THE ARGUMENT OF THE FUNCTION
C  **** Z IS THE EXIT ARGUMENT WHICH IS EVALUATED BY THE FUNCTION **
C  **** I DEFINES THE FUNCTION WHICH IS USED IN EVALUATING Z ****
C      I = 1. DENOTES THE ARCSINE FUNCTION
C      I = 2. DENOTES THE ARCCOSINE FUNCTION
C      I = 3. DENOTES THE ARCTANGENT FUNCTION
C      I = 4. DENOTES THE SQUARE ROOT FUNCTION
C  **** J IS THE ANGULAR CONVERSION FLAG
C      J = 0., Z IS EXPRESSED IN DEGREES FOR I = 1, 2, OR 3
C      J = 0., Z IS SET TO ZERO WHEN Y IS NEGATIVE FOR I = 4
C      J = 1., Z IS EXPRESSED IN RADIANS FOR I = 1, 2, OR 3
C      J = 1., Z IS SET EQUAL TO THE SQUARE ROOT OF -Y WHEN Y IS
C      NEGATIVE FOR I = 4
      PI = 3.14159265
      TEST2 = .9999999
      IF(I.GE.4) GO TO 40
      IF(J.GT.0) GO TO 50
C  **** DETERMINATION OF ANGULAR CONVERSION FACTOR ****
      RR = 57.2957795
      GO TO 51
50 RR = 1.
51 GO TO (10,20,30,40), I
C  **** ARCSINE FUNCTION ****
10 IF(ABS(Y).LT.TEST2) GO TO 11
   ARG = .5*PI*RR
   IF(Y.LT.0.) GO TO 12
   Z = ARG
   GO TO 999
12 Z = -ARG
   GO TO 999
11 Z = RR*ASIN(Y)
   GO TO 999
C  **** ARCCOSINE FUNCTION ****
20 IF(ABS(Y).LT.TEST2) GO TO 21
   IF(Y.LT.0.) GO TO 22
   Z = 0.
   GO TO 999
22 Z = PI*RR
   GO TO 999
21 Z = RR*ACOS(Y)
   GO TO 999
C  **** ARCTANGENT FUNCTION ****
30 Z = RR*ATAN(Y)
   GO TO 999
C  **** SQUARE ROOT FUNCTION ****
40 IF(Y.LT.0.) GO TO 41
   Z = SQRT(Y)
   GO TO 999
41 IF(J.LE.0) GO TO 42
   Z = SQRT(-Y)
   GO TO 999
```

GDC-08870-002

42 Z = 0.
999 RETURN
END

Appendix IV-33 Program WTVOL

```

SUBROUTINE WTVOL
OVERLAY (TTSS, 3, 0)
PROGRAM WTVOL
REAL ISP, K, LF, MR, NCREW, LBODY, NPASS
REAL NENG5
DIMENSION SK0(30), SC0(300), T04(6), T020(10), T027(6), T034(6),
1 T048(10), T049(10), T050(10), T057(6)
DIMENSION SKR(30), SCR(300), TR4(6), TR20(10), TR27(6), TR34(6),
1 TR48(10), TR49(10), TR50(10), TR57(6)
COMMON/ORPINX/
1 T01, T02, T03, T04, T05, T06, T07, T08, T09, T010, T011, T012, T013, T014,
2 T015, T016, T017, T018, T019, T020, T021, T022, T023, T024, T025, T026, T027,
3 T028, T029, T030, T031, T032, T033, T034, T035, T036, T037, T038, T039, T040,
4 T041, T042, T043, T044, T045, T046, T047, T048, T049, T050, T051, T052, T053,
5 T054, T055, T056, T057, SK0, SC0
COMMON/ORBINY/
1 TB1, TB2, TB3, TB4, TB5, TB6, TB7, TB8, TB9, TB10, TB11, TB12, TB13, TB14,
2 TB15, TB16, TB17, TB18, TB19, TB20, TB21, TB22, TB23, TB24, TB25, TB26, TB27,
3 TB28, TB29, TB30, TB31, TB32, TB33, TB34, TB35, TB36, TB37, TB38, TB39, TB40,
4 TB41, TB42, TB43, TB44, TB45, TB46, TB47, TB48, TB49, TB50, TB51, TB52, TB53,
5 TB54, TB55, TB56, TB57, SKR, SCR
COMMON/NSEC, T, TR, TS(15), V(85), Q(73, 15), W(50), Z(120), EC(3), CV(3),
1 VQ(38, 4), WQ(13, 4), XM1(1, 1), A1(1, 1), C1(1, 1, 1), XM2(25, 5), A2(12, 5),
2 C2(25, 12, 5), XM3(25, 5), A3(12, 5), C3(25, 12, 5), XNA0, XNA1(1, 1),
3 XNA2(2, 5), XNA3(2, 5), XNP(2), PST(25), TWD7(25, 2), TWD8(25, 2), SW(50),
4 TWD9(25, 2), PT(25, 4), POT(25, 4), XNPT(4), SV(50), SO(40, 5), SE(40),
5 PS(15)
COMMON / JUMPY / JUMP, WBIG, WROO
COMMON/CINPUT/
1ANENG5 ,ANTANK ,ASRATO ,ASWEEP ,C(300) ,CBRODY ,CFUFL(6) ,
2CHBODY ,CLBODY ,CSBODY ,CSFAIR ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN ,CSVERT ,CSWING ,CTHRST ,CTHST2 ,DEF(5) ,FXWOVS ,
4ISP(6) ,ITPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5NENG5 ,NLISTO ,NPASS ,NWL ,PCHAM ,QMX ,RHOFU ,
6RHOFU2 ,RHOX ,RHOX2 ,SBODY ,TOL ,TOVERC ,TPRATO ,
7TYTATL ,VRODY ,WGR0SS
COMMON/VOLCAL/RRODY,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HRODY ,LBODY ,RT00 ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPLAN ,STPS(1) ,SVERT ,SWING ,SXPOS ,TDEL ,TROOT ,
4TTOT ,TYOT2 ,TTOTAL ,VRODYA ,VRODY1 ,VRODY2 ,VCARGO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGRAY ,VOTHER ,VOXTK ,
6VOXTK2 ,VPROP ,VSTRUC
COMMON/WTCALC/ ABFSYS ,WABFTK ,WABFU ,WABPR ,WACRES ,
1WACS ,WACSF0 ,WACSTK ,WAFRO ,WAUXT ,WBASIC ,WBODY ,
2WRPUMP ,WCARGO ,WCOMM ,WCONT ,WCOVER ,WDECAY ,WDIST1 ,
3WDIST2 ,WDOCK ,WDPLOY ,WDRANS ,WDRY ,WELCAD ,WEMPTY ,
4WENGHT ,WENG5 ,WENG52 ,WFAIR ,WFCONT ,WFDCAY ,WFROST ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFUOX ,WFUREL ,
6WFUSYS ,WFUTK ,WFUTK2 ,WFUTOT ,WFUTPP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLOSS ,WLRD ,WNACEL ,WODCAY ,
9WOIL ,WOILRS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,WOXID ,

```

```

1WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPERS ,WPOWCD ,WPOWER ,WPOWFO ,
3WPOWRS ,WFOWTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
4WPRESID ,WRESRV ,WSEAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF ,WTABC ,WTHRST ,WTO ,WTPS ,WVERT ,WWAIT(10),
6WNET ,WWING ,WZROFU

```

```
COMMON/TAMP/WGROSS,OTTOT,WFUOXO,TBTO,BTTOT
```

```
COMMON/OS/OSPLAN
```

```
BTWTOL=SW(18)*.5
```

```
OMX=SE(9)
```

```
WOREQ= SQ(10,1)
```

```
WPOREQ = SQ(13,1)
```

```
GWPEQ= SQ(16,1)
```

```
IF(SW(3).GE.1.) GO TO 40
```

```
SO(15,2)=1.
```

```
40 CONTINUE
```

```
SW(3)=SW(3)+1.
```

```
BID=.00005
```

```
ROB=.00025
```

```
ILAST=0
```

```
W(3F)=.1
```

```
GUS = 0.
```

```
ACEL=0.
```

```
ACNT=0.
```

```
101 CONTINUE
```

```
ACNT=ACNT+1.
```

```
CALL ORBCAL
```

```
FIXED THRUST-TO-WEIGHT
```

```
ISP(3) = T027(3)
```

```
JUMP = 0
```

```
CALL WTSCH
```

```
CALL SOLVE
```

```
VBODY1 = VBODY
```

```
WGROSS=WGROSS
```

```
WFUOXO=WFUOX
```

```
OTTOT=TYOT
```

```
OSPLAN = SPLAN
```

```
BURN = T
```

```
GO TO 291
```

```
195 CONTINUE
```

```
CALL BOCCAL
```

```
196 IF( SE(6) .EQ. 0.) GO TO 200
```

```
FIXED THRUST-TO-WEIGHT
```

```
197 SCR(129) = 0.
```

```
C(129) = 0.
```

```
CTHRST = CTHRST*SE(1)/SE(3)
```

```
TR18 = CTHRST
```

```
GO TO 210
```

```
FIXED THRUST
```

```
200 CTHRST = 0.
```

```
TB18 = 0.
```

```
TATO = SW(6) * TR36/T036
```



```

      SCB(129) = OTTOT*TBTC/TB36
      C(129) = SCB(129)
210  ISP(3) = TB27(3)
      JUMP = 1
      BURN = SW(20)
      CALL WTSCH
      CALL SOLVE
      BTOT = TTOT
      VBODY2 = VBODY
269  TB51 = VBODY2
      T051 = VBODY1
      TB56 = WGRASS
      T056 = WGRASSO
291  CONTINUE
      IF(ILAST.NE.1 .AND. SW(16) .EQ. 0.) GO TO 292
      CALL PRINTW
      CALL PRINTV
      CALL PRWTSM
292  CONTINUE
      IF(JUMP.EQ.1) GO TO 300
      CALL ORBSUM
      CALL ORBSTO
      GO TO 195
300  CONTINUE
      C(3) = SCB(3)
      C(5) = SCB(5)
      C(7) = SCB(7)
      C(146) = SCB(146)
      C(31) = SCB(31)
      C(56) = SCB(56)
      C(129) = SCB(129)
      C(211) = SCB(211)
      C(67) = SCB(67)
      IF(SE(6).EQ.0.) GO TO 400
      CTHRST = CTHRST*SE(3)/SE(1)
400  CALL ROOSUM
      CALL ROOSTO
      CALL TAMPER(1)
      TWO = OTTOT/T056
      WRITE (6,1080) T034(3),T056,CTTOT, TWO,TB34(3),TB56,Q(1,1),SV(13)
      IF(ILAST.EQ.1) GO TO 950
      IF(
          SE(6).EQ.1. ) GO TO 500
      IF(ACNT.GE.SW(19)) GO TO 500
C   *** THIS LOOP ADJUSTS CTHRST (ORBITER T/W) TO OBTAIN
C   SPECIFIED LIFTOFF T/W ***
      IF(SW(2).EQ.1.) BTWTOL=SW(18)
      IF( ABS(SV(13)-SW(17)) .LE. BTWTOL ) GO TO 500
      T018 = T018/SV(13) * SW(17)
      GO TO 193
500  CONTINUE
      ACNT=0.
      ACFL=ACFL+1.

```

```

      IF(GWREQ.LE.0.) GO TO 800
C   *** THIS LOOP ADJUSTS PAYLOAD WEIGHT TO OBTAIN
C   SPECIFIED GROSS LIFTOFF WEIGHT ***
      DIFF=GWREQ-T856
      TOM=DIFF/GWREQ
      IF(ABS(TOM) .LT.818 ) GO TO 504
      IF(ACEL.GT.1.) GO TO 502
      SLOPE=-21. 1
      IF(SW(3) .GT.1. .OR. GUS.GT.0.) SLOPE=SW(8)
      GO TO 505
502 SLOPE = (DIFF - SDIFF)/(SCO(103) - SAPAY)
505 CONTINUE
      WRITE(6,1001) SCO(103),DIFF,SLOPE
      SW(8) = SLOPE
      SDIFF = DIFF
      SAPAY = SCO(103)
      SCO(103) = SCO(103) - DIFF/SLOPE
      IF(ACEL.GT.6.) GO TO 503
      GO TO 101
503 WRITE(6,1002)
      GO TO 600
504 WRITE(6,1001) SCO(103),DIFF,SW(8)
      TOM = DIFF/GWREQ
      WRITE(6,1003) TOM
      GO TO 600
800 CONTINUE
      IF(SE(6).EQ.1..OR.SW(19).GT.0.) GO TO 600
C   *****THIS LOOP ADJUSTS ORBITER MR(3) TO COMPENSATE FOR CHANGE IN T/W
C   AT LIFTOFF. *****
      SQ(15,1)=SQ(15,2)
      SQ(15,2)=SV(13)-1.
C   SV(13) IS LIFTOFF THRUST-TO-WEIGHT
      IF(ACEL.GE.2.) GO TO 600
      IF(ACEL.EQ.1..AND.SW(3).LE.1..AND.GUS.EQ.0.) GO TO 600
      DIFF= SQ(15,2)-SQ(15,1)
      IF(ABS(DIFF).LT.818) GO TO 600
      SLOPE=500. /SE(19)/ 32.174
      X=T034(3)*SLOPE/SQ(15,2) - .2
C   THE CONSTANT -.2 REFLECTS WEIGHT REDUCTION DUE TO QMAX DECREASE
C   ***** X IS AN ESTIMATE OF THE FEEDBACK OF T/W VARIATIONS
C   THROUGH CHANGES IN MUO AND QMAX. *****
      IF(X.GE..88) GO TO 803
      IF(X.GT..44) X=.44
      DTWT=-DIFF/SQ(15,1)/SQ(15,2)/(1.-X)
      DTWT = DIFF/ (1.-X)
      RATIO = 1. + DTWT / SQ(15,2)
      OLD=T034(3)
      T034(3) = OLD * RATIO*(-2.*SLOPE)
      DMU= T034(3)-OLD
      WRITE(6,1004) DMU,X,DTWT
C   ***** ADJUST THE QMAX ESTIMATE *****
      SE(9)=SF(9)+SE(7)*DTWT

```

```

      QMX=SE(9)
      GO TO 101
    603 WRITE(6,1008)
      W(36)= 1.1
    1008 FORMAT(5X,50H ***** RUN ABORTED IN WTVOL, FIXED THRUST LOW )
      GO TO 900
C
    600 CONTINUE
      IF(ACEL.GT.6.) GO TO 900
      ACFL = 0.
      IF(WOREQ.LE.0.) GO TO 700
C   *** THIS LOOP ADJUSTS BOOSTER MASS RATIO TO OBTAIN
C   SPECIFIED ORBITER GROSS WEIGHT. ***
      DW=WOREQ-T056
      ERR=DW/WOREQ
      IF(ABS(ERR) .LT. 808) GO TO 604
      IF(GUS.GT.0.) GO TO 602
      S= .54 *T056 * T034(3)/TB34(3)
      IF(SW(3).GT.1.) S = SQ(11,1)
      GUS=0.
      GO TO 605
    602 S = (D W-SD W)/(TB34(3)-SMR)
    605 WRITE(6,1005) D W,S
      SQ(11,1) = S
      SD W = D W
      SMR = TB34(3)
      TB34(3) = TB34(3) - D W/S
      RPAR=SQ(19,3)
      IF(RPAR.GT.100.) SCB(214)=SCB(214)+ SE(8)*(TB34(3)-SMR) /RPAR
      T034(3)= T034(3)*SMR / TB34(3)
      IF(GUS.GT.6.) GO TO 603
      GUS = GUS + 1.
      GO TO 101
    603 WRITE(6,1002)
    604 WRITE(6,1005) DW ,SQ(11,1)
      GO TO 900
    700 IF(WPOREQ.LE.0.) GO TO 900
C   *** THIS LOOP ADJUSTS BOOSTER MASS RATIO TO OBTAIN
C   SPECIFIED ORBITER PROPELLANT WEIGHT
C   INCLUDES FPR, EXCLUDES ON-ORBIT DELTA V ***
      DW=WPOREQ-WFUOXO
      ERR=DW/WPOREQ
      IF(ABS(ERR) .LE. 808) GO TO 704
      IF(GUS.GT.0.) GO TO 702
      S= .690*WFUOXO*T034(3) / TB34(3)
      IF(SW(3).GT.1.) S = SQ(11,1)
      GUS=0.
      GO TO 705
    702 S = (DW -SDW )/(TB34(3)-SMR)
    705 WRITE(6,1006) DW ,S
      SQ(11,1) = S
      SD W = D W

```

```

SMR = TB34(3)
TB34(3) = TB34(3) - DW / S
RPAR = SQ(19,3)
IF(RPAR.GT.100.) SCB(214) = SCB(214) + SE(8)*(TB34(3)-SMR) / RPAR
TO34(3) = TO34(3)*SMR / TB34(3)
IF(GUS.GT.6.) GO TO 703
GUS = GUS + 1.
GO TO 101
703 WRITE(6,1002)
704 WRITE(6,1006) DW,SQ(11,1)
900 CONTINUE
IF(SW(2).EQ.1.) ILAST=1
IF(ILAST.EQ.1) GO TO 101
950 CONTINUE
CALL TAMPER(2)
RETURN
1000 FORMAT (15H00RBITER MU = F9.5,13H WEIGHT = F9.0,13H THRUST
1= F9.0,10H T/W = F9.6/
2 15H ROOSTER MU = F8.5,13H WEIGHT = F9.0,13H THRUST
3= F9.0,10H T/W = F9.6)
1001 FORMAT(5X,11HPAYLOAD = F10.2,5X,14HLO WT DIFF = F12.2,5X,10HSLOP
1E = F12.4///)
1002 FORMAT(5X,36H***** ITERATION COUNT EXCEEDED *****/)
1003 FORMAT(5X,15HCONVERG ERR = F10.6///)
1004 FORMAT( 8X,5HDMU = F8.5, 5H X = F8.4, 8H DTWT = F9.6 / )
1005 FORMAT(5X,16HOPR WT DIFF = F10.1,5X,9HSLOPE = F10.1//)
1006 FORMAT(5X,25HOPR PROPELLANT WT DIFF = F10.1,5X,8HSLOPE = F10.1//)
END

```

```

SUBROUTINE WTSCH
C *** WTSCH ***
C
C WTSCH SUBROUTINE REPROGRAMED BY WALTER D. HONEYCUTT
C AND B. H. OMAN ON 9/17/70 INPUT DECKS PRIOR TO THIS DATA ARE NO
C LONGER VALID
C
C REAL ISP,K,LF,MR,NCREW,LBODY,NPASS
C REAL NENGs
C
C COMMON/CINPUT/
1ANENGs ,ANTANK ,ASRATO ,ASWEEP ,C(300) ,CBBODY ,CFUEL(6) ,
2CHBODY ,CLBODY ,CSBODY ,CSFAIR ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN ,CSVERT ,CSWING ,CTHRST ,CTHST2 ,DEF(5) ,FXNOVS ,
4ISP(6) ,ITPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5NENGs ,NLIST0 ,NPASS ,NML ,PCHAM ,Q ,RHOFU ,
6RHOFU2 ,RHOX ,RHOX2 ,SBODY ,TOL ,TOVERC ,TPRATO ,
7TTYTAIL ,VBODY ,WGROSS
C COMMON/VOLCAL/BBODY,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HBODY ,LBODY ,RTOD ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPLAN ,STPS(1) ,SVERT ,SWING ,SXPOS ,TOEL ,TROOT ,
4TTOT ,TTOT2 ,TTOTAL ,VBDYA ,VBDY1 ,VBDY2 ,VCARGO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGBAY ,VOTHER ,VOXTK ,
6VOXTK2 ,VPROP ,VSTRUC
C COMMON/WTCALC/ ABFSYS ,WABFTK ,WABFU ,WABPR ,WACRES ,
1WACS ,WACSF0 ,WACSTK ,WAERO ,WAUXT ,WBASIC ,WBODY ,
2WBUMP ,WCARGO ,WCOMM ,WCONT ,WCOVER ,WDECAY ,WDIST1 ,
3WDIST2 ,WDOCK ,WDPLY ,WDRANS ,WDRY ,WELCAD ,WEMPTY ,
4WENGMT ,WENGs ,WENGs2 ,WFAIR ,WFCONT ,WFDCAY ,WFROST ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFUOX ,WFURES ,
6WFUSYS ,WFUTK ,WFUTK2 ,WFUTOT ,WFUTRP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLOSS ,WLRD ,WNACEL ,WODCAY ,
9WOIL ,WOILRS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,WOXID ,
1WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPERS ,WPOWCD ,WPOWER ,WPOWFO ,
3WPOWRS ,WPOWTR ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
4WRESID ,WRESRV ,WSEAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF ,WABC ,WTHRST ,WTO ,WTPS ,WVERT ,WWAIT(10) ,
6WWET ,WWING ,WZROFU
C
C COMMON NSEC,T,TR,XXX(3257),SW(20),YYY(422)
C COMMON/JUMPY/JUMP,WBIG,WBOO
C
C NAMELIST/MASS1/JUMP,
1ABFSYS ,BBODY ,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HBODY ,LBODY ,RTOD ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPLAN ,STPS ,SVERT ,SWING ,SXPOS ,TOEL ,TROOT ,
4TTOT ,TTOT2 ,TTOTAL ,VBDYA ,VBDY1 ,VBDY2 ,VCARGO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGBAY ,VOTHER ,VOXTK ,
6VOXTK2 ,VPROP ,VSTRUC ,WABFTK ,WABFU ,WABPR ,WACRES ,
7WACS ,WACSF0 ,WACSTK ,WAERO ,WAUXT ,WBASIC ,WBODY ,

```

```

8WBPUMP ,WCARGO ,WCOMH ,WCOMT ,WCOVER ,WDECAY ,WDIST1 ,
9WDIST2 ,WDOCK ,WDPLOY ,WDRANS ,WDRY ,WELCAD ,WEMPTY ,
*WENGHT ,WENG5 ,WENG52 ,WFAIR ,WFCONT ,WFOCAY ,WFROST
  NAMELIST/MASS2/
*WFU2 ,WFUEL ,WFUL ,WFULOS ,WFUNCT ,WFOUX ,WFURES ,
*WFUSYS ,WFUTK ,WFUTK2 ,WFUTOT ,WFUTRP ,WGASPR ,WGNV ,
*WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
*WJET ,WLANCH ,WLG ,WLOSS ,WLRD ,WNACEL ,WOCAY ,
*WOIL ,WOILRS ,WORSUL ,WOVERS ,WOX ,WOX2 ,WOXIO ,
*WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
*WP ,WPASS ,WPAYL ,WPERS ,WPOWCD ,WPOWER ,WPOWFO ,
*WPOWRS ,WPOWTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
*WRESID ,WRESRV ,WSEAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
*WSURF ,WTABC ,WTHRST ,WTO ,WTPS ,WVERT ,WWAIT ,
*WWET ,WWING ,WZROFU

```

```

***** END COMMON *****

```

INITIALIZE VARIABLES

```

WTO=WGROSS
C13=1./3.
C23=2./3.
RTOD=57.3
WPREIG=C(134)
WWAIT(1)=WGROSS-WPREIG
WFROST = C(78)

```

CALCULATE PROPELLANT WEIGHTS

COMPUTE FUEL PERCENTAGE

```
DO 10 I=1,6
```

```
IF (CFUEL(I).GE.1.0) CFUEL(I)=1./(CFUEL(I)+1.)
```

COMPUTE MASS RATIO

```
IF(ISP(I).NE.0..AND.MR(I).GT.15.)MR(I)=EXP(MR(I)/32.174/ISP(I))
IF(MR(I).EQ.0.0) MR(I)=1.

```

```
10 CONTINUE
```

COMPUTE JETTSION WEIGHT

```

WJET(1)=0.0
WJET(2)=0.0
IF(JUMP.EQ.0)
1WJET(3)=WFROST

```

```

WJET(4)=WFUTRP+WOTRTP+WSRTRP+WDECAY+WFURES+WOXRES
WJET(5)=WACSFO+WFULOS+WOXLOS+WPOWFO+WGASPR+WACRES+WPOWRS
WJET(6)=0.0
WFUL=0.0
WOXID=0.0

```

```

C
C
C   COMPUTE PROPELLANT WEIGHT
C
C   DO 40 I=1,4
C
C     WFOOX = WWAIT(I)*(MR(I)-1.)*(I)
C
C     COMPUTE FUEL WEIGHTS
C
C     WFUEL(I)=WFOOX*CFUEL(I)
C
C     COMPUTE LOX WEIGHTS
C
C     WOX(I)=WFOOX-WFUEL(I)
C
C     THRUST BUILD UP PROPELLANTS INPUT CONSTANT
C
C     GO TO (25,30,30,40),I
25  WFUEL(I) = WFUEL(I) + C(132)
    WOX(I)=WOX(I)+B(133)
    WFOOX=WFUEL(I)+WOX(I)
C
C   30 CONTINUE
C
C     COMPUTE VEHICLE WEIGHTS
C
C     WWAIT(I+1)=WWAIT(I)-WFOOX-WJET(I)
C
C     SUM FUEL WEIGHTS
C
C     WFUL=WFUL+WFUEL(I)
C
C     SUM LOX WEIGHTS
C
C     WOXID=WOXID+WOX(I)
C
C   40 CONTINUE
C
C     WWAIT(5)=WWAIT(4)-WJET(4)
C
C     COMPUTE TOTAL THRUST
C
C     TTOT=CTHRST*WWAIT(2)+C(129)*NENG5
C
C     TOTAL THRUST/1000000
C

```

TTOTAL=TTOT/1000000.

COMPUTE SECONDARY THRUST

TTOT2 = WHAIT(5)*CTHST2 + C(158)

COMPUTE ACS PROPELLANTS

WACSF0=C(173)*WTO+C(174)*WHAIT(4)+C(175)

COMPUTE POWER SOURCE PROPELLANTS

WPOWF0=C(38)*WHAIT(6)+C(127)

COMPUTE SERVICE ITEM LOSSES

WOIL=C(130)*TTOT+C(131)

COMPUTE MAIN FUEL RESERVES

WFURES=C(115)*WFUEL(3)+C(116) + WFUEL(4)

COMPUTE MAIN LOX RESERVES

WOXRES=C(117)*WOX (3)+C(118) +WOX(4)

COMPUTE ACS PROPELLANT RESERVE

WACRES=C(172)*WACSF0+C(173)

COMPUTE POWER SOURCE PROPELLANT RESERVE

WPOWRS=C(119)*WPOWF0+C(120)

COMPUTE SERVICE ITEM RESERVES

WOILRS=C(121)*WOIL+C(122)

SUM PROPELLANT RESERVES

WRESRV=WFURES+WOXRES+WACRES +WPOWRS+WOILRS

COMPUTE SECONDARY FUEL WEIGHT

WFU2(1) = WHAIT(5)*(MR(5)-1.)/MR(5)*CFUEL(5)
WFUEL(5)=WFU2(1)

COMPUTE SECONDARY LOX WEIGHT

WOX2(1) = WHAIT(5)*(MR(5)-1.)/MR(5)*(1.-CFUEL(5))
WOX(5)=WOX2(1)

45 CONTINUE


```

C
C   COMPUTE TOTAL FUEL WEIGHT LESS TRAPPED AND LOSSES
C
C   WFUTOT = (WFUL + WFURES + C(110)) / (1. - C(109))
C
C   COMPUTE TOTAL LOX WEIGHT LESS TRAPPED AND LOSSES
C
C   WOXTOT = (WOXID + WOXRES + C(112)) / (1. - C(111))
C
C   SUM PROPELLANT WEIGHT LESS LOSSES AND TRAPPED
C
C   WP = WFUTOT + WOXTOT
C
C   COMPUTE FUEL LOSSES
C
C   WFULOS = C(123) * WFUTOT + C(229) * WP + C(124)
C
C   COMPUTE LOX LOSSES
C
C   WOXLOS = C(125) * WOXTOT + C(230) * WP + C(126)
C
C   SUM TOTAL FUEL WEIGHT AND LOSSES
C
C   WFUTOT = WFUTOT + WFULOS
C
C   SUM TOTAL LOX WEIGHT AND LOSSES
C
C   WOXTOT = WOXTOT + WOXLOS
C
C   SUM PROPELLANT WEIGHT LESS TRAPPED
C
C   WP = WFUTOT + WOXTOT
C
C   COMPUTE TRAPPED FUEL WEIGHT
C
C   WFUTRP = C(109) * WFUTOT + C(225) * WP + C(226) * TTOT + C(110)
C
C   COMPUTE TRAPPED LOX WEIGHT
C
C   WOXTRP = C(111) * WOXTOT + C(227) * WP + C(228) * TTOT + C(112)
C
C   COMPUTE THRUST DECAY PROPELLANTS
C
C   WDECAY = C(166) * TTOT + C(167)
C   WFDCAV = WDECAY * GFUEL(3)
C   WODCAY = WDECAY - WFDCAV
C
C   COMPUTE TOTAL FUEL WEIGHT
C
C   WFUTOT = WFUTOT + WFUTRP + WFDCAV
C
C   COMPUTE TOTAL LOX WEIGHT

```

```

C      WOXTOT=WOXTOT+WOXTRP+WODSAY
C
C      COMPUTE TANK PRESSURIZATION AND PURGE GASES
C
C      COMPUTE TRAPPED SERVICE ITEMS
C
C      WSRTRP=C(113)*WNAIT(1)+CF114)
C
C      SUM ENTRY WEIGHT
C
C      WNAIT(6)=WNAIT(5)-WFUEL(5)-WOX(5)-WJET(5)
C
C      COMPUTE AIR BREATHING FUEL
C
C      WABFU=C(215)+C(214)/(1.+C(214))*WNAIT(6)
C
C      COMPUTE AIR BREATHING FUEL TANK VOLUME
C
C      IF(RHOFU.EQ.0.0.OR.WNAIT(6).EQ.0.0) GO TO 9999
C      IF(C(212).NE.0.0.OR.C(213).NE.0.0) K(28)=WABFU/RHOFU
9999  CONTINUE
C
C      SUM LANDING WEIGHT
C
C      WNAIT(7)=WNAIT(6)-WABFU
C
C      SUM SUBTOTAL
C
C      WTO=WNAIT(1)-WFUEL(1)-WOX(1)
C      WFOOX=WFUL+WOXID+WFU2(1)+WOX2(1)
C      WRESID = WFUTRP+WOXTRP+WSRTRP
C      WLOSS=WFULOS+WOXL0S+WACSP0+WPOWFO+W0IL+WABFU+WFR0ST
C
C      *****
C      GEOMETRY SECTION
C      *****
C
C      COMPUTE BODY VOLUME TO THE 1/3 POWER
C
C      VBODY1=VBOUY**C13
C
C      COMPUTE BODY VOLUME TO THE 2/3 POWER
C
C      VBODY2=VBOUY1**2
C
C      COMPUTE LOX TANK VOLUME
C
C      VOXTK=0.0
C      IF (RHOX.NE.0.) VOXTK=(WOXTOT/RHOX)*(K(2)+1.)+K(29)

```

```

C
C
C      COMPUTE FUEL TANK VOLUME
C
C      VFUTK=0.0
C      IF (RHOFU.NE.0.) VFUTK=(WFUTOT/RHOFU)*(K(1)+1.)+K(25)+K(21)
C
C
C      COMPUTE FUEL TANK SURFACE AREA
C
C      SFUTK=0.0
C      IF(VFUTK.GT.0.0)
C      1SFUTK=CSFUTK*VFUTK**C23
C
C
C      COMPUTE LOX TANK SURFACE AREA
C
C      SOXTK=0.0
C      IF(VOXTK.GT.0.0)
C      1SOXTK=CSOXTK*VOXTK**C23
C
C
C      COMPUTE PROPELLANT TANKS INSULATION VOLUME
C
C      VINSTK=K(3)*SFUTK+K(4)+K(25)*SOXTK
C
C
C      COMPUTE SECONDARY FUEL TANK VOLUME
C
C      VFUTK2=K(7)
C      IF(RHOFU2.NE.0.0)
C      1VFUTK2=WFU2(1)/RHOFU2
C
C
C      COMPUTE SECONDARY OXIDIZER TANK VOLUME
C
C      VOXTK2=K(8)
C      IF(RHOX2.NE.0.0)
C      1VOXTK2=WOX2(1)/RHOX2
C
C
C      COMPUTE PROPULSION SYSTEM VOLUME
C
C      VPROP=K(16)*TTOT+K(17)
C
C
C      COMPUTE CARGO VOLUME
C
C      VCARGO=K(9)
C
C
C      COMPUTE CREW VOLUME
C
C      VCREW=K(5)*NCREW+K(6)
C
C
C      COMPUTE LANDING GEAR BAY VOLUME
C
C      VLGEBAY=K(12)*WNG+K(13)
C
C
C      COMPUTE VBODY,VOTHER,SBODY AND VSTRUC USING K(10) SCALING DATA

```

```

IF(K(18).LE.1.0 ) GO TO 46
VBODY=K(18)*(VFUTK+VOXTK+K(23)
VOTHER=VBODY-VFUTK-VOXTK-VINSTK-VCREW-VCARGO-VSTRUC-VLGBAY-VPROP
*-VFUTK2-VOXTK2
SBODY=CSBODY*VBODY**C23
VSTRUC=K(10)*SBODY+K(11)
GO TO 60
46 CONTINUE

C
C   SAVE BODY VOLUME
C
C   VBODYX=VBODY
C
C   SUB TOTAL BODY VOLUME -STRUCTURE-MISC.
C
C   VBODYA=VOXTK+VFUTK+VINSTK          +VPROP+VCARGO+VCREW+VLGBAY
1+VFUTK2+VOXTK2
C
C   COMPUTE BODY SURFACE AREA
C
50 SBODY=CSBODY*VBODY2
C
C   COMPUTE STRUCTURAL VOLUME
C
C   VSTRUC=K(10)*SBODY+K(11)
C
C   COMPUTE MISCELLANEOUS VOLUME
C
C   VOTHER=K(18)*(VBODY-VCARGO-VSTRUC)+K(19)
C
C   SUM BODY VOLUME
C
C   VBODY=VBODYA+VSTRUC+VOTHER
C
C   IF(ABS(VBODYX-VBODY).LE.2.0) GO TO 60
C
C   SAVE BODY VOLUME
C
C   VBODYX=VBODY
C
C   UPDATE BODY VOLUME TO THE 2/3 POWER
C
C   IF(VBODY.LE.0.0) VBODY=1.0
VBODY2=VBODY**C23
C
C   GO TO 50
C
C   UPDATE BODY VOLUME TO THE 1/3 POWER
C
60 VBODY1=VBODY**C13
C
C   UPDATE BODY VOLUME TO THE 2/3 POWER

```

```

C
VBODY2=VBODY1**2
WGASPR = C(106)*VFUTK+C(107)*VOXTK+C(108)
WRESID = WRESID + WGASPR
C
IF(NWL.LE.0.ORVNWL.GE.8) NWL=6
C
TEST FOR WING SIZE METHOD-IF INPUT WING LOADING (FXWOVS) NOT EQUAL
C TO ZERO SIZE ON INPUT -IF INPUT WING LOADING EQUAL ZERO USE INPUT
C WING AREA
C
IF(FXWOVS.NE.0:0) GO TO 70
C
VARIABLE WING LOADING
C
WOVERS=0.0
IF (SWING.NE.0:0) WOVERS=WWAIT(NWL)/SWING
C
GO TO 80
C
FIXED WING LOADING
C
70 SWING=WWAIT(NWL)/FXWOVS
WOVERS=FXWOVS
C
COMPUTE AERO WING SPAN
C
80 GSPAN= CRT(ASRATO*SWING)
C
COMPUTE WING ROOT CHORD
C
CROOT = 2.*SWING/((1.+TPRATO)*GSPAN)
C
COMPUTE WING TIP CHORD
C
CTIP=CROOT*TPRATO
C
COMPUTE WING STRUCTURAL SPAN 50PC
C
CSPAN=0.0
IF (GSPAN.NE.0:0) CSPAN=GSPAN/COS(ATAN(TAN(ASWEEP/RTOD)-(.5*CROOT
1*(1.+TPRATO)/(GSPAN/2.))))
C
COMPUTE WING THICKNESS AT ROOT
C
TROOT=TOVERC*CROOT
C
COMPUTE BODY WIDTH
C
WBODY=CBBODY*VBODY1
C
COMPUTE WING EXPOSED AREA

```

```

C      SXPOS = SWING-(CROOT*BBODY-(.5*BBODY)**2*TAN(ASWEEP/RTOD))
C
C      COMPUTE HORIZONTAL TAIL AREA
C
C      SHORZ=CSHORZ*SWING
C      IF(C(234).EQ.0.) GO TO 82
C      SHORZ = C(234)
C      CSHORZ = SHORZ/SWING
82  CONTINUE
C
C      COMPUTE BODY SURFACE AREA
C
C      SBODY=CSBODY*VBODY2
C
C      COMPUTE VERTICAL TAIL AREA
C
C      SVERT=CSVERT*VBODY2
C      IF(C(235).EQ.0.) GO TO 84
C      SVERT = C(235)
C      CSVERT = SVERT/VBODY2
84  CONTINUE
C
C      COMPUTE FAIRING AREA
C
C      SFAIR=CSFAIR*SBODY
C
C      PLANFORM AREA
C
C      SPLAN=CSPLAN*VBODY2
C
C      COMPUTE BODY HEIGHT
C
C      HBODY=CHBODY*VBODY1
C
C      COMPUTE BODY LENGTH
C
C      LBODY=CLBODY*VBODY1
C
C      COMPUTE THERMAL PROTECTION AREA
C
C      IF(ITPS.LT.1.OR.ITPS.GT.8) ITPS=1
C      GO TO (90,100,110,120,130,140,150,160), ITPS
90  STPS(1)=0.
C      GO TO 170
100 STPS(1)=SBODY
C      GO TO 170
110 STPS(1)=SBODY+SHORZ
C      GO TO 170
120 STPS(1)=SBODY+SHORZ+SVERT
C      GO TO 170

```

```

130 STPS(1)=SBODY+SHORZ+SVERT+SWING
GO TO 170
140 STPS(1)=SHORZ+SVERT+SWING
GO TO 170
150 STPS(1)=SBODY+SWING
GO TO 170
160 STPS(1)=SBODY+SVERT
C
170 CONTINUE
C
C *****
C WEIGHTS
C *****
C
C COMPUTE WING WEIGHT
C
C MWING= C(2)*SWING + C(3)
C IF (TROOT.NE.0(0)) MWING=C(1)*(MWAIT(6)* LP* CSPAN*SWING/TROOT
1 *1.E-09)**C(12) + MWING
C
C COMPUTE VERTICAL TAIL WEIGHT
C
C MVERT = C(4)*SVERT**C(135)+C(24)*SVERT+C(5)
C
C COMPUTE HORIZONTAL TAIL WEIGHT
C
C MHORZ=C(6)*(MVERS**1.21*SHORZ** .814*Q** .467)**C(176)+C(25)*SHOR
2+C(7)
C
C COMPUTE FAIRING WEIGHT
C
C MFAIR=C(8)*SFAIR+C(9)
C
C SUM AERO SURFACES
C
C MSURF=MWING+MVERT+MHORZ+MFAIR
C
C COMPUTE INTEGRAL FUEL TANK WEIGHT
C
C MINFUT=C(10)*VPUTK+C(11)
C
C COMPUTE INTEGRAL LOX TANK WEIGHT
C
C MINOXT=C(138)*VOXTK+C(139)
C
C COMPUTE BODY BASIC STRUCTURE WEIGHT
C
C MBASIC=C(13)*SBODY+C(14)*VBDY+C(15)
C
C COMPUTE BODY SECONDARY STRUCTURE WEIGHT
C

```

```

C      WSECST=C(23)*SBODY+C(169)
C
C      COMPUTE THRUST STRUCTURE WEIGHT
C
C      WTHRST=C(168)*TTOT+C(163)
C
C      SUM BODY WEIGHT
C
C      WBODY=WINFUT+WINOXT+WBASIC+WSECST+WTHRST
C
C      COMPUTE TPS INSULATION WEIGHT
C
C      WINSUL=C(180)*STPS(1)+C(26)
C
C      COMPUTE TPS COVER PANELS WEIGHT
C
C      WCOVER=C(181)*STPS(1)+C(27)
C
C      SUM TPS WEIGHT
C
C      WTPS=WINSUL+WCOVER
C
C      COMPUTE LAUNCH EQUIPMENT WEIGHT
C
C      WLAUNCH=C(143)*WTO+C(144)
C
C      COMPUTE LANDING GEAR WEIGHT
C
C      WLG = C(30)*WHAIT(7)*C(182)+C(31)
C
C      COMPUTE DOCKING EQUIPMENT
C
C      WDOCK=C(147)*WHAIT(5)+C(148)
C
C      COMPUTE DEPLOYABLE AREA DEVICES
C
C      WDPLOY=C(145)*WHAIT(7)+C(146)
C
C      SUM LAUNCH AND RECOVERY GEAR
C
C      WLRD=WLAUNCH+WLG+WDOCK+WDPLOY
C
C      COMPUTE CREW WEIGHT
C
C      WPERS=C(97)*NCREW+C(98)
C
C      COMPUTE ENGINE MOUNT WEIGHT
C
C      WENGHT=C(183)*TTOT+C(184)
C
C      COMPUTE GIMBAL TORQUE

```



```

TDEL=0.0
IF(NENG5.NE.0.0.AND.PCHAM.NE.0.0) TDEL=750.*(TTOT/NENG5/PCHAM)**
1 1.25

```

```

C
C   COMPUTE GIMBAL SYSTEM WEIGHT
C
C   WSTAB = NENG5*VC(28)*TDEL**C(160))+C(161)
C
C   COMPUTE SECONDARY ROCKET ENGINE WEIGHT
C
C   WENG52=C(140)+FTOT2+C(141)
C
C   COMPUTE AIR BREATHING ENGINE WEIGHT
C
C   WABPR=C(210)*WNAIT(6)+C(211)
C
C   COMPUTE AIR BREATHING NACELLES WEIGHT
C
C   WNACEL=C(36)*WABPR+C(37)
C
C   TEST-FOR AIR BREATHING FUEL TYPE IF LH2 GO AROUND
C   JP-4 SYSTEM WEIGHT EQUATION
C
C   WBPUMP=0.0
C   WPRSYS=0.0
C   WDIST1=0.0
C   IF (C(212).NE.0.0.OR.C(213).NE.0.0) GO TO 180
C
C   COMPUTE JP-4 FUEL PUMPS WEIGHT
C
C   WBPUMP= C(149)*ANENG5*(1.75+.266*ANENG5)/1000.
C
C   COMPUTE JP-4 PRESSURE SYSTEM WEIGHT
C
C   WABFPS=0.0009*8(149)*ANENG5*ANTANK
C
C   COMPUTE JP-4 FUEL DISTRIBUTION SYSTEM WEIGHT
C
C   WDIST1=ANENG5*C(191)*SORT(C(149))
C
180 CONTINUE
C
C   COMPUTE ROCKET ENGINE WEIGHT
C
C   WENG5=C(32)*TTOT+C(219)*TTOT*C(220)**C(221)+C(33)*NENG5+WENGHT
C
C   COMPUTE ROCKET FUEL TANK WEIGHT NON-STRUCTURAL
C
C   WFUTK=C(39)*VFOTK+C(40)
C
C   COMPUTE ROCKET LOX TANK WEIGHT NON-STRUCTURAL
C

```

```

C      WOXTK=C(41)*VOXTK+C(42)
C      COMPUTE ROCKET SECONDARY FUEL TANK WEIGHT
C      WFUTK2=C(170)*VFUTK2+C(136)
C      COMPUTE ROCKET SECONDARY LOX TANK WEIGHT
C      WOXTK2=C(171)*VOXTK2+C(137)
C      COMPUTE ROCKET PROPELLANT TANK INSULATION WEIGHT
C      WINSTK=C(43)*SFUTK+C(77)*SOXTK+C(44)
C      COMPUTE ROCKET FUEL SYSTEM WEIGHT
C      WFUSYS=C(45)*TTOT+C(46)*NBODY+C(47)
C      COMPUTE ROCKET LOX SYSTEM WEIGHT
C      WOXSYS=C(48)*TTOT+C(49)*NBODY+C(50)
C      COMPUTE ROCKET PRESSURE SYSTEM WEIGHT
C      WPRSYS=C(51)*VFUTK+C(52)*VOXTK+C(187)
C      TEST-FOR AIR BREATHING FUEL TYPE IF LH2 GO AROUND
C      JP-4 SYSTEM WEIGHT EQUATION
C
C      GAL=0.0
C      WFUNCT=0.0
C      WDIST2=0.0
C      WFCNT=0.0
C      WREFUL=0.0
C      WDRANS=0.0
C      WSEAL=0.0
C      ABFSYS=0.0
C      WABFS = 0.
C      IF (C(212).NE.0.0.OR.C(213).NE.0.0) GO TO 190
C
C      COMPUTE JP-4 FUEL IN GALLONS
C
C      GAL=WABFU/6.5
C
C      COMPUTE JP-4 FUEL SYSTEM TANKS WEIGHT
C
C      IF(ANTANK.NE.0.0) WFUNCT=C(189)*(GAL/ANTANK)**.6*ANTANK+C(190)
C
C      COMPUTE JP-4 FUEL SYSTEM DISTRIBUTION WEIGHT
C
C      WDIST2=.255*GAL**.7*ANTANK**.25
C

```

```

C      COMPUTE JP-4 FUEL SYSTEM CONTROLS WEIGHT
C
C      WFCONT=.169*ANTANK*SQRT(GAL)
C
C      COMPUTE JP-4 FUEL SYSTEM REFUEL SYSTEM WEIGHT
C
C      WREFUL=ANTANK*.43+.45*GAL**C13)
C
C      COMPUTE JP-4 FUEL SYSTEM REFUEL SYSTEM WEIGH
C
C      WDRANS=.159*GAL**.65
C
C      COMPUTE JP-4 FUEL TANK SEALING
C
C      IF(ANTANK.NE.0) WSEAL=.045*ANTANK*(GAL/ANTANK)**.75
C
C      SUM JP-4 FUEL SYSTEM WEIGHT
C
C      ABFSYS=WBPUMP+WDIST1+WDIST2+WFCONT+WREFUL+WDRANS+WSEAL+WFUNCT
1+WABFPS
C
C      SUM JP FUEL SYSTEM WEIGHT LESS TANKS
C
C      WABFS=ABFSYS-WFUNCT
C
190 CONTINUE
C
C      COMPUTE AIRFUEL TANK WEIGHT
C
C      WABFTK=C(212)*WABFU+C(213) +ABFSYS
C
C      SUM PROPULSION SYSTEM WEIGHT
C
C      WPROP=WENG5+WNACEL+WFUTK+WOXTK+WINSTK+WFUSYS+WOXSYS+WPRSYS+WENG5
1+WFUTK2+WOXTK2+WABPR+WABFTK
C
C      COMPUTE AERO SURFACE CONTROL WEIGHT
C
C      WAERO=0.0
C      WAERO=C(55)*(WNAIT(5)**.689*(LBODY+CSPAN)**.267)**C(185)+C(56)
C
C      COMPUTE SEPARATION SYSTEM WEIGHT
C
C      IF(JUMP.EQ.0.0) WAUXT=C(153)*WTO+C(154)
C      IF(JUMP.EQ.1.0) WAUXT=C(153)*WPAYL+C(154)
C
C      COMPUTE ACS SYSTEM WEIGHT
C
C      WACS=C(156)*WNAIT(4)**C(155)+C(157)
C
C      COMPUTE ACS PROPELLANT TANK WEIGHT
C

```

```

C      WACSTK=C(164)*(WACSF0+WAGRES)+C(165)
C
C      SUM ORIENTATION CONTROLS WEIGHT
C
C      WORSUL=WSTAB+WACS+WAERO+WAUXT+WACSTK
C
C      COMPUTE GUIDANCE AND NAVIGATION SYSTEM WEIGHT
C
C      HGNAV=C(68)
C
C      COMPUTE INSTRUMENTATION SYSTEM WEIGHT
C
C      WINST =C(69)*LBODY+C(70)
C
C      COMPUTE COMMUNICATION SYSTEM WEIGHT
C
C      WCOMM=C(71)*NCREW+C(72)
C
C      SUM AVIONICS SYSTEM
C
C      HAVIOC=HGNAV+WINST +WCOMM
C
C      COMPUTE PRIME POWER SOURCE AND DISTRIBUTION
C
C      WSORCE=C(62)*(HAVIOC+WABPS)**C(63)+C(64)
C
C      COMPUTE PRIME POWER SOURCE TANKAGE
C
C      WPOWTK=C(29)*WPOWFO+C(60)
C
C      SUM ELECTRICAL POWER SYSTEMS
C
C      WPOWER=WSORCE+WPOWTK
C
C      COMPUTE HYDRAULIC/PNEUMATIC SYSTEM
C
C      WHYCAD=C(65)*(((SWING+SHORZ+SVERT)*.001*Q)**1.3125+(LBODY+CSPAN)
147  2**1.06125)**C(66)+C(67)
C
C      SUM POWER CONTROL AND DISTRIBUTION
C
C      WPOWCD=WHYCAD
C
C      COMPUTE PERSONNEL ACCOMMODATIONS WEIGHT
C
C      WPPROV=C(74)*HWAIT(7)+C(75)*NCREW+C(76)
C
C      COMPUTE DRY WEIGHT
C
C      WDRY=WSURF+WBODY+WTPS+WLRD+WPROP+WORSUL+WPOWCD+HGNAV+WINST +WCOMM+
147  1WPPROV +WPOWER
C

```

```

C      COMPUTE GROWTH AND CONTINGENCY
C
C      WCONT=C(96)*WDRY+C(162)
C
C      SUM WEIGHT WEIGHT EMPTY
C
C      WEMPTY=WDRY+WCONT
C
C      COMPUTE CARGO WEIGHT
C
C      WCARGO=C(102)*WPASS+C(103)
C
C      COMPUTE PASSENGER WEIGHT
C
C      WPASS=C(104)*NPASS+C(105)
C
C      SUM PAYLOAD WEIGHT
C
C      WPAYL=WPASS+WCARGO
C
C      SUM OPERATING WEIGHT EMPTY
C
C      WNET =WEMPTY+WRESID+WPERS+WPAYL+WRESRV
C
C      SUM ZERO FUEL WEIGHT
C
C      WZROFU=  WNET+      WFULOS+WOXLOS
1          +WACSFO      +WPOWFO+WOIL+WABFU+WFROS
C
C      SUM TAKEOFF WEIGHT
C
C      WTO=WZROFU+WFUOX+WDECAY-WFUEL(1)-WOX(1)
C
C      SUM GROSS WEIGHT
C
C      WGROSS=WT0+MPREIG+WFUEL(1)+WOX(1)
C
C      UPDATE WWAIT(1)
C
C      WWAIT(1)=WGROSS -MPREIG
C
C      GROSS WEIGHT LESS PAYLOAD
C
C      WABC=WGROSS-WPAYL
C
C
C      MISCELLANEOUS CONTROL STATEMENTS
C
C      IF(NLISTO.NE.0) WRITE(6,KASS1)
C      IF(NLISTO.NE.0) WRITE(6,HASS2)
C

```

GDC-08870-002

IF (WGROSS.GT.20000000..AND.JUMP.EQ.1.OR.WGROSS.GT.10000000..AND.J
1UMP.EQ.0) GO TO 210

C

RETURN

C

210 WRITE(6,MASS1)
WRITE(6,MASS2)

C

CALL EXIT

C

END

Appendix IV-35 Subroutine SOLVE

```

C
C
SUBROUTINE SOLVE
THIS VERSION OF SOLVE DRIVES SUBROUTINE WTSCH TO CLOSE ON WEIGHT

REAL ISP,K,LF,MR,NCREW,LBODY,NPASS
REAL NENG5
REAL MIN,MAX
COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
COMMON/CINPUT/
1ANENG5 ,ANTANK ,ASRATO ,ASWEEP ,C(300) ,CBBODY ,CFUEL(6),
2CHBODY ,CLBODY ,CSBODY ,CSFAIR ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN ,CSVERT ,CSWING ,CTHRST ,CTHST2 ,DEF(5) ,FXWOVS ,
4TSP(6) ,ITPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5NENG5 ,NLISTO ,NPASS ,NWL ,PCHAM ,QMX ,RHOFU ,
6RHOFU2 ,RHOX ,RHOX2 ,SHODY ,TOL ,TOVERC ,TPRATO ,
7TYTAIL ,VBODY ,WGROSS
COMMON/VOLCAL/BBODY,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HBODY ,LBODY ,RTOD ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPLAN ,STPS(1) ,SVERT ,SWING ,SXPOS ,TDEL ,TROOT ,
4TTOT ,TTOT2 ,TTOTAL ,VBOYA ,VBOY1 ,VBOY2 ,VCARGO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGRAY ,VOTHER ,VOXTK ,
6VOXTK2 ,VPROP ,VSTRUP
COMMON/WTCALC/ ABFSYS ,WABFTK ,WABFU ,WABPR ,WACRES ,
1WACS ,WACSF0 ,WACSTK ,WAERO ,WAUXT ,WBASIC ,WBODY ,
2WBUMP ,WCARGO ,WCOMM ,WCONT ,WCOVER ,WDCAY ,WDIST1 ,
3WDIST2 ,WDOCK ,WDPLY ,WDPANS ,WDRY ,WELCAD ,WEMPTY ,
4WENGHT ,WENG5 ,WENG52 ,WFAIR ,WFCONT ,WFCAY ,WFROST ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFOOX ,WFURES ,
6WFUSYS ,WFUTK ,WFUTK2 ,WFUTOT ,WFUTEP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLOSS ,WLRD ,WNACEL ,WODCAY ,
9WOIL ,WOILRS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,WOXID ,
1WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPER5 ,WPOWCO ,WPOWER ,WPOWFO ,
3WPOWRS ,WPOWTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
4WRESID ,WRESRV ,WSFAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF ,WTABC ,WTHRST ,WTO ,WTPS ,WVERT ,WWAIT(10),
6WNET ,WNING ,WZROFU
IF(SW(13).GT.2.5) WRITE(6,1001)

C
WK=MR(3)
IF(TOL.LT.1..OR.TOL.GT.100.) TOL = 5.
ITER = 0
MAX=2.5*WGROSS
MIN=0.4*WGROSS

C
V1=VBODY
W1 = WGROSS

```

```

      CALL WTSCH
5005  CONTINUE
      DW1=WGROSS-W1
      DV1=VBODY -V1

10  CONTINUE
      W2=WGROSS+WK*DW1
      R=W2/WGROSS
      V2=VBODY*R
      VBODY=V2
      WGROSS=W2
      ITER=ITER+1
      CALL WTSCH
      DW2=WGROSS-W2
      DV2=VBODY -V2
      IF(W2.GT.MAX.OR.W2.LT.MIN) GO TO 120
      IF(ABS(DW2).LT.TOL.AND.ABS(DV2).LT.5.) GO TO 100
      IF(ITER.GT.15)GO TO 120
      WK=(W2-W1)/(DW1-DW2) - 1.
      IF(WK.GT.5.) WK=5.
      IF(WK.LT.2.) WK=2.
      W1 = W2
      DW1 = DW2
      V1 = V2
      DV1 = DV2
      IF(SW(13).GT.2.5) WRITE(6,1000)  W2,DW2, V2,DV2, WK,ITER
      IF(0) 5005,200,5005
      THIS CARD NECESSARY TO PROGRAM AROUND COMPILER
      OPTIMIZATION BUG ON UNIVAC 1108 EXEC II.

110 CONTINUE
100 CONTINUE
      IF(SW(13).EQ.2.0) WRITE(6,1001)
      IF(SW(13).GT.1.5) WRITE(6,1000)  W2,DW2, V2,DV2, WK,ITER
      RETURN
120 CONTINUE
      WRITE(6,1002)
1002 FORMAT(10X,30H ***** BLOWUP IN SOLVE *****  //)
      WRITE(6,1001)
      WRITE(6,1000)  W2,DW2, V2,DV2, WK,ITER
      RETURN
1000 FORMAT( 5X,2( F12.2,F12.4),F12.3,I4 )
1001 FORMAT(/9X,2HW2 10%,3HDW2 8X, 2HV2 10X, 3HDV2 10X,
1 2HWK 8X,4HITER )

      END

```



```

SUBROUTINE TAMPER (NCALL)
REAL KIN
REAL ISP,K,LF,MR,NCREW,LBODY,NPASS
REAL NENG5
REAL LRODY0,LRODYB
DIMENSION SK0(30),SC0(300),T04(6),T020(10),T027(6),T034(6),
1 T048(10),T049(10),T050(10),T057(6)
COMMON / JUMPY / JUMP, WRT6, W800
COMMON/TAMP/WGROSS,OTTOT,WFOUXO,TBTO,STTOT
COMMON/ORBIX/
1 T01,T02,T03,T04,T05,T06,T07,T08,T09,T010,T011,T012,T013,T014,
2 T015,T016,T017,T018,T019,T020,T021,I022,I023,I024,I025,I026,T027,
3 I028,I029,I030,I031,I032,T033,T034,T035,T036,I037,T038,T039,T040,
4 T041,T042,T043,T044,T045,T046,T047,T048,T049,T050,T051,T052,T053,
5 T054,T055,T056,T057,SK0,SC0
DIMENSION SKB(30),SCB(300),TB4(6),TB20(10),TB27(6),TB34(6),
1 TB48(10),TB49(10),TB50(10),TB57(6)
COMMON/ORBINY/
1 TB1,TB2,TB3,TB4,TB5,TB6,TB7,TB8,TB9,TB10,TB11,TB12,TB13,TB14,
2 TB15,TB16,TB17,TB18,TB19,TB20,TB21,I022,I023,I024,I025,I026,TB27,
3 I028,TB29,TB30,TB31,I032,TB33,TB34,TB35,TB36,I037,TB38,TB39,TB40,
4 TB41,TB42,TB43,TB44,TB45,TB46,TB47,TB48,TB49,TB50,TB51,TB52,TB53,
5 TB54,TB55,TB56,TB57,SKB,SCB
COMMON/CINPUT/
1ANENG5 ,ANTANK ,ASRATO ,ASWEEP ,C(300) ,C0BODY ,CFUEL(6),
2CH00DY ,CLBODY ,CSBODY ,CSFAIR ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN ,CSVERT ,CSHING ,CTHRST ,CTHST2 ,DEF(5) ,FXWOVS ,
4ISP(6) ,ITPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5NENG5 ,NLISTO ,NPASS ,NWL ,PCHAM ,OMX ,RHOFU ,
6RHOFU2 ,RHOX ,RHOX2 ,SRODY ,TOL ,TOVERC ,TPRATO ,
7TY ,AIL ,VBODY ,WGROSS
COMMON/VOLCAL/BBODY,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HRODY ,LBODY ,RTON ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPLAN ,STPS(1) ,SVERT ,SWING ,SXPOS ,TDEL ,TROOT ,
4TTOT ,TTOT2 ,TTOTAL ,VRODYA ,VRODY1 ,VRODY2 ,VCARGO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGRAY ,VOTHER ,VOXTK ,
6VOXTK2 ,VPROP ,VSTRUC
COMMON/WTCALC/ ABFSYS ,WARFTK ,WABFU ,WABPR ,WACRES ,
1WACS ,WACSF0 ,WACSTK ,WAERO ,WAUXT ,WBASIC ,WRODY ,
2WRPUMP ,WCARGO ,WCOMM ,WCONT ,WCOVER ,WDECAY ,WDIST1 ,
3WDIST2 ,WDOCK ,WDPLOY ,WDTRANS ,WDRY ,WELCAD ,WEMPTY ,
4WENGMT ,WENG5 ,WENG52 ,WFAIR ,WFCONT ,WFDCAY ,WFROST ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFOUX ,WFURES ,
6WFUSYS ,WFUTK ,WFUTK2 ,WFUTOT ,WFUTPP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLOSS ,WLRD ,WNACEL ,WODCAY ,
9WOIL ,WOILRS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,WOXID ,
1WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPER5 ,WPOWER ,WPOWER ,WPOWFO ,
3WPOWRS ,WPOWTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
4WRESID ,WRESRV ,WSFAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF ,WTABC ,WTHPST ,WTO ,WTPS ,WVERT ,WWAIT(10),

```

```

6WNET      ,WWING      ,WZROFU
COMMON/SUMVM/WOP,      WABFUB,WFUOXB,WORYB,WGROSS,WOTHB,
1 VFUTKB,VOXTKB,VOTMB,VBODYB,VARFUB,
2 LRODYB,SBODYB,SPLANB,WOVRSB,
1          WPAYLO,WORYO,WOTHO,  WABFUO,
2 VFUTKO,VOXTKO,VCARO,VOTHO,VBODYO,
3 LRODYO,SBODYO,SPLANO,WOVRSO
4 ,WORRTO,WORRTR,WRTRNO,WRTRNB,WENTRO,WENTRB,WLANDB,WLANDO
5 , WCONTO,WCONTR
COMMON NSEC,T,TR,TS(15),V(15),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
COMMON/EMS/WORBIT,WETURN,WENTRY,WLAND
COMMON/OS/OSPLAN

C
IFIFE = SE(5)
V(7)=  WGROSS
C
AERO. REFERENCE AREAS
DO 2 I=1,3
Q(5,I) = TB45
2 Q(6,I) = TB45
DO 3 I = 4,7
Q(5,I) = T045
3 Q(6,I) = T045
DO 4 I=8,15
Q(5,I) = TB45
4 Q(6,I) = TB45
C
Q(26,3) = -WPAYL
Q(23,2)= V(7)-  WFUOX
C
THRUST
Q(1,5)= OTTOT * SE(23)
Q(1,6)= OTTOT * SE(24)
Q(1,7)= OTTOT *SE(25)
C
ISP
Q(2,5)=SE(15)
Q(2,6)=SE(17)
Q(2,7)=SE(19)
BWDOT1 = BTTOT*SE(26)/SE(1)
BWDOT2 = BTTOT * SE(27) / SE(11)
SE(40) = 0.
GO TO (5,9),IFIFE
C
SIMULTANEOUS FIRING OF ROOSTER AND ORBITER
C
THRUST AND ISP
5 Q(1,1) = OTTOT*SE(21) + BTTOT*SE(26)
Q(1,2) = OTTOT*SE(22) + BTTOT*SE(27)
OWDOT1 = OTTOT*SE(21)/SE(2)
WDOTT1 = OWDOT1 + BWDOT1
OWDOT2 = OTTOT*SE(22)/SE(12)

```

```

WDOTT2 = OWDOT2 + BWDOT2
Q(2,1)=Q(1,1)/WDOTT1
Q(2,2)=Q(1,2)/WDOTT2
Q(3,1) =(OWDOT1 * SE(4) + BWDOT1 * SE(3) ) / WDOTT1
Q(3,2) =(OWDOT2 *SE(14) + BWDOT2 * SE(13) ) / WDOTT2
IF(SE(39) .EQ. 0.) GO TO 16
T2 = ( WFOOX - BWDOT1*Q(23,1))/BWDOT2
SE(40) = OWDOT1*Q(23,1)+OWDOT2*T2
Q(26,3) = Q(26,3) + SE(40)
Q(23,2) = Q(23,2) - SE(40)
GO TO 16
9 CONTINUE
C SEQUENTIAL FIRING OF BOOSTER AND ORBITER OR
C TANDEM FIRING OF BOOSTER AND ORBITER
C THRUST AND ISP
Q(1,1) = RTTOT * SE(26)
Q(1,2) = RTTOT * SE(27)
Q(2,1) = Q(1,1) /BWDOT1
Q(2,2) = Q(1,2) /BWDOT2
Q(3,1) = SE(3)
Q(3,2) = SE(13)
16 Q(3,5) = SE(16)
Q(3,6) = SE(18)
Q(3,7) = SE(20)
SV(11) = Q(1,1)/V(7)
SV(13) = SV(11)*Q(3,1)/Q(2,1)
IF(SQ(20,1).LE.0.) GO TO 18
V(7) = V(7) + SQ(21,5)
SV(13) = (Q(1,1) * Q(3,1) / Q(2,1) + SQ(21,4) ) / V(7)
18 IF(NCALL.EQ.1) RETURN
C GET PARAMETERS FOR PITX AND OPRIME GUESSES
PIT=Q(62,1)
SV(30)=SV(13)-1.
IF(SE(30).NE.0.) SV(30) = .5*Q(1,2)/V(7)*Q(3,2)/Q(2,2)+.5*SV(13)
1 -1.
DELMU=TR34(3)-2.5
SV(23)=(1.+2.80*SV(30)*DELMU)*(1.+.0072/ SV(30)**5 )
IF(SW(7).GT.1.) GO TO 20
C GUESS PITX IF NONE HAS BEEN INPUT
TEMDEG=SIN(V(4)*.01745329251)
TEM= SQRT(TEMDEG)
SV(28)= .18 - .10 * TEM
SV(14)=SV(30)-SV(28)
PFIFTY=(30.-15.*TEM)*SV(14)**2.15
DQDP=SV(17)*SV(23)
DMU=TR34(3)-2.90
DGDQ= -6.0 +7.2 *DMU
ICODE=IFIX(VQ(7,1))
IF(ICODE.EQ.92) SV(24)=DQDP
IF(ICODE.EQ. 3) SV(24)=DQDP*DGDQ
TARG=VQ(25,1)
IF(ICODE.EQ. 3) TARG= 2.1 - 1.0 * DMU + VQ(25,1)/DGDQ

```

```

IF(ABS(SV(24)).LT..001)SV(24)=100.
IF(PIT.LE.0.) Q(62,1)=PFIFTY + (TARG-1.700)/DQDP
Q(62,2)=Q(62,1)
GO TO 30

```

```

20 CONTINUE

```

```

  ADJUST QPRIME = D LOG Q/ D PITX
SV(17)=SV(24)/SV(27)
SV(24)=SV(17)*SV(23)
  ADJUST PITX
SV(14)=SV(30)-SV(28)
Q(62,1)=PIT*(SV(14)/SV(15))**2.15 *(1.+.10*(TB34(3)-SV(16)))
Q(62,2)=Q(62,1)

```

```

30 CONTINUE

```

```

  ESTIMATE ORBITER BURN TIME FOR LINEAR TAN STEERING OR ALPHA DOT
  W1 IS ORBITER IGNITION WEIGHT
W1=-Q(26,3)

```

```

  W3 IS WT AT ORBITAL INJECTION

```

```

W3 = W1 - WFUOXO + SE(40)

```

```

  WS IS WT AT WHICH AXIAL G LIMIT IS REACHED

```

```

WS = OTTOT*SE(24) / Q(66,7)

```

```

WDOT = OTTOT*SE(24) / Q(2,7)

```

```

RT57 = (W1-WS) / WDOT + Q(2,7)/Q(66,7) * ALOG(WS/W3)

```

```

Q(37,7)=-Q(45,6)/(BT57-Q(23,5))*0.7

```

```

Q(37,6)=Q(37,7)

```

```

Q(37,5)=RT57

```

```

  Q(37,7)=Q(37,6) IS ALPHA DOT FOR ORBITER TRAJECTORY

```

```

  Q(37,5) IS ORBITER BURN TIME ESTIMATE FOR LINEAR TANGENT STEERING

```

```

SV(15)=SV(14)

```

```

SV(16)=TB34(3)

```

```

SV(27)=SV(23)

```

```

SV(34) = WAIT(6)

```

```

RETURN

```

```

ENTRY ORBSUM

```

```

ORBITER SUMMARY WEIGHTS

```

```

SQ(37,4) = WFURES + WOXRES

```

```

WOP=WFU2(1)+WOX2(1)

```

```

WPAYLO = WPAYL

```

```

WABFUO = ABS(WABFU)

```

```

WCONTO = WCONT

```

```

WDRYO = WDRY

```

```

WOTHO = WGROSS - WFUOXO - WABFUO - WPAYL - WDRY - SQ(37,4) - WCONT

```

```

WORBTO = WORBIT

```

```

WTRTNO = WETURN

```

```

WENTRO = WENTRY

```

```

WLANDO = WLAND

```

```

ORBITER SUMMARY VOLUMES

```

VFUTKO = VFUTK
 VOXTKO = VOXTK
 VCARGO = VCARGO
 VOTH0 = VBODY - VCARGO - VOXTK - VFUTK
 VBODY0 = VBODY

C
 C
 C

ORBITER SUMMARY GEOMETRY

LBODY0 = LBODY
 SBODY0 = SBODY
 SPLAN0 = SPLAN
 WOVRS0 = WOVERS
 RETURN

C
 C
 C
 C

ENTRY ROOMSUM

BOOSTER SUMMARY WEIGHTS

SQ(37,5) = WFURES + WOXRES
 WABFUB = ABS(WABFU)
 WFUOXB = WFUOX
 WCONTR = WCONT
 WORYR = WORY
 WGROSR = WROO
 WOTHR = WGROSR - WFUOX - WABFUB - WORY - SQ(37,5) - WCONT
 WORBTB = WORBIT
 WTRNRB = WETURN
 WENTRB = WENTRY
 WLANDR = WLAND

C
 C
 C

BOOSTER SUMMARY VOLUMES

VFUTKB = VFUTK
 VOXTKB = VOXTK
 VOTHR = VBODY - VFUTK - VOXTK
 VABFUB = VOXTK? + VFUTK?
 VBODYB = VBODY

C
 C
 C

BOOSTER SUMMARY GEOMETRY

LBODYB = LBODY
 SBODYB = SBODY
 SPLANB = SPLAN
 WOVRSB = WOVERS
 RETURN
 END

PRECEDING PAGE BLANK NOT FILMED

Appendix IV-37 Subroutine PRINTW

GDC-08870-002

C SUBROUTINE PRINTW
C *** PRINTW ***

C PRINTW - PRINTS WEIGHTS
C

```

REAL KWSTAB
REAL ISP,K,LF,MR,NCREW,LBODY,NPASS
REAL NENG5
REAL KHARPR,KHARFT,KHABFU
*,KILO ,KWSURF,KWING,KWVERT,KWHORZ,KWFAIR,KWBODY,KWINFU,KWBASI,
1KWSECS,KWTPS ,KWGOVE,KWINSU,KWLG ,KWPROP,KWENG5,KWNACE,KWFUTK,
2KWOXTK,KWINST,KWFUSY,KWOXSY,KWPRSY,KWAERO,KWPOWE,KWAPU ,KWPOWT,
3KWELCA,KWHYCA,KWGNV,KWPPRO,KWDPY ,KWCONT,KWPERS,KWPAYL,KWCARG,
4KWPASS,KWPESI,KWGASP,KWFUTR,KWOXTR,KWSRTR,KWRESR,KWFURE,
*KWOXRF,KWPOWR,KWOILR,KWACRE,      KHWET ,KWLOSS,KWFULO,KWOXLO,
*KWPOWF,KWOIL ,KWDECA,KWFUOX,KWFUL ,KWPREI,KWGROS
*,KFUTK?,KOXTK?
REAL KWLRO,KWLANC,KWDPLO,KWDOCK,KWORSU,KWAUXT,KWACS
*,KWACSF,KWSORC,KWINS1,KWFU2,KWOX2
1,KWINOX,KWTHPS,KWENG2,KWPOWC,KWTO,KWOXID,KWACST,KWBOO
COMMON / JUMPY / JUMP, WRIG, WROO
COMMON/CINPUT/
1ANENG5 ,ANTANK ,ASPART ,ASWEEP ,C(300) ,CBBODY ,CFUEL(6),
2CHBODY ,CLBODY ,CSBODY ,CSFAIP ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN ,CSVERT ,CSWING ,CTHRST ,CTHST2 ,DEF(5) ,FXWOVS ,
4ISP(6) ,ITPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5NENG5 ,NLISTO ,NPASS ,NWL ,PCHAM ,Q ,RHOFU ,
6RHOFU2 ,RHOX ,RHOX2 ,SBODY ,TOL ,TOVERC ,TPRATO ,
7TYTAIL ,VBOOY ,WGROSS
COMMON/VOLCAL/BRODY,CROOT ,CSPAN ,CTIP ,GAL ,CSTAB ,
2HRODY ,LBODY ,RTOD ,SFAIR ,SFUTK ,SHORZ ,SUTY ,
3SPLAN ,STPS(1) ,SVERT ,SWING ,SXPOS ,TDEL ,TRIDY ,
4TTOT ,TTOT2 ,TTOTAL ,VBOOYA ,VBOOY1 ,VBOOY2 ,VGRADO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGBAY ,VOTHER ,VOXTK ,
6VOXTK? ,VPROP ,VSTRUC
COMMON/WTCALC/ ABFSYS ,HARFTK ,HARFU ,HARPR ,HACRES ,
1HACS ,HACSF0 ,HACSTK ,HAERO ,HAUXT ,HBASIC ,HBODY ,
2HWPUMP ,WCARGO ,WCOMM ,WCONT ,WCOVER ,WDECAY ,WDIST1 ,
3WDIST2 ,WDOCK ,WDPOY ,WDPRNS ,WDRY ,WELCAD ,WEMPTY ,
4WENGMT ,WENG5 ,WENG5? ,WFAIR ,WFCONT ,WFDCAY ,WFROST ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFUOX ,WFURES ,
6WFUSYS ,WFUTK ,WFUTK? ,WFUTOT ,WFUTRP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLOSS ,WLRD ,WNACEL ,WODCAY ,
9WOIL ,WOILRS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,WOXID ,
1WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPERS ,WPOWC ,WPOWER ,WPOWFO ,
3WPOWRS ,WPOWTK ,WPPROV ,WPRETG ,WPROP ,WPRSYS ,WREFUL ,
4WRESTO ,WRESPV ,WSEAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF ,WTARC ,WTHRST ,WTO ,WTPS ,WVERT ,WHAIT(10),
6HWET ,HWING ,W7ROFU
KILO = 0.4535924

```

KWABPR = WABPR * KILO
KWABFT = WABFTK * KILO
KWABFU = WABFU * KILO
KWSTAB=KILO*WSTAR
KWSURF = KILO*WSURF
KWWING = KILO*WWING
KWVERT = KILO*WVERT
KWHORZ = KILO*WHORZ
KWFAIR = KILO*WFAIR
KWACSF = KILO*WACSF0
KWBODY = KILO*WBODY
KWBASI = KILO*WBASIC
KWSECS = KILO*WSECS
KWTPS = KILO*WTPS
KWCOVE = KILO*WCOVER
KWDPLO=KILO*WDPLY
KWINSU = KILO*WINSUL
KWDOCK = KILO * WDOCK
KWLG = KILO*WLG
KWPROP = KILO*WPROP
KWGROS = KILO*WGROSS
KWTO = KILO* WTO
KWENGs = KILO*WENGs
KWNACE = KILO*WNACEL
KWFUTK = KILO*WFKTK
KWOXTK = KILO*WOKTK
KWINST = KILO*WINSTK
KWFUSY = KILO*WFUSYS
KWOXSY = KILO*WOKSYS
KWACS = KILO*WACS
KWAUXT = KILO* WAUXT
KWPPSY = KILO*WPPSYS
KWSORC = KILO*WSORCE
KWAERO = KILO*WAERO
KWPOWE = KILO*WPOWER
KWPOWT = KILO*WPOWTK
KWELCA = KILO*WELCAD
KWHYCA = KILO*WHYCAD
KWGNV = KILO*WGNV
KWINS1 = KILO*WINST
KWCOMM = KILO*WCOMM
KWPPRO = KILO*WPPROV
KWLRD = KILO*WLRD
KWLANC = KILO*WLANCH
KWDRY = KILO*WDRY
KWCONT = KILO*WCONT
KWPERs = KILO* WPERs
KWPAYL = KILO*WPAYL
KWCARG = KILO*WCARGO
KWPASS = KILO*WPASS
KWRESI = KILO*WRESID
KWGASP = KILO*WGASPR

```

KWFUTP = KILO*WFUTRP
KWOXTR = KILO*WOXTRP
KWSRTR = KILO*WSRTRP
KWRESR = KILO*WRESRV
KWFURF = KILO*WFURES
KWOXRE = KILO*WOXRES
KWPOWR = KILO*WPOWRS
KWOILR = KILO*WOILRS
KWACPE = KILO*WACRES
KWMET = KILO*WMET
KWLOSS = KILO*WLOSS
FROST=KILO*WFRST
KWFULO = KILO*WFULOS
KWOXLO = KILO*WOXLOS
KFUTK2=KILO*WFUTK2
KWOXK2=KILO*WOXK2
KWPOWF = KILO*WPOWFO
KWOIL = KILO*WOIL
KWDECA = KILO*WDECA
KWFUOX = KILO*WFUOX
KWFUL = KILO*WFUL
KWOXID = KILO*WOXID
KWFU2 = KILO * WFU2(1)
KWOX2 = KILO * WOX2(1)
KWORSU = KILO*WORSUL
KWACST = KILO*WACSTK
KWPREI = KILO*WPREIG
KWINFU = KILO * WINFUT
KWINOX =KILO*WINOXT
KWTHRS =KILO*WTHRST
KWENG2 =KILO*WENG2S
KWPOWC =KILO*WPOWCD
KWACRE =KILO*WACRES
WRITE(6,66)
66 FORMAT (1H1)
CALL DATE
67 CONTINUE
IF(JUMP.EQ.1) GO TO 45
WRITE(6,25)
25 FORMAT (1H0,32HWEIGHT BREAKDOWN - ORBITAL STAGE,33X6HPOUNDS,20X
* 9HKILOGRAMS)
GO TO 50
45 WRITE (6,26)
26 FORMAT (1H0,31HWEIGHT BREAKDOWN - BOOST STAGE ,33X6HPOUNDS,20X
* 9HKILOGRAMS)
50 WRITE(6,100) WSURF,KWSURF
150 WRITE(6,200) WWING,KWWING
250 WRITE (6,13500) WHORZ,KWHORZ
350 WRITE(6,400)WVEPT,KWVERT
650 WRITE(6,700)WFAIR,KWFAIR
750 WRITE(6,800)WBODY,KWBODY
WRITE(6,900) WINFUT,KWINFU

```



```

        WRITE(6,1000) WINOXT,KWINOX
1050    WRITE(6,1100) WRASTC,KWRAST
        WRITE(6,1300) WSEOST,KWSECS
1150    WRITE(6,1400) WTHRST,KWTHRS
1550    WRITE(6,1600) WTPS,KWTPS
1750    WRITE(6,1800) WCOVER,KWCOVE
1950    WRITE(6,1900) WINSUL,KWINSU
2150    WRITE(6,2200) WLRD,KWLRD
2250    WRITE(6,2300) WLANCN ,KWLANC
2350    WRITE(6,2400)   WDPLOY,KWDPLO
2450    WRITE(6,2500)   WLG,KWLG
2550    WRITE(6,2600) WDOCK,KWDOCK
2750    WRITE(6,2800) WPROP,KWPROP
2850    WRITE(6,2900) WENG5,KWENG5
        WRITE(6,2870) WENG52,KWENG2
        WRITE(6,3300) WFUTR,KWFUTR
        WRITE(6,3400) WOXTK,KWOXTK
        WRITE(6,3410) WFUTR2,KFUTR2
        WRITE(6,3420) WOXTK2,KOXTK2
        WRITE(6,3500) WINSTK,KWINST
3550    WRITE(6,3600) WFUSYS,KWFUSY
        WRITE(6,3700) WOXSYS,KWOXSY
        WRITE(6,3800) WPRSYS,KWPRSY
        WRITE(6,4110) WABPR,KWABPR
        WRITE(6,3100) WNACEL,KWNACE
        WRITE(6,4115) WABFTK,KWABFT
4250    WRITE(6,4300) WORSUL,KWORSU
4350    WRITE(6,4400) WAUXT,KWAUXT
        WRITE(6,4470) WSTAB,KWSTAB
4450    WRITE(6,4500) WAERO,KWAERO
4550    WRITE(6,4600) WACS,KWACS
        WRITE(6,4800) WACSTK,KWACST
4850    WRITE(6,4900) WPOWER,KWPOWE
4950    WRITE(6,5000) WSORCE,KWSORC
        WRITE(6,5300) WPOWTK,KWPOWT
5350    WRITE(6,5400) WPOWCD,KWPOWC
5550    WRITE(6,5600) WHYCAD,KWHYCA
5650    WRITE(6,5700) WGNV,KWGNV
5750    WRITE(6,5800) WINST,KWINS1
5850    WRITE(6,5900) WCOMM,KWCOMM
6550    WRITE(6,6600) WPPROV,KWPPRO
7250    WRITE(6,7300) WDRY,KWDRY
        WRITE(6,7400) WCONT,KWCONT
7450    WRITE(6,7500) WPERS,KWPERS
7750    WRITE(6,7800) WPAYL ,KWPAYL
7851    WRITE(6,7900) WPASS, KWPASS
        WRITE(6,7860) WCARGO, KWCARG
8350    WRITE(6,8400) WRESID,KWRESI
        WRITE(6,8500) WGASPR,KWGASP
8550    WRITE(6,8600) WFUTRP,KWFUTP
        WRITE(6,8700) WOXTRP,KWOXTR
8850    WRITE(6,8900) WSRTRP,KWSRTR

```

```

8950 WRITE(6,9000)WRFSRV,KWRESR
9050 WRITE(6,9100)WFURES,KWFUPE
      WRITE(6,9200)WOXRES,KWOXRE
      WRITE(6,9300)WPOWRS,KWPOWR
      WRITE(6,9400)WOILRS,KWOILR
      WRITE(6,9500)WACRES,KWACRE
9650 WRITE(6,9700)WWET,KWWET
9750 WRITE(6,9800)WLOSS,KWLOSS
      WRITE(6,9898)WFRST,FROST
9850 WRITE(6,9900)WFULOS,KWFULO
9950 WRITE(6,10000)WOXLOS,KWOXLO
10050 WRITE(6,10100)WPOWFO,KWPOWF
10150 WRITE(6,10200)WACSEF,KWACSEF
      WRITE(6,10400)WOIL,KWOIL
      WRITE(6,10410)WABFU,KWABFU
      WRITE(6,10600)WDECA,KWDECA
10650 WRITE(6,10700)WFOOX,KWFOOX
10750 WRITE(6,10800)WFUL,KWFUL
      WRITE(6,11400)WOXID,KWOXID
      WRITE(6,11500)WFO2(1),KWFO2
      WRITE(6,11600)WOX2(1),KWOX2
      WRITE(6,11900)WTO,WTO
      WRITE(6,12300)WPREIG,KWPREI
      WRITE(6,12500)WGROSS,KWGROSS
      IF(JUMP.EQ.0) GO TO 9999
      WBOO = WGROSS * WPASS
      KWBOO = WBOO * KILO
      WRITE(6,13010)WBOO,KWBOO
9999 CONTINUE
      RETURN
100  FORMAT(1M,52HAERO DYNAMIC SURFACES
      *F18.0,F27.0 )
200  FORMAT(1M,52H      WING + WING MOUNTED CONTROL SURFACES
      *F9.0,F27.0)
400  FORMAT(1M,52H      VERTICAL SURFACES
      *F9.0,F27.0)
700  FORMAT(1M,52H      FAIRINGS,SHROUDS AND ASSOCIATED STRUCTURE
      *F9.0,F27.0)
800  FORMAT(1M,52HBODY STRUCTURE
      *F18.0,F27.0 )
900  FORMAT (1M,50H      STRUCTURAL FUEL CONTAINERS
      *F11.0,F27.0)
1000 FORMAT (1M,50H      STRUCTURAL OXIDIZER CONTAINERS
      *F11.0,F27.0)
1100 FORMAT(1M,52H      BASIC BODY STRUCTURE
      *F9.0,F27.0)
1300 FORMAT(1M,52H      SECONDARY STRUCTURE
      *F9.0,F27.0)
1400 FORMAT(1M,52H      THRUST STRUCTURE
      *F9.0,F27.0)
1600 FORMAT(1M,52HINDUCED ENVIRONMENTAL PROTECTION
      *F18.0,F27.0 )

```

1800	FORMAT(1H,52H	COVER PANELS, NON-STRUCTURAL
	*F9.0,F27.0)	
1900	FORMAT(1H,52H	VEHICLE INSULATION
	*F9.0,F27.0)	
2200	FORMAT(1H0,52H	LAUNCH, RECOVERY AND DOCKING
	*F18.0,F27.0)	
2300	FORMAT(1H,52H	LAUNCH GEAR
	*F9.0,F27.0)	
2400	FORMAT(1H,52H	DEPLOYABLE AERODYNAMIC DEVICES
	*F9.0,F27.0)	
2500	FORMAT(1H,52H	ALIGNING GEAR
	*F9.0,F27.0)	
2600	FORMAT(1H,52H	DOCKING STRUCTURE
	*F9.0,F27.0)	
2800	FORMAT(1H0,52H	PROPULSION
	*F18.0,F27.0)	
2870	FORMAT(1H,52H	SECONDARY ENGINES AND ACCESSORIES
	*F9.0,F27.0)	
2900	FORMAT(1H,52H	ENGINES AND ACCESSORIES
	*F9.0,F27.0)	
3100	FORMAT(1H,52H	NACELLES, PODS, PYLONS, SUPPORTS
	*F9.0,F27.0)	
3300	FORMAT(1H,52H	FUEL CONTAINERS AND SUPPORTS (NON-STRUCTURAL)
	*F9.0,F27.0)	
3400	FORMAT(1H,52H	OXIDIZER CONTAINERS AND SUPPORTS (NON-STRUCTURAL)
	*F9.0,F27.0)	
3410	FORMAT(1H,52H	SECONDARY FUEL TANKAGE AND SYSTEMS
	* F9.0,F27.0)	
3420	FORMAT(1H,52H	SECONDARY OXIDIZER TANKAGE AND SYSTEMS
	* F9.0,F27.0)	
3500	FORMAT(1H,52H	PROPELLANT INSULATION
	*F9.0,F27.0)	
3600	FORMAT(1H,52H	FUEL SYSTEM - MAIN
	*F9.0,F27.0)	
3700	FORMAT(1H,52H	OXIDIZER SYSTEM - MAIN
	*F9.0,F27.0)	
3800	FORMAT(1H,52H	PURGE SYSTEMS
	*F9.0,F27.0)	
4110	FORMAT(1H,52H	AIRBREATHING ENGINES AND INSTALLATION
	*F9.0,F27.0)	
4115	FORMAT(1H,52H	AIRBREATHING PROPULSION TANKAGE AND SYSTEMS
	*F9.0,F27.0)	
4300	FORMAT(1H0,52H	ORIENTATION, SEPARATION AND ULLAGE CONTROL
	*F18.0,F27.0)	
4400	FORMAT(1H,52H	SEPARATION SYSTEMS
	*F9.0,F27.0)	
4470	FORMAT(1H,52H	STABILITY AND CONTROL, ENGINE GIMBAL
	*F9.0,F27.0)	
4500	FORMAT(1H,52H	AERODYNAMIC CONTROLS
	*F9.0,F27.0)	
4600	FORMAT(1H,52H	SPATIAL ATTITUDE CONTROL SYSTEM
	*F9.0,F27.0)	

4800 FORMAT(1H,52H CONTROL PROPELLANT TANKAGE AND SYSTEMS
 *F9.0,F27.0)
 4900 FORMAT(1H0,52HPRIME POWER SOURCE AND DISTRIBUTION
 *F18.0,F27.0)
 5000 FORMAT(1H,52H POWER SOURCE UNITS
 *F9.0,F27.0)
 5300 FORMAT(1H,52H POWER SOURCE TANKAGE AND SYSTEMS
 *F9.0,F27.0)
 5400 FORMAT(1H0,52HPOWER CONVERSION AND DISTRIBUTION
 *F18.0,F27.0)
 5600 FORMAT(1H,52H HYDRAULIC/PNEUMATIC
 *F9.0,F27.0)
 5700 FORMAT(1H0,52HGUIDANCE AND NAVIGATION
 *F18.0,F27.0)
 5800 FORMAT(1H0,52HINSTRUMENTATION
 *F18.0,F27.0)
 5900 FORMAT(1H0,52HCOMMUNICATION
 *F18.0,F27.0)
 6600 FORMAT(1H0,52HPERSONNEL PROVISIONS
 *F18.0,F27.0)
 7300 FORMAT(1H0,52HORY STRUCTURE
 *, 7X,1H(,F10.0,1H),17X,1H(,F10.0,1H))
 7400 FORMAT(1H0,52HDESIGN RESERVE (CONTINGENCY)
 *F18.0,F27.0)
 7500 FORMAT(1H0,52HPERSONNEL
 * F18.0,F27.0)
 7800 FORMAT(1H0,52HPAYLOAD
 *F18.0,F27.0)
 7860 FORMAT(1H,52H MISSION EQUIPMENT/PAYLOAD
 *F9.0,F27.0)
 7900 FORMAT(1H,52H CARGO OR UPPER STAGE
 *F9.0,F27.0)
 8400 FORMAT(1H0,52HRESIDUAL PROPELLANT AND SERVICE ITEMS
 *F18.0,F27.0)
 8500 FORMAT(1H,52H TANK PRESSURIZATION AND PURGE GASES
 *F9.0,F27.0)
 8600 FORMAT(1H,52H TRAPPED FUEL
 *F9.0,F27.0)
 8700 FORMAT(1H,52H TRAPPED OXIDIZER
 *F9.0,F27.0)
 8900 FORMAT(1H,52H SERVICE ITEMS RESIDUALS
 *F9.0,F27.0)
 9000 FORMAT(1H0,52HRESERVE PROPELLANT AND SERVICE ITEMS
 *F18.0,F27.0)
 9100 FORMAT(1H,52H FUEL-MAIN PROPULSION
 *F9.0,F27.0)
 9200 FORMAT(1H,52H OXIDIZER-MAIN PROPULSION
 *F9.0,F27.0)
 9300 FORMAT(1H,52H POWER SOURCE PROPELLANTS
 *F9.0,F27.0)
 9400 FORMAT(1H,52H LUBRICANTS
 *F9.0,F27.0)

9500 FORMAT(1H ,52H ATTITUDE CONTROL PROPELLANTS
 *F9.0,F27.0)
 9700 FORMAT(1H0,52HOPERATING WEIGHT EMPTY
 *, 7X,1H(,F10.0,1H),17X,1H(,F10.0,1H))
 9800 FORMAT(1H0,52HIN-FLIGHT LOSSES
 *F18.0,F27.0)
 9898 FORMAT(1H ,52H ICE AND FROST
 *F9.0,F27.0)
 9900 FORMAT(1H ,52H FUEL VENTED
 *F9.0,F27.0)
 10000 FORMAT(1H ,52H OXIDIZER VENTED
 *F9.0,F27.0)
 10100 FORMAT(1H ,52H POWER SOURCE PROPELLANTS
 *F9.0,F27.0)
 10200 FORMAT(1H ,52H ATTITUDE CONTROL PROPELLANTS
 *F9.0,F27.0)
 10400 FORMAT(1H ,52H LUBRICANTS AND OTHER SERVICE ITEMS
 *F9.0,F27.0)
 10410 FORMAT(1H ,52H FLYBACK FUEL
 *F9.0,F27.0)
 10600 FORMAT(1H0,52HTHRUST DECAY PROPELLANTS
 *F18.0,F27.0)
 10700 FORMAT(1H0,52HPROPELLANTS
 *F18.0,F27.0)
 10800 FORMAT(1H ,52H FUEL - MAIN
 *F9.0,F27.0)
 11400 FORMAT(1H ,52H OXIDIZER - MAIN
 *F9.0,F27.0)
 11500 FORMAT(1H ,52H FUEL - SECONDARY
 *F9.0,F27.0)
 11600 FORMAT(1H ,52H OXIDIZER - SECONDARY
 *F9.0,F27.0)
 11900 FORMAT(1H0,52HMASS AT FULL THRUST
 *F18.0,F27.0)
 12300 FORMAT(1H0,52HPRE-IGNITION LOSSES
 *F18.0,F27.0)
 12500 FORMAT(1H0,52HMAXIMUM GROSS WEIGHT
 *, 7X,1H(,F10.0,1H),15X,1H(,F10.0,1H))
 13010 FORMAT(1H0,50HGROSS WEIGHT OF ONE BOOSTER
 * 2XF18.0,F27.0)
 13500 FORMAT(1H ,52H HORIZONTAL SURFACES
 * F9.0,F27.0)
 END

```

SUBROUTINE PRINTV
C *** PRINTV ***
C
C PRINTV - PRINT VOLUMES
C
REAL KILO,KVSTRU,KVCARG,KVLGRA,KVPROP,KVFUTK,KVOXTK,KVINST,KVOTHE
1,KVBODY,KVFUT2,KVCXT2,KVCREW,ISP,K,LF,MR,NCREW,LBODY,NPASS
REAL NENG$
COMMON/CINPUT/
1ANENG$ ,ANTANK ,ASRATO ,ASWEEP ,C(300) ,CBBODY ,CFUEL(6),
2CHBODY ,CLBODY ,CSBODY ,CSFAIR ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN ,CSVERT ,CSWING ,CTHRST ,CTHST2 ,DEF(5) ,FXHOVS ,
4ISP(6) ,ITPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5NENG$ ,NLISTO ,NPASS ,NWL ,PCHAM ,Q ,RHOFU ,
6RHOFU2 ,RHOX ,RHOX2 ,SBODY ,TOL ,TOVERC ,TPRATO ,
7TYTAIL ,VBODY ,WGROSS
COMMON/VOLCAL/BODY,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HBODY ,LBODY ,RTOD ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPLAN ,STPS(1) ,SVERT ,SWING ,SXPOS ,TDEL ,TROOT ,
4TTOT ,TTOT2 ,TTOTAL ,VBOYA ,VBOY1 ,VBOY2 ,VCARGO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGRAY ,VOTHER ,VOXTK ,
6VOXTK2 ,VPROP ,VSTRUC
COMMON/WTALC/
1WACS ,WACSEO ,WACSTK ,WABFTK ,WABFU ,WABPR ,WACRES ,
2WBPUMP ,WCARGO ,WCOMM ,WAERO ,WAUXT ,WABASIC ,WBODY ,
3WDIST2 ,WDOCK ,WDPLOY ,WDRANS ,WDRY ,WDECAY ,WDIST1 ,
4WENGMT ,WENG$ ,WENG$2 ,WFAIR ,WFCONT ,WEDCAY ,WEMPTY ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFUOX ,WFURES ,
6WFUSYS ,WFUTK ,WFUTK2 ,WFUTOT ,WFUTRP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLOSS ,WLRD ,WNACEL ,WODCAY ,
9WOIL ,WOILPS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,WOXID ,
1WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPERS ,WPOWCD ,WPOWER ,WPOWFO ,
3WPOHRS ,WPOWTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
4WRESID ,WRESRV ,WSFAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF ,WTABC ,WTHRST ,WTO ,WTPS ,WVERT ,WWAIT(10),
6WWET ,WWING ,WZROFU
9 KILO = .0283
KVSTRU =KILO *VSTRUC
KVCREW=KILO *VCREW
KVOXT2=KILO*VOXTK2
KVFUT2=KILO*VFUTK2
KVCARG =KILO *VCARGO
KVLGRA =KILO *VLGRAY
KVPROP =KILO *VPROP
KVFUTK =KILO *VFUTK
KVOXTK =KILO *VOXTK
KVINST = VINSTK*KILO
KVOTHE =KILO *VOTHER
KVBODY =KILO *VBODY
WRITE (6,8)

```

```

CALL DATE
WRITE(6,10)
WRITE(6,100) VSTRU,KVSTRU
WRITE(6,200) VCREW,KVCREW
WRITE(6,300) VCARGO,KVCARGO
WRITE(6,400) VLGBAY,KVLGBA
WRITE(6,500) VPROP,KVPROP
WRITE(6,700) VFUTK,KVFUTK
WRITE(6,800) VOXTK,KVOXTK
WRITE(6,900) VINSTK,KVINST
WRITE(6,1400) VFUTK2,KVFUT2
WRITE(6,1500) VOXTK2,KVOXT2
1110 FORMAT(1H0,40HTOTAL ENTRY VEHICLE VOLUME          F9.1,F10.1)
950  WRITE(6,1000) VOTHEP,KVOTHE
      WRITE(6,1110) VBODY,KVBODY
      IF(VBODY.EQ.0.) VBODY = 1.
      CALL PROTHR
      RETURN
8  FORMAT (1H1)
10  FORMAT( 8H VOLUMES ,35X,10H CU FT      ,10H CU M      /  )
100 FORMAT(1H0,40HBODY STRUCTURE          2F10.2)
200 FORMAT(1H ,40HCREW AND PASSENGER COMPARTMENTS 2F10.2)
300 FORMAT(1H ,40HCARGO COMPARTMENT           2F10.2)
400 FORMAT(1H ,40HLANDING GEAR BAYS          2F10.2)
500 FORMAT(1H ,40HPPROPULSION BAY WITHIN BODY 2F10.2)
700  FORMAT(1H ,40HFUEL CONTAINERS            2F10.2)
800  FORMAT(1H ,40HOXIDIZER CONTAINERS        2F10.2)
900  FORMAT(1H ,40HPROPELLANT INSULATION      2F10.2)
1000 FORMAT(1H ,40HOTHER BODY VOLUME         2F10.2)
1400 FORMAT(1H ,40HSECONDARY FUEL CONTAINERS 2F10.2)
1500 FORMAT(1H ,40HSECONDARY OXIDIZER CONTAINERS 2F10.2)
END

```

Appendix IV-39 Subroutine PROTHR

```

C *** SUBROUTINE PROTHR ***
C
C PROTHR - PRINT ALL OTHER QUANTITIES
C
REAL ISP,K,LF,MP,NCREW,LBODY,NPASS
REAL NENG5
REAL KILO, KSBODY, KSWING, KNOVER, KSVERT, KSHORZ, KUWWIN,
1 KUWVER, KUWHOP, KUWBOD, KLROOT, KLBODY, KRBODY, KMBODY,
2 KSXPOS, KSPLAN
REAL KCSPAN,KCROOT
COMMON/CYINPUT/
1ANENG5 ,ANTANK ,ASRATO ,ASWEEP ,C(300) ,CBBODY ,CFUEL(6),
2CHBODY ,CLBODY ,CSBODY ,CSFAIR ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN ,CSVERT ,CSWING ,CTHRST ,CTHST2 ,DEF(5) ,FXMOVS ,
4ISP(6) ,ITPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5NENG5 ,NLIST0 ,NPASS ,NWL ,PCHAM ,Q ,RHOFU ,
6RHOFU2 ,RHOX ,RHOX2 ,SBODY ,TOL ,TOVERC ,TPRATO ,
7TYTAIL ,VRODY ,WGR0SS
COMMON/VOLCAL/BRODY,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HRODY ,LBODY ,RT00 ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPLAN ,STPS(1) ,SVERT ,SWING ,SXPOS ,TDEL ,TROOT ,
4TTOT ,TTOT2 ,TTOTAL ,VRODYA ,VRODY1 ,VRODY2 ,VCARGO ,
5VCREW ,VFUTK ,VFUTK2 ,VINSTK ,VLGRAY ,VOTHER ,VOXTK ,
6VOXTK2 ,VPROP ,VSTRUC
COMMON/WTALC/ ABFSYS ,WABFTK ,WABFU ,WABPR ,WACRES ,
1WACS ,WACSF0 ,WACSTK ,WAERO ,WAUXT ,WBASIC ,WBODY ,
2WBUMP ,WCARGO ,WCOMM ,WCONT ,WCOVER ,WDECAY ,WDIST1 ,
3WDIST2 ,WDOCK ,WDPOY ,WDRANS ,WDRY ,WELCAD ,WEMPTY ,
4WENGMT ,WENG5 ,WENG52 ,WFAIR ,WFCONT ,WFDCAY ,WFROST ,
5WFU2(3) ,WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFUOX ,WFURES ,
6WFUSYS ,WFUTK ,WFUTK2 ,WFUTOT ,WFUTPP ,WGASPR ,WGNV ,
7WHORZ ,WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) ,WLANCH ,WLG ,WLOSS ,WLRD ,WNACEL ,WODCAY ,
9WOIL ,WOILRS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,WOXID ,
1WOXLOS ,WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP ,WPASS ,WPAYL ,WPERS ,WPOMCO ,WPOWER ,WPOWFO ,
3WPOWRS ,WPOWTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
4WRESID ,WRESRV ,WSEAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF ,WTABC ,WTHRST ,WTO ,WTPS ,WVERT ,WWAIT(10),
6WNET ,WWING ,WZROFU
UWING = ).
UWVERT=0.0
UWHORZ=0.0
IF(SBODY.NE.0.0) UWBODY=(WBASIC+WSECST+WINFUT+WINOXT)/SBODY
IF(SWING.NE.0.) UWING = WING / SWING
IF(SVERT.NE.0.) UWVERT = WVERT / SVERT
IF(SHORZ.NE.0.) UWHORZ = WHORZ / SHORZ
KILO = .0929
SFUTKK = KILO * SFUTK
SOXTKK = KILO * SOXTK
KSBODY = SBODY *KILO

```



```

KSWING = SWING *KILO
KSXPOS=KILO*SXPOS
KSPLAN =KILO*SPLAN
KSVERT = SVERT *KILO
KSHORZ = SHORZ *KILO
KILO=4.8824
KUWIN = KILO* UWIN
KUVER = KILO* UVERT
KUWHOR = KILO* UWHORZ
KUWBOO = KILO* UWBOO
KWOVER = MOVERS*KILO
KILO = .0929**5
KBOOY = KILO*BOOY
KLROOY = KILO*CRROOY
KCSPAN=SPAN*KILO
KCROOY=CRROOY*KILO
KHOY = KILO*HOY
WRITE(6,40)
50 WRITE(6,100)
150 WRITE(6,200) SBOOY ,KSBOOY
    WRITE(6,620) SFUTK,SFUTKK
    WRITE(6,625) SOXTK,SOXTKK
650 WRITE(6,700)
    WRITE(6,999) SWING,KSWING
    WRITE(6,3000) SXPOS,KSXPOS
    WRITE(6,9000) SPAN,KSPLAN
950 WRITE(6,1000)SVERT ,KSVERT
1050 WRITE(6,1100)SHORZ ,KSHORZ
1250 WRITE(6,1300)
1350 WRITE(6,1400)UWIN,KUWIN
1450 WRITE(6,1500)UVERT,KUVER
1550 WRITE(6,1600)UWHORZ,KUWHOR
1650 IF(WINFUT.EQ.0..AND.WINOXT.EQ.0.)WRITE (6,1700) UWBOOY, KUWBOO
850 WRITE(6,910) MOVERS ,KWOVER
3250 WRITE(6,2700)
    WRITE(6,2500)
    WRITE(6,2000) SPAN,KCSPAN
    WRITE(6,2100) CRROOY,KCROOY
    WRITE(6,3100) TOVERC
    WRITE(6,2600)
7270 WRITE(6,2200)LBOOY,KLBOOY
    WRITE(6,2300) BOOY,KBOOY
    WRITE(6,2400) HOY,KHOY
    RETURN
40 FORMAT (/// 15X11HDESIGN DATA )
100 FORMAT(1H0,12HWETTED AREAS 29X,10H SQ. FT. ,10H SQ.M.
*)
200 FORMAT(1H0,40H GROSS BODY (SBOOY) 2F10.2)
620 FORMAT (1H ,18HFUEL TANKS (SFUTK), 22X, 2F10.2)
625 FORMAT (1H ,22HOXIDIZER TANKS (SOXTK), 18X, 2F10.2)
700 FORMAT(1H0,12HPLAN AREAS 29X,10H SQ. FT. ,10H SQ.M. )

```

```

999 FORMAT(1M ,40H THEORETICAL WING AREA (SWING) 2F10.2)
910 FORMAT(1M ,40H WING LOADING ,2F10.2)
1500 FORMAT(1M ,40H VERTICAL SURFACES 2F10.2 )
1100 FORMAT(1M ,40H HORIZONTAL SURFACES (SPORZ) 2F10.2 )
1300 FORMAT(1M0,12HUNIT WEIGHTS 28X,10H LB/SQ.FT. ,10H KG/SQ.M. )
1400 FORMAT(1M0,40H WING 2F10.2 )
1000 FORMAT(1M ,40H VERTICAL SURFACES (SVERT) 2F10.2 )
1600 FORMAT(1M ,40H HORIZONTAL SURFACES 2F10.2 )
1700 FORMAT(1M ,40H BODY STRUCTURE (BASIC) 2F10.2 )
2000 FORMAT(1M ,40H STRUCTURAL SPAN 2F10.2)
2100 FORMAT(1M ,40H ROOT CHORD LENGTH 2F10.2)
2200 FORMAT(1M ,40H LENGTH (LBODY) 2F10.2)
2300 FORMAT(1M ,40H WIDTH (BBODY) 2F10.2)
2400 FORMAT(1M ,40H HEIGHT (HBODY) 2F10.2)
2500 FORMAT(1M ,40H WING )
2600 FORMAT(1M ,40H BODY )
2700 FORMAT(1M0,16HDIMENSIONAL DATA,27X,10H FEET ,10H METERS )
3000 FORMAT(1M ,40H EXPOSED WING AREA (SXPOS) 2F10.2)
3100 FORMAT(1M ,30H THICKNESS RATIO 2F10.2)
9000 FORMAT(1M ,41H PLANFORM, INCL. ELEVONS (SPAN)
* F9.2,F10.2)
END

```

PRECEDING PAGE BLANK NOT FILMED

GDC-08870-002

Appendix IV-40 Subroutine PRWTSM

```

SUBROUTINE PRWTSM
C *** PRWTSM ***
C
C PRWTSM - PRINT WEIGHT SUMMARY
C
REAL ISP,K,LF,MR,NCREW,LBODY,NPASS
REAL NENG$
REAL KWABFU
REAL      KDELTV
REAL KWJET(6),KWFUEL(6),KWDX(6)
*      ,KWPRIG,KWGROS
REAL KILO,KHWAIT
DIMENSION KDELTV(6)
DIMENSION DELTAV(6)
DIMENSION OF(6)
DIMENSION KHWAIT(10)
COMMON / JUMPY / JUMP, WBI6, WBOO
COMMON/CENPUT/
1ANENG$  ,ANTANK  ,ASRATO  $ASWEEP  ,C(300)  ,CBBODY  ,CFUEL(6),
2CHBODY  ,CLBODY  ,CSBODY  $CSFAIR  ,CSFUTK  ,CSHORZ  ,CSOXTK  ,
3CSPLAN  ,CSVERT  ,CSWING  $CTHRST  ,CTHST2  ,DEF(5)  ,FXWOVS  ,
4ISP(6)  ,ITPS    ,K(30)   $KIN     ,LF      ,MR(6)   ,NCREW   ,
5NENG$   ,NLISTO  ,NPASS   $NWL     ,PCHAM   ,QMX     ,RHOFU   ,
6RHOFU2  ,RHOX    ,RHOX2   $SBODY   ,TOL     ,TOVERC  ,TPRATO  ,
7TYTAIL  ,VBODY   ,WGROSS
COMMON/VOLCAL/BBODY,CROOT $CSPAN   ,CTIP    ,GAL     ,GSPAN   ,
2HBODY   ,LBODY   ,RTOD    $SFAIR   ,SFUTK   ,SHORZ   ,SOXTK   ,
3SPLAN   ,STPS(1) ,SVERT   $SWING   ,SXPOS   ,TDEL    ,TROOT   ,
4TTOT    ,TTOT2   ,TTOTAL  $VBOYA   ,VBOY1    ,VBOY2    ,VCARGO  ,
5VCREW   ,VFUTK   ,VFUTK2  $VINSTK  ,VLGBAY  ,VOTHER  ,VOXTK   ,
6VOXTK2  ,VPROP   ,VSTRUC
COMMON/WCALC/  ABFSYS  $WABFTK  ,WABFU   ,WABPR   ,WACRES  ,
1WACS    ,WACSF0  ,WACSTK  ,WAERO   ,WAUXT   ,WBASIC  ,WBODY   ,
2WBUMP   ,WCARGO  ,WCONM   $WCONT   ,WCOVER  ,WDECAY  ,WDIST1  ,
3WDIST2  ,WDOCK   ,WDPLOY  $WDRANS  ,WDY     ,WELCAD  ,WEMPTY  ,
4WENGHT  ,WENG$   ,WENG$2  $WFAIR   ,WFCONT  ,WFOCAY  ,WFROST  ,
5WFU2(3) ,WFOEL(6),WFUL    $WFULOS  ,WFUNCT  ,WFOOX   ,WFURES  ,
6WFUSYS  ,WFUTK   ,WFUTK2  $WFUTOT  ,WFUTRP  ,WGASPR  ,WGNVAV  ,
7WHORZ   ,WHYCAD  ,WINFUT  $WINOXT  ,WINSTK  ,WINST   ,WINSUL  ,
8WJET(6) ,WLANCH  ,WLG     $WLOSS   ,WLRD    ,WNACEL  ,WODCAY  ,
9WOIL    ,WOILRS  ,WORSUL  $WOVERS  ,WOX(6)  ,WOX2(3) ,WOXID   ,
1WOXLOS  ,WOXRES  ,WOXSYS  $WOXTK   ,WOXTK2  ,WOXTOT  ,WOXTRP  ,
2WP      ,WPASS   ,WPAYL   $WPERS   ,WPOWCO  ,WPOWER  ,WPOWFO  ,
3WPOHRS  ,WPOWTK  ,WPPROV  $WPREIG  ,WPROP   ,WPRSYS  ,WREFUL  ,
4WRESID  ,WRESRV  ,WSEAL   $WSEOST  ,WSORCE  ,WSRTRP  ,WSTAB   ,
5WSURF   ,WTAOC   ,WTHRST  $WTO     ,WTPS    ,WVERT   ,HWAIT(10),
6HWET    ,HWING   ,WZROFU
COMMON/ENS/WORBIT,WETURN,WENTRY,WLAND
WORBIT = 0.
WLAND = 0.
WENTRY = 0.
WETURN = 0.

```

```

KILO = 0.4535924
KI = 6
IF(WPREIG.NE.0.) WPREIG=-WPREIG
5 DO 111 I = 1, KI
  IF(CFUEL(I).EQ.0.0) CFUEL(I)=1.
  OF(I) = (1-CFUEL(I)) / CPUEL(I)
  IF(WJET(I).NE.0.) WJET(I)=-WJET(I)
  IF(WFUEL(I).NE.0.) WFUEL(I)=-WFUEL(I)
  IF(WOX(I).NE.0.) WOX(I)=-WOX(I)
  KWJET(I)=WJET(I)*KILO
  KWFUEL(I)=KILO*WFUEL(I)
  KWOX(I)=KILO*WOX(I)
  DELTAV(I)=32.174*ISP(I)*ALOG(MR(I))
  KDELTV(I)=DELTAV(I)/3.281
111 CONTINUE
  KWPRIG=KILO*WPREIG
  KWGROS=KILO*WGROSS
  DO 10 I=1,10
10    KWWAIT(I) = KILO* WWAIT(I)
    DO 20 J = 1,6
    IF(MR(J) .NE. 1.) GO TO 25
20    CONTINUE
    GO TO 50
25    CONTINUE
    WRITE (6,664)
    CALL DATE
    WRITE(6,665)
    WRITE (6,666) (I,MR(I),OF(I),ISP(I),DELTAV(I),KDELTV(I), I=1,6)
50    WRITE(6,100)
    WRITE(6,240)
150   WRITE(6,200) WGROSS,KWGROS
    WRITE(6,205) WPREIG,KWPRIG
    WRITE(6,207) WWAIT(1),KWWAIT(1)
160   WRITE(6,210) WFUEL(1),KWFUEL(1)
    WRITE(6,220) WOX (1),KWOX (1)
    WRITE(6,230) WJET(2),KWJET(1)
    WRITE(6,300) WWAIT(2),KWWAIT(2)
    WENTRY = WWAIT(6)
    JUMP = JUMP + 1
    GO TO (260,280), JUMP
260   WRITE (6,210) WFUEL(3), KWFUEL(3)
    WRITE (6,220) WOX(3), KWOX(3)
    WRITE(6,230) WJET(3),KWJET(3)
    WRITE (6,401) WWAIT(4), KWWAIT(4)
    WORBIT = WWAIT(4)
    WRITE(6,230) WJET(4),KWJET(4)
    WRITE (6,500) WWAIT(5), KWWAIT(5)
    WETURN = WWAIT(5)
    WRITE (6,210) WFUEL(5), KWFUEL(5)
    WRITE (6,220) WOX(5), KWOX(5)
    WRITE(6,230) WJET(5),KWJET(5)
    WRITE (6,567) WWAIT(6), KWWAIT(6)

```

```

JUMP = JUMP - 1
IF(JUMP.EQ.0) GO TO 765
250 WRITE (6,210) WFUEL(3), KWUEL(3)
WRITE (6,220) WOX(3), KWOX(3)
WRITE(6,237) WJET(3),KWJET(3)
WRITE (6,1000) WAIT(4), KWAIT(4)
WETURN = WAIT(4)
WRITE (6,1005) WJET(4), KWJET(4)
WRITE (6,500) WAIT(5), KWAIT(5)
WRITE(6,2010) WJET(5),KWJET(5)
WRITE (6,1015) WAIT(6), KWAIT(6)
JUMP=JUMP-1
765 KWABFU = WABFU * KILO
WABFU = - WABFU
KWABFU = - KWABFU
WRITE (6,225) WABFU, KWABFU
WAIT = WAIT(7)
WLAND = WAIT
WABFU = - WABFU
WAIT2 = KWAIT(7)
WRITE (6,800) WAIT, WAIT2
IF(WPREIG.NE.0.) WPREIG=-WPREIG
DO 120 I=1,KI
IF(WJET(I).NE.0.) WJET(I)=-WJET(I)
IF(WFUEL(I).NE.0.) WFUEL(I)=-WFUEL(I)
IF(WOX(I).NE.0.) WOX(I)=-WOX(I)
120 CONTINUE
IF(WJET(6).EQ.WLOSS) WJET(6) = 0.
CALL PRIVEQ
RETURN
100 FORMAT(/ / 15H WEIGHT SUMMARY ,20X,10H POUNDS ,10H KILOGRAMS )
200 FORMAT(1W0,40H MAXIMUM GROSS WEIGHT (WGROSS) 2F10.0)
205 FORMAT(1W0,40H PRE-IGNITION LOSSES 2F10.0)
207 FORMAT(1W0,40H MASS AT IGNITION 2F10.0)
210 FORMAT(1W0,40H FUEL 2F10.0)
220 FORMAT(1W ,40H OXIDIZER 2F10.0)
225 FORMAT (2W0,40H FLYBACK FUEL 2F10.0)
230 FORMAT(1W ,40H JETTISON 2F10.0)
237 FORMAT (2W ,40H PAYLOAD ADDED/REMOVED 2F10.0)
240 FORMAT(1W0,2H )
300 FORMAT(1W0,40H END OF PRE-FLIGHT PHASE (WTO) 2F10.0)
401 FORMAT (2W0,40H WEIGHT IN ORBIT (WORBIT) 2F10.0)
500 FORMAT(1W0,40H RETURN CONDITION (WETURN) 2F10.0)
567 FORMAT(1W0,40H ENTRY WEIGHT (WENTRY) 2F10.0)
664 FORMAT (PH1)
665 FORMAT(69X,31H THEORETICAL VELOCITY INCREMENT / 51X
* 14H0/F RATIO ISP,5X,20H FEET/SECOND METERS/SECOND )
666 FORMAT(1W0,31H MASS RATIO DURING FLIGHT PHASE ,I2,1W=F10.4,
* F12.4/3XF6.2,2XF10.1,5XF10.1)
800 FORMAT(1W0,40H LANDING WEIGHT (WLAND) 2F10.0 )
1000 FORMAT (2W0,40H INITIAL SEPARATION 2F10.0)
1005 FORMAT (2W ,40H THRUST DECAY AND RESIDUALS 2F10.0)

```

GDC-08870-002

1810 FORMAT (1H ,40H NETTISON AND EXPENDABLES
1015 FORMAT (1H0,40HINIIVIAL FEYBACK
END

2F10.0)
2F10.0)

Appendix IV-41 Subroutine PRITEC

```

      SUBROUTINE PRI/EQ
C *** PRITEQ ***
C
C
      REAL ISP,K,LF,MR,NCREW,LBODY,NPASS
      REAL NENGS
      COMMON NSEC,T,TR,TS(15),V(85),QX(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),HQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0-XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PTF(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
      COMMON/CINPUT/
1ANENGs  /ANTANK  ,ASRATO  /ASWEEP  ,C(3009  ,CBBODY  ,CFUEL(6),
2CHBODY  /CLBODY  ,CSBODY  /CSFAIR  ,CSFUTK  ,CSHORZ  ,CSOXTK  ,
3CSPLAN  /CSVERT  ,CSWING  /CTHRST  ,CTHST2  ,DEF(5)  ,FXMOVS  ,
4ISP(6)  /ITPS    ,K(30)   /KIN     ,LF      ,MR(6)   ,NCREW  ,
5NENGs   /NLISTO  ,NPASS   /NWL     ,PCHAN   ,Q       ,RHOFU   ,
6RHOFU2  /RHOX    ,RHOX2   /SBOOY   ,TOL     ,TOVERC  ,TPRATO  ,
7TYTAIL  /VBOOY   ,WGROSS
      COMMON/VOLCAL/BBODY,CROOT /CSPAN  ,CTIP    ,GAL     ,GSPAN  ,
2HBODY   /LBODY   ,RTOD    /SFAIR   ,SFUTK   ,SHORZ   ,SOXTK   ,
3SPAN    /STPS(1) ,SVERT   /SWING   ,SXPOS   ,TDEL    ,TROOT   ,
4TTOT    /TTOT2   ,TTOTAL  /VBOOYA  ,VBOOY1  ,VBOOY2  ,VCARGO  ,
5VCREW   /VFUTK   ,VFUTK2  /VINSTK  ,VLGBAY  ,VOTHER  ,VOXTK   ,
6VOXTK2  /VPROP   ,VSTRUC
      COMMON/WTCALC/  ABFSYS  /WABFTK  ,WABFU   ,WABPR   ,WACRES  ,
1WACS    /WACSF0  ,WACSTK  /WAERO   ,WALRT   ,WBASIC  ,WBODY   ,
2WBUMP   /WCARGO  ,WCOMH   /WCONT   ,WCOVER  ,WDECAY  ,WDIST1  ,
3WDIST2  /WDOCK   ,WDPOY   /WDRANS  ,WDY     ,WELCAD  ,WEMPTY  ,
4WENGHT  /WENGs   ,WENGs2  /WFAIR   ,WFCONT  ,WFDCAY  ,WFROST  ,
5WFU2(3) /WFUEL(6),WFJL    /WFULOS  ,WFUNCT  ,WFUOX   ,WFURES  ,
6WFUSYS  /WFUTK   ,WFUTK2  /WFUTOT  ,WFUTRP  ,WGASPR  ,WGNV   ,
7WHORZ   /WHYCAD  ,WINFUT  /WINOXT  ,WINSTK  ,WINST  ,WINSUL  ,
8WJET(6) /WLANCH  ,WLG     /WLOSS   ,WLRD   ,WNACEL  ,WNOOY   ,
9WOIL    /WOILRS  ,WORSUL  /WOVERS  ,WOX(6) ,WOX2(3),WOXID  ,
1WOXLOS  /WOXRES  ,WOXSYS  /WOXTK   ,WOXTK2 ,WOXTOT  ,WOXTRP  ,
2WP      /WPASS   ,WPAYL   /WPERS   ,WPOWCD  ,WPOWER  ,WPOWFO  ,
3WPOWRS  /WPONTR  ,WPPROV  /WPREIS  ,WPROP   ,WPRSYS  ,WREFUL  ,
4WRESID  /WRESFV  ,WSEAL   /WSECST  ,WSORCE  ,WSRTRP  ,WSTAB   ,
5WSURF   /WABC    ,WTHRST  /WTO     ,WTPS    ,WVERT   ,WMAIT(10),
6WNET    /WNING   ,WZROFU
      900 WRITE(6,2150)
      1150 FORMAT(1M1,28HWEIGHT COEFFICIENTS = C(I), /)
      DO 300 I = 1,300
      IF(C(I).NE.0.) WRITE (6,1200) I,C(I)
      300 CONTINUE
      1200 FORMAT(I5,F13.5)
      WRITE(6,2250)
      1250 FORMAT(1M8,38HVOLUME COEFFICIENTS = K(I), I = 1,30 =/)
      WRITE(6,2300) (I,K(I),I=1,30)
      1300 FORMAT( 5(I5,F13.5))

```

GDC-D8870-002

CALL PRITVA
RETURN
END

Appendix IV-42 Subroutine PRITVA

```

SUBROUTINE PRITVA
C *** PRITVA ***
C
C PRITVA - PRINTS VARIABLES
C
REAL ISP,K,LF,MR,NCREW,LBODY,NPASS
REAL NENBS
COMMON / JUMPY / JUMP, WBI8, W800
COMMON NSEC,T,TR,TS(15),V(85),Q(73,15),W(50),Z(120),EC(3),CV(3),
1 VQ(38,4),WQ(13,4),XM1(1,1),A1(1,1),C1(1,1,1),XM2(25,5),A2(12,5),
2 C2(25,12,5),XM3(25,5),A3(12,5),C3(25,12,5),XNA0,XNA1(1,1),
3 XNA2(2,5),XNA3(2,5),XNP(2),PST(25),TWD7(25,2),TWD8(25,2),SW(50),
4 TWD9(25,2),PT(25,4),PDT(25,4),XNPT(4),SV(50),SQ(40,5),SE(40),
5 PS(15)
COMMON/CENPUT/
1ANENG8 /ANTANK ,ASRATO ,ASWEEP ,C(300) ,CBBODY ,CFUEL(6),
2CHBODY /CLBODY ,CSBODY ,CSFAIR ,CSFUTK ,CSHORZ ,CSOXTK ,
3CSPLAN /CSVERT ,CSHING ,CTHRST ,CTHST2 ,DEF(5) ,FXWOVS ,
4ISP(6) /ITPS ,K(30) ,KIN ,LF ,MR(6) ,NCREW ,
5ENG8 /NLIST0 ,NPASS ,NWL ,PCHAM ,QMX ,RHOFU ,
6RHOFU2 /RHOX ,RHOX2 ,SBODY ,TOL ,TOVERC ,TPRATO
7TTYTAIL /VBODY ,WGROSS
COMMON/VOLCAL/BBODY,CROOT ,CSPAN ,CTIP ,GAL ,GSPAN ,
2HBODY /LBODY ,RTOD ,SFAIR ,SFUTK ,SHORZ ,SOXTK ,
3SPLAN /STPS(1) ,SVERT ,SHING ,SXPOS ,TOEL ,TROOT ,
4TTOT /TTOT2 ,TTOTAL ,VBODYA ,VBODY1 ,VBODY2 ,VCARGO ,
5VCREW /VFUTK ,VFUTK2 ,VINSTK ,VLGBAY ,VOTHER ,VOXTK ,
6VOXTK2 /VPROP ,VSTRUC
COMMON/WTCALC/ ABFSYS ,WABFTK ,WABFU ,WABPR ,WACRES ,
1WACS /WACSF0 ,WACSTK ,WAERO ,WAUXT ,WBASIC ,WBODY ,
2WBUMP /WCARGO ,WCOMM ,WCONT ,WCOVER ,WDECAY ,WDIST1 ,
3WDIST2 /WDOCK ,WDPOY ,WDRANS ,WDRY ,WELCAD ,WEMPTY ,
4WENGMT /WENG8 ,WENG82 ,WFAIR ,WFCONT ,WFDCAY ,WFROST ,
5WFU2(3) /WFUEL(6) ,WFUL ,WFULOS ,WFUNCT ,WFUOX ,WFURES ,
6WFUSYS /WFUTK ,WFUTK2 ,WFUTOT ,WFUTRP ,WGASPR ,WGNV ,
7WHORZ /WHYCAD ,WINFUT ,WINOXT ,WINSTK ,WINST ,WINSUL ,
8WJET(6) /WLANCH ,WLG ,WLOSS ,WLRO ,WNACEL ,WODCAY ,
9WOIL /WOILRS ,WORSUL ,WOVERS ,WOX(6) ,WOX2(3) ,WOXID ,
1WOXLOS /WOXRES ,WOXSYS ,WOXTK ,WOXTK2 ,WOXTOT ,WOXTRP ,
2WP /WPASS ,WPAYL ,WPERS ,WPOWCO ,WPOWER ,WPOWFO ,
3WPOWRS /WPOWTK ,WPPROV ,WPREIG ,WPROP ,WPRSYS ,WREFUL ,
4WRESID /WRESRV ,WSEAL ,WSECST ,WSORCE ,WSRTRP ,WSTAB ,
5WSURF /WABC ,WTHRST ,WTO ,WTPS ,WVERT ,WWAIT(10),
6WNET /WING ,WZROFU
WRITE (6,26001)
CALL DATE
WRITE (6,1000) CBBODY
WRITE (6,2000) CHBODY
WRITE (6,3000) CLBODY
WRITE (6,4000) CSBODY
WRITE (6,6000) CSFAIR
WRITE (6,7000) CSFUTK

```

```

WRITE( 6, 8000) CSWORZ
WRITE( 6, 9000) CSOXTK
WRITE( 6,10000) CSPLAN
WRITE( 6,11000) CSVERT
WRITE( 6,13000) CTWRST
WRITE(6,13002) CTHST2
WRITE(6,16000) NCREW
WRITE(6,17000) NENG$
WRITE(6,20000) RHOFU
WRITE(6,21000) RHOFU2
WRITE(6,22000) RHOX
WRITE(6,22010) RHOX2
WRITE(6,24000) CTHRST,WWAIT(3),C(129),NENG$,TTOT
WRITE(6,25000) CTHST2,WWAIT(6),C(158),TTOT2
100 WRITE( 6,26001)
WRITE( 6,26000) (I,WJET(I),I=1,6)
IF(JUMP.EQ.0) GO TO 200
WRITE( 6,1500) SW(I2)
WRITE( 6,1505) SW(I4)
WRITE( 6,1510) SW(6)
WRITE( 6,1515) SW(I1)
WRITE( 6,1520) SW(I5)
WRITE(6,1525) SQ(10,2)
WRITE(6,30000) SE(9)
WRITE(6,30001) SQ(5,1)
WRITE(6,30002) SQ(6,1)
WRITE(6,30003) SQ(9,1)
WRITE(6,30004) SQ(7,1)
WRITE(6,30005) SQ(8,1)
IF(JUMP.EQ.1) WRITE( 6,999)
200 RETURN
999 FORMAT (1H1)
1000 FORMAT (1H0,41HCBODY = WIDTH OF BODY / VBODY ** .3333 =,F9.4)
1500 FORMAT (28H0SPECIFIC FUEL CONSUMPTION = F7.5)
1505 FORMAT (28H CRUISE BACK VELOCITY = F9.2,8H FT/SEC )
1510 FORMAT (28H THRUST RATIO = F8.5)
1515 FORMAT (28H L/D = F8.5)
1520 FORMAT (28H FLYBACK RANGE = F9.2,7H N. MI.)
1525 FORMAT(29H BANK ANGLE DURING REENTRY = F6.2,4H DEG)
2000 FORMAT (1H ,41HCHBODY = HEIGHT OF BODY / VBODY** .3333 =,F9.4)
3000 FORMAT (1H ,41HCLBODY = LENGTH OF BODY / VBODY** .3333 =,F9.4)
4000 FORMAT (1H ,47HCSBODY = WETTED AREA OF BODY / VBODY ** .6667 =,
* F9.4)
6000 FORMAT (1H ,42HCSFAIR = PAIRING AREA / BODY WETTED AREA =,F9.4)
7000 FORMAT (1H ,58HCSFUTK = WETTED AREA OF FUEL TANKS / TANK VOLUME **
*.6667 =,F9.4)
8000 FORMAT(1H ,43HCSHOWZ = STABILIZER AREA/THEO WING AREA= ,F9.4)
9000 FORMAT (1H ,63HCSOXTK = WETTED AREA OF OXIDIZER TANKS / TANK VOLUM
*E ** .6667 =,F9.4)
10000 FORMAT (1H ,54HCSPLAN = VEHICLE PLANFORM AREA / BODY VOLUME **.666
*7 =,F9.4)
11000 FORMAT (1H ,57HCSVERT = AREA OF EACH VERTICAL SURFACE / VBODY ** .

```

*6667 =,F9.4)

13000 FORMAT (2H ,30HCTHRST = ENGINE THRUST / WTO =,F9.4)

13002 FORMAT(47H1CTHST2 = AIRBREATHING ENGINE THRUST / WENTRY =,F9.4)

16000 FORMAT (2H0,24HNCREW = NUMBER OF CREW =,F5.0)

17000 FORMAT (2H ,27HNEGS = NUMBER OF ENGINES =,F4.0)

20000 FORMAT (2H ,22HRHOFU = FUEL DENSITY =,F6.2)

21000 FORMAT (2H ,43HRHOFU2= SECONDARY PROPULSION FUEL DENSITY =,F6.2)

22000 FORMAT (2H ,25HRHOX = OXIDIZER DENSITY =,F6.2)

22010 FORMAT (2H ,45HRHOX2 = SECONDARY PROPULSION OXIDIZER DENSITY,F6.2)

24000 FORMAT (2H ,66HTTOT = MAIN PROPULSION THRUST = CTHRST * WORBIT + C
 *(129) * WENG5 = / 10X
 * F6.3,3H * ,F7.0, 3H + ,F8.0,3H * ,F2.0,3H = ,F7.0)

25000 FORMAT (2H0,64HTTOT2 = AIRBREATHING ENGINE THRUST = CTHST2 X WETUR
 *N C(158) = * / 10X F6.3,3H * ,F7.0,3H + ,F7.0,3H = ,F7.0)

26000 FORMAT (2H ,5HNJET(,I1,3H) =,F10.1)

26001 FORMAT(1H0)

30000 FORMAT(1H0,42HMAXIMUM ASCENT DYNAMIC PRESSURE = F8.2,
 14H PSF)

30001 FORMAT(1H ,42HTIME AT MAX Q = F8.3,
 14H SEC)

30002 FORMAT(1H ,42HALTITUDE AT MAX Q = F8.0,
 13H FT)

30003 FORMAT(1H ,42HMACH NUMBER AT MAX Q = F7.3)

30004 FORMAT(1H ,42HRELATIVE VELOCITY AT MAX Q = F9.2,
 14H FPS)

30005 FORMAT(1H ,42HRELATIVE FLIGHT PATH ANGLE AT MAX Q = F6.2,
 14H DEG)

END

PRECEDING PAGE BLANK NOT FILMED

GOC-08870-002

Appendix IV-43 Block Data

BLOCK DATA

REAL

1 LAT, LNG, ITT1, ITT2, ITT3, LAM, MULT, JETW, LAMC,

2 LAMDT

DIMENSION

1 TWD11(15), TWD12(15), TWD13(15), AREF1(15), AREF2(15), AREF3(15),
 2 STEP(15), TOL1(15), TOL2(15), TOL3(15), TOL4(15), TOL5(15),
 3 TOL6(15), TOL7(15), HTL1(15), HTL2(15), HTL3(15), HTL4(15),
 4 HTL5(15), HTL6(15), HTL7(15), STGC(15), STGV(15), STGT(15),
 5 STGD(15), JETW(15), ATMC(15), AC1(15), AC2(15), AC3(15),
 6 TWC(15), SIGC(15), ALPC(15), LAMC(15), PSIDT(15), SIGDT(15),
 7 ALPDT(15), LAMDT(15), HMIN(15), HCOEF(15), BCKT(15), BCKTT(15),
 8 SPSI(15), SSIG(15), SALP(15), SLAM(15), XKA1(1,1), XKA2(2,5),
 9 XKA3(2,5), TWD1(15), TWD2(15), TWD3(15), TWD4(15), TWD5(2),
 1 TWD6(15), TWD10(15), GDOT(15), XOUT(15), CVC1(4), CVC2(4),
 2 CVC3(4), CVN1(4), CVN2(4), CVN3(4), ECC1(4), ECC2(4),
 3 ECC3(4), ECN1(4), ECN2(4), ECN3(4), DCV1(4), DCV2(4),
 4 DCV3(4), CVM1(4), CVM2(4), CVM3(4), CVL1(4), CVL2(4),
 5 CVL3(4), ECT1(4), ECT2(4), ECT3(4), EC1(4), EC2(4),
 6 EC3(4), GIS(4), ITT1(4), ITT2(4), ITT3(4), CRGF(4),
 7 TLAT(4), TLNG(4), TOL8(15), HTL8(15), ORB(15), GMC1(15),
 8 GMC2(15), GMC3(15), GMC4(15), XKPT(4), PITX(15), PSEC(15),
 9 PTME(15), PRIF(15), TWD14(15), VSET(15), VINC(15), WOUT(15),
 1 PDALT(4)

COMMON/DATAX/ HANM,HPNM,TRANOM ,

1 RAD, VEL, GAM, AZM, LAT, LNG, WGH, ALTO, SEC,
 2 PLD, CVC1, CVC2, CVC3, CVN1, CVN2, CVN3, ECC1, ECC2,
 3 ECC3, ECN1, ECN2, ECN3, DCV1, DCV2, DCV3, CVM1, CVM2,
 4 CVM3, CVL1, CVL2, CVL3, ECT1, ECT2, ECT3, GO, CK,
 5 WE, D2, D3, D4, AGQ, ATQ, PO, CNV1, CNV2,
 6 CNV3, ALTO, STPF, A, B, RDALT, TO, EC1, EC2, EC3,
 7 GIS, ZOUT, ITT1, ITT2, ITT3, PSI, SIG, ALP, LAM,
 8 MULT, VELO, PRNT, PSIF, CRGF, TLAT, TLNG, ALTF, AZMF,
 9 TWE, TWD11, TWD12, TWD13, AREF1, AREF2, AREF3, STEP, TOL1,
 1 TOL2, TOL3, TOL4, TOL5, TOL6, TOL7, HTL1, HTL2, HTL3,
 2 HTL4, HTL5, HTL6, HTL7, STGC, STGV, STGT, STGD, JETW,
 3 ATMC, AC1, AC2, AC3, TWC, SIGC, ALPC, LAMC, PSIDT,
 4 SIGDT, ALPDT, LAMDT, HMIN, HCOEF, BCKT, BCKTT, SPSI, SSIG,
 5 SALP, SLAM, YOUT, CNV4, XKA0, XKA1, VSET, VINC, WOUT,
 6 XKA2, XKA3, RCON, AZMI, GTIP, TWD1, TWD2, TWD3, TWD4,
 7 TWD5, TWD6, TWD14, PRIF, PTME, TWD10, VGAI, GDOT, XOUT,
 8 TOL8, HTL8, ORB, GMC1, GMC2, GMC3, GMC4, XKPT, PITX,
 9 PSEC

DATA HANM, HPNM,TRANOM / 3*0. / ,

1 RAD, VEL, GAM, AZM, LAT,
 2 LNG, WGH, ALTO, SEC, PLD,
 3 CVC1, CVC2, CVC3, CVN1, CVN2,
 4 CVN3, ECC1, ECC2, ECC3, ECN1,
 5 ECN2, ECN3, DCV1, DCV2, DCV3,
 6 CVM1, CVM2, CVM3, CVL1, CVL2,
 7 CVL3, ECT1, ECT2, ECT3, GO,
 8 CK, WE, D2, D3, D4,

9 AGC,	ATQ,	PO,	CNV1,	CNV2/
1 0.,	0.,	90.,	270.,	34.5815,
2 -120.6233,	0.,	0.,	7.,	0.,
3 4*0.,	4*0.,	4*0.,	4*0.,	4*0.,
4 4*0.,	4*0.,	4*0.,	4*0.,	4*0.,
5 4*0.,	4*0.,	4*0.,	4*0.,	4*0.,
6 4*0.,	4*0.,	4*0.,	4*0.,	4*0.,
7 4*0.,	4*0.,	4*0.,	4*0.,	32.174049,
8 1.407654 E16,	.41780746 E-2,	1082.30 E-6,	-2.30 E-6,	-1.80 E-6,
9 85.,	0.,	14.751,	3.14159265,	57.29578/

DATA

1 CNV3,	ALTO,	STPF,	A,	B,
2 TO,	EC1,	EC2,	EC3,	GIS,
3 ZOUT,	ITT1,	ITT2,	ITT3,	PSI,
4 SIG,	ALP,	LAM,	MULT,	VELQ,
5 PRNT,	PSIF,	CRGF,	TLAT,	TLNG,
6 ALTF,	A7MF,	TWE,	RFCN,	A7MI,
7 GTIP,	VGAI,	YOUT,	CNV4,	RDALT/
1 6076.1033,	300000.,	0.,	20925741.,	20855591.,
2 0.,	4*0.,	4*0.,	4*0.,	4*0.,
3 0.,	4*10.,	4*10.,	4*10.,	0.,
4 0.,	0.,	0.,	0.,	10.,
5 0.,	1.,	4*0.,	4*0.,	4*0.,
6 0.,	0.,	1800.,	0.,	1000.,
7 1000.,	0.,	0.,	.30480061 E-3,	4*0./

DATA

1 TWD11,	TWD12,	TWD13,	AREF1,	AREF2,
2 AREF3,	STEP,	TOL1,	TOL2,	TOL3,
3 TOL4,	TOL5,	TOL6,	TOL7,	HTL1,
4 HTL2,	HTL3,	HTL4,	HTL5,	HTL6/
1 15*0.,	15*0.,	15*0.,	15*0.,	15*0.,
2 15*0.,	2., 14*5.,	15*1.,	15*.5,	15*.1,
3 15*.1,	15*.1,	15*.1,	15*1.,	15*.1,
4 15*.05,	15*.01,	15*.01,	15*.01,	15*.01/

DATA

5 HTL7,	STGC,	STGV,	STGT,	STGD,
6 JETW,	ATMC,	AC1,	AC2,	AC3,
7 TWC,	SIGC,	ALPC,	LAMC,	PSIDT,
8 SIGDT,	ALPDY,	LAMDT,	HMIN,	HCOEF,
9 BCKT,	BCKTT,	SPSI,	SSIG,	SALP/
5 15*.1,	15*0.,	15*0.,	15*.001,	15*1.,
6 15*0.,	15*0.,	15*0.,	15*0.,	15*0.,
7 15*0.,	15*1.,	15*1.,	15*1.,	15*0.,
8 15*0.,	15*0.,	15*0.,	15*.05,	15*2.,
9 15*500.,	15*.001,	15*0.,	15*0.,	15*0./

DATA

1 SLAM,	XKA0,	XKA1,	XKA2,	XKA3,
2 TWD1,	TWD2,	TWD3,	TWD4,	TWD5,
3 TWD6,	TWD10,	GDOT,	XOUT,	TOL8,
4 HTL8,	ORR,	GMC1,	GMC2,	GMC3,
5 GMC4,	XKPT,	PITX,	PSEC,	PTNE,
6 PRIF,	TWD14,	VSET,	VINC,	WOUT/

GOC-08870-002

```
1 15*0.,      193.,      0.,      10*0.,      10*0.,
2 15*1.,      15*0.,      15*1.,      15*0.,      2*0.,
3 15*0.,      15*0.,      15*0.,      15*1.,      15*50000000.,
4 15*10000000., 15*0.,      15*0.,      15*0.,      15*0.,
5 15*0.,      4*0.,      15*1.,      15*1.,      15*0.,
6 15*0.,      15*0.,      15*0.,      15*0.,      15*0./
END
```

APPENDIX V

SPECIFIED \$DATA1 PARAMETERS

Certain \$DATA1 input parameters are either internally specified, or the program operates in a manner which is equivalent to an internal specification, or the parameter must necessarily be redefined as part of the synthesis process. Those parameters in the first two categories are listed together with their corresponding specified value in sub -appendix V-1. The latter category of parameters is listed in sub-appendix V-2. These parameters (all three categories) are elements in the \$DATA1 input array and consequently can be physically included in the \$DATA1 input, however because they are internally defined (subsequent to the reading of \$DATA1) the internal value is used during the program operation and any externally input value is lost. This internal specification of values also supersedes the parameter values which are compiled in by any DATA statements in BLOCK DATA. It is noted that the DATA statement values are listed in the paragraphs of Section 4.2.3; the underlined values indicate that the corresponding parameter could have its value redefined and consequently that the user should reference this appendix to determine if the value has been redefined and if any different values can be input.

APPENDIX V-1

SPECIFIED VALUE PARAMETERS

The values of the parameters * in this sub- appendix are either specified internally or else the program operates in a manner which is equivalent to an internal specification. The value of these parameters cannot be redefined by input. Most of these parameters are redefined in subroutine TRAJC.

PARAMETER	SPECIFIED VALUE
AC2(1) →AC2(3) **	1., and 2., simultaneously
AC2(4) →AC2(7) **	3.,
AC3(1) →AC3(3) **	1., and 2., simultaneously
AC3(4) →AC3(7)	3.,
ALPC(1) & ALPC(2)	8.,
ALPC(3) & ALPC(4)	1.,
CVC1(1)	62.,
CVN1(1)	201.,
CVN1(2)	5.,
DCV1(1)	.01,
ECC1(2)	8.,
ECC2(2)	42.,
ECN1(1)	3.,
ECN1(2)	7.,
ECN2(2)	7.,
GIS(1)	1.,
GIS(2)	4.,
PLD	0.,
RFCN	7.,
SALP(7)	0.,
STGC(2)	7.,
STGC(3)	0.,
STGC(4)	0.,
STGC(5)	0.,

*Note the subscript values, if appropriate. If a \$DATA1 input parameter together with its subscript value does not appear in Appendix V, then this parameter can be input (e.g., AC2(9)) can be input even though AC2(K) for K=1,7 appear in this list since AC2(K) for K=9 is not included).

**The parameter values are not actually redefined, however the program operation is equivalent to these parameters having the values listed in the "Specified Value" column.

STGC(7)	41.,
STGD(2)	-1.,
STGD(3)	1.,
STGD(4)	1.,
STGD(5)	1.,
STGD(7)	1.,
STGT(2)	1.,
STGT(3)	.001,
STGT(4)	.001,
STGT(5)	.001,
STGT(7)	1.,
TWC(1)	6.,
TWC(2)	6.,
TWC(3)	0.,
TWC(4)	0.,
TWC(5)	6.,
TWC(6)	6.,
TWC(7)	6.,

If "SEC" is input with a value less than 7, SEC is internally set equal to 7. Input "SEC" values greater than seven are accepted and processed accordingly.

If "HANM" is input with a value greater than 0., the option to target the orbiter to apogee altitude, perigee altitude, and injection true anomaly is specified. In this case the values of the following \$DATA1 input parameters are internally redefined.

<u>PARAMETER</u>	<u>SPECIFIED VALUE</u>
EC1(2)	0.,
EC2(2)	0.,
ECC1(2)	101.,
ECC2(2)	102.,
STGC(7)	103.,
STGV(7)	0.,

If "SOLID" (a \$DATA2 input parameter) is input with a value greater than 0., the option to include strap-on solid rocket motors is specified. In this case the following \$DATA1 input parameters are internally redefined.

<u>PARAMETER</u>	<u>SPECIFIED VALUE</u>
JETW(1)	SINERT *SOLID where "SINERT" and "SOLID" are \$DATA2 input parameters
STGC(1)	0.,
STGD(1)	1.,
STGT(1)	.001,
STGV(1)	TSBO where "TSBO" is a \$DATA2 input parameter

If "FLYBCK" (a \$DATA2 input parameter) is input with a value equal to 1., "ORB(3)" is set equal to 2. This redefinition occurs in subroutine TRAJA.

APPENDIX V-2 PARAMETER DEFINED DURING SYNTHESIS

This sub-appendix lists those \$DATA1 input parameters which by necessity as part of the synthesis process must be redefined in order to obtain a trajectory which is compatible with the current synthesis status. The implied synthesis parameter definition supersedes any input (via \$DATA1) or any BLOCK DATA compiled in DATA statements. These parameters are redefined in subroutine TAMPER.

<u>PARAMETER</u>	
ALPDT(K)	for K = 5, 6, 7
AREF2(K)	for K = 1 to and including 15
AREF3(K)	for K = 1 to and including 15
JETW(3)	
PITX(K)	for K = 1, 2
STGV(2)	
TWD11(K)	for K = 1, 2, 5, 6, 7
TWD12(K)	for K = 1, 2, 5, 6, 7
TWD13(K)	for K = 1, 2, 5, 6, 7
WGH	

APPENDIX VI

SYNTHESIS DRIVER PRINCIPAL SUBSCRIPTED ARRAYS

In the SSSP \$DATA2 input parameters and internally computed parameters necessary for driving the synthesis process are stored in subscripted arrays for ease of data handling when transferring from one OVERLAY to another and therefore will be found throughout the program as a subscripted parameter. These principal arrays are defined in this appendix. If a parameter is a \$DATA2 input, the parameter is defined in Section 4.2.2, \$DATA2. The principal arrays and their internal dimensions are: SV(50), SE(40), SW(50), SQ(40,5). Where an array parameter is not listed, it has no internal value and has been reserved for future use.

SV ARRAY

<u>Internal Parameter</u>	<u>Input Parameter</u>	
SV(1)		Number of stages - OBSOLETE
SV(2)	IDVEL	
SV(3)		Current total characteristic velocity (fps) from ascent trajectory
SV(4)		Current orbiter burnout weight in parking orbit (lb)
SV(5)		Current orbiter propellant expended to attain parking orbit insertion (lb)
SV(6)		Current orbiter main impulse mass ratio
SV(7)		Current orbiter gross weight (lb)
SV(8)		Current staging relative velocity magnitude (fps)
SV(9)		Current staging altitude (ft)
SV(10)		Current staging relative flight path angle (deg)
SV(11)		Current booster vacuum thrust/gross weight
SV(12)		Current staging dynamic pressure (psf)
SV(13)		Current booster sea level thrust/gross weight
SV(14)		SV(30) - SV(28)
SV(15)		Previous value of SV(14)
SV(16)		Previous value of booster main impulse mass ratio
SV(17)		SV(24)/SV(27)
SV(18)		NOT USED
SV(19)		NOT USED
SV(20)		Correlation factor used to estimate initial pitch rate during ascent, a function of the launch azimuth
SV(21)		NOT USED
SV(22)		NOT USED

SV(23)	Correlation factor per SV(20), a function of current liftoff thrust/weight and booster main impulse mass ratio
SV(24)	Correlation factor per SV(20), a function of staging dynamic pressure and previous value of initial pitch rate
SV(25)	NOT USED
SV(26)	NOT USED
SV(27)	Previous value of SV(23)
SV(28)	Correlation factor per SV(20)
SV(29)	COPIES
SV(30)	Current value of SV(13) - 1.0
SV(31)	Current relative azimuth at staging (deg)
SV(32)	Current geocentric latitude at staging (deg)
SV(33)	Current geocentric longitude at staging (deg)
SV(34)	Current booster weight prior to cruise phase of flight (lb)
SV(35)	GTSM subprogram flag: = 1., for ascent simulation = 2., for booster entry/return simulation (if used)
SV(36)	Section number of last simulation section used for ascent trajectory (=7)
SV(37)	Current geocentric radius magnitude at staging (ft)
SV(38)	Current vehicle bank angle at staging (deg)
SV(39)	Current vehicle pitch angle-of-attack at staging (deg)
SV(40)	Current vehicle yaw angle-of-attack at staging (deg)
SV(41)	Current vehicle pitch attitude (as measured from geo- centric radius vector) at staging (deg)
SV(42)	Current maximum load factor obtained during booster entry/return trajectory (if simulated)

SV(43)

Control flag to test whether the return trajectory (if simulated) should be printed for the final converged synthesis cycle.

= 1.1, no print, iterations continuing

= 2.1, no print, convergence attained but cycle weight/
volume print and ascent trajectory print first

= 3.1, print return for converged synthesis case

SV(44) - SV(50)

NOT USED

SE ARRAY

<u>Internal Parameter</u>	<u>Input Parameter</u>
SE(1)	IVACB(1)
SE(2)	IVACO(1)
SE(3)	ISLB(1)
SE(4)	ISLO(1)
SE(5)	FIRE
SE(6)	BOOTW
SE(7)	QMXS
SE(8)	FBPAR
SE(9)	QMAX
SE(10)	NOT USED
SE(11)	IVACB(2)
SE(12)	IVACO(2)
SE(13)	ISLB(2)
SE(14)	ISLO(2)
SE(15)	IVACO(5)
SE(16)	ISLO(5)
SE(17)	IVACO(6)
SE(18)	ISLO(6)
SE(19)	IVACO(7)
SE(20)	ISLO(7)
SE(21)	TFCTRO(1)
SE(22)	TFCTRO(2)
SE(23)	TFCTRO(5)
SE(24)	TFCTRO(6)
SE(25)	TFCTRO(7)

SE(26)	TFCTRB(1)
SE(27)	TFCTRB(2)
SE(28)	NOT USED
SE(29)	IVACB(3)
SE(30)	IVACO(3)
SE(31)	ISLB(3)
SE(32)	ISLO(3)
SE(33)	IVACB(4)
SE(34)	IVACO(4)
SE(35)	ISLB(4)
SE(36)	ISLO(4)
SE(37)	TFCTRB(4)
SE(38)	TFCTRO(4)
SE(39)	NXFOB
SE(40)	Orbiter fuel burned during initial ascent sections 1 & 2 when SE(39) = 1. and SE(5) = 1.

SW ARRAY

<u>Internal Parameter</u>	<u>Input Parameter</u>	
SW(1)		Number of calls to OVERLAY 1 (SYNTH)
SW(2)		Flag for basic synthesis convergence. = 0.5, continue iterations = 1.0, convergence = 2.0, printout converged case
SW(3)		Basic synthesis iteration counter
SW(4)	SYNIT	
SW(5)	TOLMU	
SW(6)	TRATIO	
SW(7)	PERISP	
SW(8)		Slope used for Newton-Raphson iteration procedure if $SG(15,1) > 0$.
SW(9)	CLVG	
SW(10)		Current range angle from liftoff to staging (rad)
SW(11)	ALD	
SW(12)	SFC	
SW(13)	SLVOUT	
SW(14)	VCRUSE	
SW(15)		Current booster cruise reference range requirement (n.mi)
SW(16)	WTOUT	
SW(17)	TWLO	
SW(18)	TOLTW	
SW(19)	TWLOI	
SW(20)		Current value of absolute time (sec) from liftoff at staging
SW(21) - SW(50)		NOT USED

SQ ARRAY

<u>Internal Parameter</u>	<u>Input Parameter</u>
SQ(1,1)	PRNTX(1)
SQ(1,2)	PRNTX(2)
SQ(1,3)	PRNTX(3)
SQ(2,1)	FSEC
SQ(3,1)	\$DATA1 input value of HANM = apogee altitude of parking orbit (n. mi)
SQ(3,2)	\$DATA1 input value of HPNM = perigee altitude of parking orbit (n. mi)
SQ(3,3)	\$DATA1 input value of TRANOM = true anomaly at insertion point of parking orbit (deg)
SQ(3,4)	Cosine of \$DATA1 input value of TRANOM
SQ(3,5)	Sine of \$DATA1 input value of TRANOM
SQ(5,1)	Current value of absolute time (sec) at Q_{max}
SQ(6,1)	Current value of altitude (ft) at Q_{max}
SQ(7,1)	Current value of relative velocity (fps) at Q_{max}
SQ(8,1)	Current value of relative flight path angle (deg) at Q_{max}
SQ(9,1)	Current value of Mach number at Q_{max}
SQ(10,1)	WOREQ
SQ(10,2)	Bank angle (deg) used to constrain total load factor ≤ 4 g when SQ(19,5)=1.
SQ(10,3)	DRNG
SQ(11,1)	Slope used for Newton-Raphson iteration procedure if SQ(13,1) > 0., or SQ(10,1) > 0.
SQ(12,1)	Current ballistic impact range angle (rad) at staging if SQ(19,5) = 4.
SQ(13,1)	WPOREQ

SQ(14, 1)		Current value of booster characteristic velocity (fps)
SQ(14, 2)		Current value of vehicle characteristic velocity (fps) to parking orbit insertion
SQ(14, 3)		Current value of orbiter characteristic velocity (fps) to parking orbit insertion
SQ(14, 4)		Parking orbit insertion inertial-velocity (fps)
SQ(14, 5)		Parking orbit insertion altitude (ft)
SQ(15, 1)		Previous value of SQ(15, 2)
SQ(15, 2)		Current value of liftoff thrust/weight - 1.0 for vehicle
SQ(15, 3)		Parking orbit insertion inertial flight path angle (deg)
SQ(15, 4)		Parking orbit inclination (deg)
SQ(16, 1)	GWREQ	
SQ(19, 1)		Previous value of SW(15)
SQ(19, 2)		Previous value of booster main impulse mass ratio if SQ(13, 1) > 0., or SQ(10, 1) > 0.
SQ(19, 3)		Cruise parameter for booster used to update cruise performance parameter if SQ(13, 1) > 0., or SQ(10, 1) > 0.
SQ(19, 5)	FLYBCK	
SQ(20, 1)	SOLID	
SQ(20, 2)	AS	
SQ(20, 3)	BS	
SQ(20, 4)	SISP	
SQ(20, 5)	SINERT	
SQ(21, 1)	SAE	
SQ(21, 2)	TSBO	
SQ(21, 3)		Total propellant for solid rockets
SQ(21, 4)		Total sea level thrust for solid rockets (lb)

SQ(21, 5)	Gross weight of solid rockets (lb)
SQ(22, 1)	Propellant weight per solid rocket (lb)
SQ(22, 2)	Sea level thrust per solid rocket (lb)
SQ(30, 1)	Total solid rocket thrust at time t (lb)
SQ(30, 2)	Total shuttle main thrust at time t (lb), throttled or unthrottled
SQ(30, 3)	Total shuttle main propellant flow rate (lb/sec) at time t, throttled or unthrottled
SQ(30, 4)	Total solid rocket propellant flow rate (lb/sec)
SQ(30, 5)	Total shuttle main thrust at time t (lb), unthrottled condition
SQ(31, 1)	Total shuttle main propellant flow rate at time t (lb/sec), unthrottled condition
SQ(32, 1)	FBFUEL
SQ(32, 2)	CA
SQ(32, 3)	CB
SQ(32, 4)	WFLYX
SQ(32, 5)	RT
SQ(33, 1)	R1
SQ(33, 2)	R3
SQ(33, 3)	SFC1
SQ(33, 4)	SFC2
SQ(33, 5)	SFC3
SQ(34, 1)	ALD1
SQ(34, 2)	ALD2
SQ(34, 3)	ALD3
SQ(34, 4)	VFLY1
SQ(34, 5)	VFLY2
SQ(35, 1)	VFLY3

SQ(35, 2)	Idle descent booster cruise fuel weight (lb)
SQ(35, 3)	Main booster cruise fuel weight (lb)
SQ(35, 4)	Final descent booster cruise fuel weight (lb)
SQ(35, 5)	Cruise range (n. mi)
SC(36, 1)	Current drag velocity loss (fps) at staging
SC(36, 2)	Current gravity velocity loss (fps) at staging
SC(36, 3)	Current total drag velocity loss (fps) at parking orbit insertion
SC(36, 4)	Current total gravity velocity loss (fps) at parking orbit insertion
SC(36, 5)	Current total thrust misalignment velocity loss (fps) at parking orbit insertion
SC(37, 1)	Booster burn time (sec)
SC(37, 2)	Orbiter burn time (sec) with SE(5) = 2.
SC(37, 3)	Absolute time from liftoff to end of flight simulation section 4. (sec)
SC(37, 4)	Orbiter flight performance reserve propellant weight (lb)
SC(37, 5)	Booster flight performance reserve propellant weight (lb)

APPENDIX VII

SAMPLE CASE LISTING

This appendix contains the listing of the input and output for the sample SSSP case discussed in Section 6. A listing of the \$DATA1 input data block is included for comparison with the normal \$DATA1 output for which certain parameters will only have values after the initial iteration for the configuration has been completed.

CIVERY = 0.455E+03,
CAXING = 0.1010E+04,
CINRST = 0.0,
CINSTR = 0.0,
FNUOTS = 0.0,
YAP = 0.0, 0.0, 0.4516E+03, 0.4516E+03, 0.416, 0.0,
YAP = 2,
K = 0.5E+01, 3.9E+01, 0.0, 0.0, 0.0, 0.0, 0.15E+00, 0.0, 0.0,
0.10E+05, 0.45E+00, 0.0, 0.0, 0.0, 0.0, 0.10E+05, 0.0,
0.0, 0.55E+00, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0,
KIN = 0.1E+01,
LF = 3.32E+01,
NR = 0.0, 0.0, 0.14003E+05, 0.225E+03, 0.15E+04, 0.0,
NRN = 0.2E+01,
NENG = 0.2E+01,
NLISTO = 0,
NPAS = 0.0,
LNL = 0,
PCMAN = 0.1E+04,
O = 0.7E+03,
RPOPU = 0.44E+01,
RPOFUE = 0.29E+01,
RNOX = 0.71E+02,
RNOZ = 0.49E+02,
SEOOY = 0.11991E+05,
TOL = 0.0,
TOVERC = 0.14E+00,
YRATO = 0.396E+00,
YVAIL = 0.0,
YCOY = 0.725E+05,
WROSS = 0.637304E+00,

SPANTX

ANENG - 8.0%

ANTANK = 0.0,

150A70 = 0.698E+01,

ASWEEP = 0.14E+02,

2

[illegible]

C8800Y = 0.32E+02,

CPUEL = 0.0, 0.0, 0.0E+01, 9.5E+01, 0.5E+01, 0.0,

CH007 - 0.315E+02,

CLP00Y * 0.214837+03,

CSNOY - 0.215120

ESPAIN - 0.0.

CSFUTK = 6.654E+01,

CGNORZ = 0.2643E+04,

CSQYTK = 0.55E+01,

CSPLAH - 0.7276E+04,

```

CSVERT = 0.1E+04,
CSNING = 0.4841E+04,
CTMRST = 0.0,
CTMSTZ = 0.0,
FNOVS = 0.0,
ISP = 0.0, 0.0, 0.4374E+03, 0.0, 0.0, 0.0,
TTPS = 2,
K = 0.5E-01, 0.5E-01, 0.0, 0.0, 0.0, 0.25E+03, 0.0, 0.0, 0.0,
      0.6E+00, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.3E+00, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.1E+05, 0.0, 0.0,
--EN KIN = 0.1E+01,
LP = 0.322E+01,
MQ = 0.0, 0.0, 0.3045E+01, 0.0, 0.0, 0.0,
MCREN = 0.2E+01,
MEMS = 0.11E+02,
MLISTO = 0,
MPASS = 0.0,
MWL = 0,
PCMHM = 0.1E+04,
O = 0.7E+03,
RNOFU = 0.44E+01,
RNOFUZ = 0.0,
--EN RNOX = 0.71E+02,
RNOX2 = 0.0,
SNOOY = 0.2083E+05,
TCL = 0.0,
TOVE=C = 0.14E+00,
TPRATO = 0.336E+00,
--EN TTTAIL = 0.0,
VQDOY = 0.10826E+06,
VQMOSS = 0.3384401E+07,
SEND

```

OUTPUT OF \$DATA2 INPUT

```

SCATA2
ALO = 0.05E+01,
FOPAR = 0.25E+03,
TOVEL = 0.30201E+05,
YALB = 0.3910E+03, 0.3910E+03, 0.3910E+03, 0.3910E+03,
YSLC = 0.230E+03, 0.230E+03, 0.230E+03, 0.230E+03, 0.230E+03,
YVACB = 0.4374E+03, 0.4374E+03, 0.4374E+03, 0.4374E+03,
YVACB = 0.4516E+03, 0.4516E+03, 0.4516E+03, 0.4516E+03, 0.4516E+03,
PERISP = 0.0097E+00,
OMAX = 0.55E+03,
OMX3 = 0.9E+03,
SFC = 0.3E+00,
SLVOUT = 0.0,
COPIES = 0.2E+01,
SYNIT = 0.5E+01,
YFACTRB = 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01,
YFACTRB = 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01,
TOLMU = 0.5E-03,
TOLTM = 0.1E-02,
TRATIO = 0.96056E+00,
YHLO = 0.1371E+01,
YHLOI = -0.1E-01,
WTOUY = 0.0,
FTRE = 0.2E+01,
R2OTW = 0.0,
VCRUSE = 0.29E+03,
MHF00 = 0.0,
PEN7X = 0.1E+01, 0.0, 0.0,
FSEC = 0.1E+01,
CLVC = 0.1E+01,

```

1051

ORNG = 0.0,
COLTO = 0.0,
AS = 0.0,
OS = 0.0,
STSP = 0.0,
SINERT = 0.0,
AN SAE = 0.0,
YAMO = 0.0,
FLYACK = 0.2E+01,
WFOPEO = 0.0,
WOREO = 0.0,
GUREO = 0.330639E+07,
NA PFUEL = 0.1E+01,
CA = 0.0,
CM = 0.0,
WFLVX = 0.0,
QT = 0.0,
R1 = 0.0,
R2 = 0.0,
SFC1 = 0.0,
SFC2 = 0.0,
SFC3 = 0.0,
AL01 = 0.1E+01,
AL02 = 0.1E+01,
AL03 = 0.1E+01,
VFLV1 = 0.1E+01,
VFLV2 = 0.1E+01,
VFLV3 = 0.1E+01,
SEND

APSDATA1

MTL2 = 2°.015, 9°.04, MTL3 = 2°.000A, 5°.000, TOL2 = 2°.05, 4°.005,

LISTING OF \$DATA1 INPUT CARDS

(not normal output on UNIVAC 1108 computer)

```

ATMC= 7.1.,
SALP= 4.0.,
ALPOY= 7.0.,
WHIM= 7.0.5,
ORR= 6.0., 1.,
PSEC=1.,
XKPT(1)= 3.,
MCKT= 2.400., 2.50., 200., 2.400.,
VOUT=1.,
STEP= 5.2., 2.5.,
MTL2= 2.015, 5.05,
MTL3= 2.001, 4.001,
ALTO=60000.,
TOL2= 2.1, 5.4,
PIYX=2.0-5,
XOUT=7.5.,
VOUT=4.,
LAT=20.32, LMG=-00.54,
AZM=37.65,
PT(1,1)= 0., 14., 24.,
POY(1,1)= 0., 20,
SALP(5)= 15., 5.,
ALPOY(5)= 0., 2.0, 0.4,
TM014= 2.2.5, 2.0., 3.3.,
XKA2(1,1)= 17.,
XKA2(2,1)= 2.,
A2(1,1)= -90., 90.,
XK2(1,1)= -1., 0., 15., 5., 75., 03, 1., 1.05, 1.12,
XK2(10,1)= 1.2, 1.4, 1.75, 2.3, 3., 5., 20., 50.,
C2(1,1,1)= 2.0-1205, 2.0-0927, -.13043, -.17059, -.3914, -.4617, -.6710,
C2(10,1,1)= -.4617, -.4015, -.3614, -.3100, -.2400, 3.0-2200,
C2(1,2,1)= 2.0-1205, 2.0-0927, -.13043, -.17059, -.3914, -.4617, -.6710,
C2(10,2,1)= -.4617, -.4015, -.3614, -.3100, -.2400, 3.0-2200,
XKA2(1,2)= 2.,
A2(1,2)= -90., 90.,
XK2(1,2)= 0., 100.,
C2(1,1,2)= 0., 0.,
C2(1,2,2)= 0., 0.,
XKA3(1,2)= 2.,
XKA3(2,2)= 2.,
XK3(1,2)= 7., 100.,
A3(1,2)= -90., 90.,
C3(1,1,2)= 0., 0.,
C3(1,2,2)= 0., 0.,
STGV(1)= 2.5,
STGC(1)= 33.,
STG0(1)= 1.,
STGT(1)= .001,
TOL7= 10., MTL7= 1.,
STGV(3)= 3., 4., 00., 3., 25053.,
STGC(3)= 30., 37., 41.,
STG0(1)= 5.1.,
STGT(3)= 4.001, 1.,
EC1= 0.3010,
ECG1= 51., ECT1= .001,
ECC1=92.,
EC1(2)= 309801., EC2(2)= 0.,
ECT1(2)= 2000., ECT2(2)= .005,
SALP(5)= 30., 20.,
CVC1(2)=45.,
CVC2(2)=45.,

```

14 1050

```

CVM2(2) = 6.,
ALPC(5) = 1.,
ALPC(6) = 1.,
ALPC(7) = 7.,
OCV1(2) = 5.,
OCV2(2) = 5.,
CVM1(1) = 10.,
CWL1(1) = 01.,
CWL1(2) = 60.,
CVM1(2) = 60.,
CVM2(2) = 60.,
CWL2(2) = 60.,
XKA2(1,2) = 2.,
XKA2(2,2) = 2.,
A2(1,2) = -90., 90.,
XK2(1,2) = 0., 100.,
C2(1,1,2) = 0., 0.,
C2(1,2,2) = 0., 0.,
XKA3(1,1) = 2.,
XKA3(2,1) = 2.,
XK3(1,1) = 0., 100.,
A3(1,1) = -90., 90.,
C3(1,1,1) = 0., 0.,
C3(1,2,1) = 0., 0.,
XKA3(1,2) = 2.,
XKA3(2,2) = 2.,
XK3(1,2) = 0., 100.,
A3(1,2) = -90., 90.,
C3(1,1,2) = 0., 0.,
C3(1,2,2) = 0., 0.,
XKA2(1,3) = 2.,
XKA2(2,3) = 2.,
A2(1,3) = -90., 90.,
XK2(1,3) = 0., 100.,
C2(1,1,3) = 0., 0.,
C2(1,2,3) = 0., 0.,
XKA3(1,3) = 2.,
XKA3(2,3) = 2.,
XK3(1,3) = 0., 100.,
A3(1,3) = -90., 90.,
C3(1,1,3) = 0., 0.,
C3(1,2,3) = 0., 0.,
ALPC(5) = 3+19.,
CVC1(2) = 45., CVC2(2) = 35., CVM2(2) = 5.,
OCV1(2) = 4., OCV2(2) = 4.,
CVM1(2) = 170., CVM2(2) = 170.,
CWL1(2) = -70., CWL2(2) = -70.,
SALP(5) = -20., 0., 0., PSIOT(5) = 50., 0., 0.,
ALDOT(5) = 246., 0., 0.,
YOUT = 1.,
SEND

```

OUTPUT OF \$DATA1 INPUT

NOTE: items circled will be computed by the synthesis process and will be supplied to this data block

[illegible]

1049

GENERAL ITERATION BLOCK 2

GTS = 4 *****

CVC1	CVC2	CVC3	CVL1	CVL2	CVL3
CVM1	CVM2	CVM3	CVM1	CVM2	CVM3
OCV1	OCV2	OCV3	EC1	EC2	EC3
ECG1	ECG2	ECG3	ECM1	ECM2	ECM3
ECT1	ECT2	ECT3	ITT1	ITT2	ITT3
4.5000000E+01	3.5000000E+01	0.	-7.0000000E+01	-7.0000000E+01	0.
1.7000000E+02	1.7000000E+02	0.	5.0000000E+00	5.0000000E+02	0.
4.0000000E+00	4.0000000E+00	0.	3.0000000E+05	0.	0.
0.0000000E+00	4.2000000E+01	0.	7.0000000E+00	7.0000000E+00	0.
2.0000000E+03	5.0000000E-03	0.	1.0000000E+01	1.0000000E+01	1.0000000E+01

17

***** SECTION DATA *****

AC1 AREF2 GMC1 HTL1 HTL7 PIVX	AC2 AREF3 GMC2 HTL2 HTL8 PSIDY	AC3 ATMC GMC3 HTL3 JETW PRIF	ALPC ACKT GMC4 HTL4 LAMC PSEC	ALPOT GCKT MCOEF HTL5 LAMOT PTME	AREF1 GCKT MMIN HTL6 ORB SALP
SIGC TOL3 TMC TMD10 VSET	SIGTY STGO TOL4 TMD1 TMD11 MOUT	SLAM STGT YOLS TMD2 TMD12 XOUT	SPSI STGV TOL5 TMD3 TMD13	SSIG TOL1 TOL7 TMD4 TMD14	STEP TOL2 TOL8 TMD6 VINC
0. 0. 1.0000000E-01 1.0000000E+00 -5.0000000E-01	0. 0. 1.5000000E-02 1.0000000E+07 0.	0. 0. 1.0000000E-03 0. 0.	4.0000000E+00 4.0000000E+02 0. 1.0000000E-02 1.0000000E+00 1.0000000E+00	0. 1.0000000E-03 2.0000000E+00 1.0000000E-02 0. 0.	0. 0. 5.0000000E-01 1.0000000E-02 0. 0.
1.0000000E+00 3.3000000E-01 1.0000000E-01 6.0000000E+00 0. 0.	0. 1.0000000E+00 1.0000000E-01 1.0000000E+00 0.	0. 1.0000000E-03 1.0000000E-01 0. 5.0000000E+00	0. 2.5000000E+00 1.0000000E-01 1.0000000E+00 0.	0. 1.0000000E+00 1.0000000E-01 0. 2.5000000E+00	2.0000000E+00 1.0000000E-01 5.0000000E+07 0.
0. 0. 1.0000000E-01 1.0000000E-01 -5.0000000E-01	0. 1.5000000E-02 1.0000000E+07 0.	0. 1.0000000E-03 0. 0.	8.0000000E+00 4.0000000E+02 0. 1.0000000E-02 1.0000000E+00 1.0000000E+00	0. 1.0000000E-03 2.0000000E+00 1.0000000E-02 0.	0. 0. 5.0000000E-01 1.0000000E-02 0.
1.0000000E+00 7.0000000E+00 1.0000000E-01 6.0000000E+00 0.	-1.0000000E+00 1.0000000E-01 1.0000000E+00 0.	0. 1.0000000E+00 1.0000000E-01 0. 5.0000000E+00	0. 0. 0. 0. 0.	0. 1.0000000E+00 1.0000000E+00 0. 2.5000000E+00	2.0000000E+00 1.0000000E-01 5.0000000E+07 0.
0. 0. 1.0000000E-01 1.0000000E-01 1.0000000E+00	0. 5.0000000E-02 1.0000000E+07 0.	0. 1.0000000E+00 0. 0.	1.0000000E+00 5.0000000E+01 0.	0. 1.0000000E-03 1.0000000E+00 0. 2.5000000E+00	0. 0. 5.0000000E-01 1.0000000E-02 0.
1.0000000E+00 0. 1.0000000E-01 0.	0. 1.0000000E+00 1.0000000E-01 1.0000000E+00	0. 1.0000000E-03 1.0000000E-01 0. 5.0000000E+00	0. 3.0000000E+00 1.0000000E+01 0.	0. 1.0000000E+00 1.0000000E+00 0. 2.5000000E+00	2.0000000E+00 1.0000000E-01 5.0000000E+07 0.

ATMOSPHERE TABLE

ALTITUDE (FT)

-3	0.000000E+07	0.000000E+03	1.000000E+03	2.000000E+03	3.000000E+03	4.000000E+03
5	0.000000E+03	6.000000E+03	7.000000E+03	8.000000E+03	9.000000E+03	1.000000E+04
1	0.000000E+04	1.200000E+04	1.300000E+04	1.400000E+04	1.500000E+04	1.600000E+04
1	0.000000E+04	1.800000E+04	1.900000E+04	2.000000E+04	2.100000E+04	2.200000E+04
2	0.000000E+04	2.400000E+04	2.500000E+04	2.600000E+04	2.700000E+04	2.800000E+04
2	0.000000E+04	3.000000E+04	3.100000E+04	3.200000E+04	3.300000E+04	3.400000E+04
3	0.000000E+04	3.600000E+04	3.700000E+04	3.800000E+04	3.900000E+04	4.000000E+04
3	0.000000E+04	4.200000E+04	4.300000E+04	4.400000E+04	4.500000E+04	4.600000E+04
4	0.000000E+04	4.800000E+04	4.900000E+04	5.000000E+04	5.100000E+04	5.200000E+04
4	0.000000E+04	5.400000E+04	5.500000E+04	5.600000E+04	5.700000E+04	5.800000E+04
5	0.000000E+04	6.000000E+04	6.100000E+04	6.200000E+04	6.300000E+04	6.400000E+04
5	0.000000E+04	6.600000E+04	6.700000E+04	6.800000E+04	6.900000E+04	7.000000E+04
7	0.000000E+04	7.200000E+04	7.300000E+04	7.400000E+04	7.500000E+04	7.600000E+04
7	0.000000E+04	7.800000E+04	7.900000E+04	8.000000E+04	8.100000E+04	8.200000E+04
9	0.000000E+04	8.400000E+04	8.500000E+04	8.600000E+04	8.700000E+04	8.800000E+04
9	0.000000E+04	9.000000E+04	9.100000E+04	9.200000E+04	9.300000E+04	9.400000E+04
1	0.000000E+05	1.000000E+05	1.100000E+05	1.200000E+05	1.300000E+05	1.400000E+05
1	0.000000E+05	1.200000E+05	1.300000E+05	1.400000E+05	1.500000E+05	1.600000E+05
1	0.000000E+05	1.300000E+05	1.400000E+05	1.500000E+05	1.600000E+05	1.700000E+05
1	0.000000E+05	1.500000E+05	1.600000E+05	1.700000E+05	1.800000E+05	1.900000E+05
1	0.000000E+05	1.700000E+05	1.800000E+05	1.900000E+05	2.000000E+05	2.100000E+05
1	0.000000E+05	1.900000E+05	2.000000E+05	2.100000E+05	2.200000E+05	2.300000E+05
2	0.000000E+05	2.400000E+05	2.500000E+05	2.600000E+05	2.700000E+05	2.800000E+05
2	0.000000E+05	2.900000E+05	3.000000E+05	3.100000E+05	3.200000E+05	3.300000E+05
3	0.000000E+05	3.400000E+05	3.500000E+05	3.600000E+05	3.700000E+05	3.800000E+05
3	0.000000E+05	3.900000E+05	4.000000E+05	4.100000E+05	4.200000E+05	4.300000E+05
4	0.000000E+05	4.400000E+05	4.500000E+05	4.600000E+05	4.700000E+05	4.800000E+05
4	0.000000E+05	4.900000E+05	5.000000E+05	5.100000E+05	5.200000E+05	5.300000E+05
5	0.000000E+05	5.400000E+05	5.500000E+05	5.600000E+05	5.700000E+05	5.800000E+05
5	0.000000E+05	5.900000E+05	6.000000E+05	6.100000E+05	6.200000E+05	6.300000E+05
6	0.000000E+06	6.400000E+06	6.500000E+06	6.600000E+06	6.700000E+06	6.800000E+06
1	0.000000E+06	6.900000E+06	7.000000E+06	7.100000E+06	7.200000E+06	7.300000E+06
1	0.000000E+06	7.400000E+06	7.500000E+06	7.600000E+06	7.700000E+06	7.800000E+06
2	0.000000E+06	7.900000E+06	8.000000E+06	8.100000E+06	8.200000E+06	8.300000E+06
5	0.000000E+09					

PRESSURE (Lb/IN²)

1	0.750550E+01	1.475100E+01	1.424400E+01	1.375100E+01	1.327300E+01	1.280000E+01
1	0.750550E+01	1.191900E+01	1.149400E+01	1.108200E+01	1.068200E+01	1.029500E+01
9	0.750550E+01	9.555000E+00	9.202200E+00	8.862000E+00	8.528500E+00	8.207600E+00
7	0.750550E+01	7.595000E+00	7.302900E+00	7.020000E+00	6.745900E+00	6.484000E+00
7	0.750550E+01	5.974000E+00	5.733000E+00	5.500000E+00	5.274100E+00	5.056000E+00
4	0.750550E+01	4.670700E+00	4.441000E+00	4.251000E+00	4.066400E+00	3.881200E+00
3	0.750550E+01	3.549600E+00	3.369300E+00	3.245900E+00	3.086200E+00	2.932000E+00
2	0.750550E+01	2.673700E+00	2.546000E+00	2.425200E+00	2.306650E+00	2.197000E+00
1	0.750550E+01	1.907800E+00	1.800100E+00	1.709900E+00	1.623000E+00	1.541000E+00
1	0.750550E+01	1.465300E+00	1.392100E+00	1.326000E+00	1.256700E+00	1.194100E+00
1	0.750550E+01	1.081000E+00	9.772600E-01	8.847000E-01	8.015100E-01	7.266000E-01
6	0.750550E+01	5.903200E-01	5.474000E-01	4.912000E-01	4.493500E-01	4.089900E-01
3	0.750550E+01	3.394500E-01	3.095200E-01	2.823650E-01	2.577000E-01	2.334000E-01
2	0.750550E+01	1.967800E-01	1.795400E-01	1.642400E-01	1.503300E-01	1.376700E-01
1	0.750550E+01	1.156400E-01	1.060700E-01	9.734000E-02	8.938900E-02	8.212300E-02
7	0.750550E+01	6.942700E-02	6.129600E-02	5.402000E-02	4.795000E-02	4.286600E-02
3	0.750550E+01	3.745800E-02	2.975300E-02	2.647700E-02	2.356300E-02	2.102700E-02
1	0.750550E+01	1.673600E-02	1.494100E-02	1.334200E-02	1.191500E-02	1.063500E-02

1047

9.40460000E-03
3.66930000E-03
9.97760000E-04
2.29490000E-04
4.40900000E-05
4.10410000E-06
4.24220000E-07
1.26970000E-07
4.57420000E-08
6.05180000E-09
1.26730000E-09
3.53640000E-10
1.14300000E-10
4.09600000E-11
0.

8.45200000E-03
2.98290000E-03
7.90960000E-04
1.76770000E-04
3.35640000E-05
2.65370000E-06
3.26700000E-07
1.12630000E-07
3.08130000E-08
4.56460000E-09
1.02260000E-09
2.90170000E-10
9.57400000E-11
3.47000000E-11

7.42540000E-03
2.41600000E-03
6.24030000E-04
1.75190000E-04
2.53700000E-05
1.73340000E-06
2.54720000E-07
9.47490000E-08
2.14130000E-08
3.44130000E-09
8.17730000E-10
2.39030000E-10
6.64000000E-11
2.95400000E-11

6.69410000E-03
1.94910000E-03
4.09910000E-04
1.02420000E-04
1.92390000E-05
1.16260000E-06
2.10940000E-07
3.05170000E-08
1.52330000E-08
2.66230000E-09
6.58070000E-10
1.97750000E-10
6.76910000E-11
2.52000000E-11

5.49330000E-03
1.56600000E-03
3.82600000E-04
7.77760000E-05
1.12910000E-05
8.84700000E-07
1.76010000E-07
7.94510000E-08
1.10350000E-08
2.08420000E-09
5.32540000E-10
1.64250000E-10
5.71340000E-11
2.15000000E-11

4.49770000E-03
1.25200000E-03
2.97390000E-04
5.07650000E-05
6.79450000E-06
5.78130000E-07
1.49540000E-07
7.17560000E-08
6.11970000E-09
1.63200000E-09
4.32900000E-10
1.36830000E-10
4.63340000E-11
0.

VELOCITY OF SOUND (FT/SEC)

1.13799900E+03
1.11060000E+03
1.09260000E+03
1.07780000E+03
1.05160000E+03
1.02310000E+03
9.93550000E+02
9.66050000E+02
9.46070000E+02
9.36280000E+02
9.42290000E+02
9.61580000E+02
9.79270000E+02
1.01050000E+03
1.02940000E+03
1.05680000E+03
1.07490000E+03
1.03730000E+03
9.77940000E+02
9.19250000E+02
8.44000000E+02
9.64117000E+02
1.22930000E+03
1.74490000E+03
2.26450000E+03
2.93940000E+03
3.13240000E+03
3.27360000E+03
3.36500000E+03
3.41000000E+03

1.13600000E+03
1.11520000E+03
1.09520000E+03
1.07300000E+03
1.04700000E+03
1.01410000E+03
9.84700000E+02
9.62100000E+02
9.43720000E+02
9.36530000E+02
9.47400000E+02
9.64150000E+02
9.83890000E+02
9.94560000E+02
1.01660000E+03
1.03750000E+03
1.06140000E+03
1.08220000E+03
1.07120000E+03
1.02790000E+03
9.67900000E+02
9.09470000E+02
8.40000000E+02
9.94400000E+02
1.31360000E+03
1.81900000E+03
2.37210000E+03
2.74900000E+03
2.97620000E+03
3.16070000E+03
3.25070000E+03
3.37600000E+03

1.13700000E+03
1.11190000E+03
1.09170000E+03
1.06890000E+03
1.04240000E+03
1.01320000E+03
9.81700000E+02
9.58370000E+02
9.41680000E+02
9.36530000E+02
9.47400000E+02
9.64150000E+02
9.83890000E+02
9.94560000E+02
1.01660000E+03
1.03750000E+03
1.06140000E+03
1.08220000E+03
1.06710000E+03
1.01810000E+03
9.58010000E+02
8.99450000E+02
8.40000000E+02
1.02750000E+03
1.41060000E+03
1.89300000E+03
2.45320000E+03
2.78120000E+03
3.01950000E+03
3.14770000E+03
3.30770000E+03
3.36700000E+03

1.12960000E+03
1.10450000E+03
1.08200000E+03
1.06480000E+03
1.03760000E+03
1.00220000E+03
9.79050000E+02
9.54900000E+02
9.39600000E+02
9.36600000E+02
9.37510000E+02
9.54590000E+02
9.74090000E+02
9.87900000E+02
1.00440000E+03
1.02290000E+03
1.04730000E+03
1.07360000E+03
1.08110000E+03
1.05490000E+03
9.98140000E+02
9.38380000E+02
8.84000000E+02
9.16350000E+02
1.11700000E+03
1.58660000E+03
2.02470000E+03
2.58410000E+03
2.86440000E+03
3.07620000E+03
3.23920000E+03
3.34140000E+03
3.40000000E+03

1.12580000E+03
1.10520000E+03
1.08450000E+03
1.06500000E+03
1.03280000E+03
1.00330000E+03
9.74550000E+02
9.51670000E+02
9.38600000E+02
9.37570000E+02
9.37510000E+02
9.54590000E+02
9.74090000E+02
9.87900000E+02
1.00440000E+03
1.02290000E+03
1.04730000E+03
1.07360000E+03
1.08110000E+03
1.05490000E+03
9.98140000E+02
9.38380000E+02
8.84000000E+02
9.16350000E+02
1.11700000E+03
1.58660000E+03
2.02470000E+03
2.58410000E+03
2.86440000E+03
3.07620000E+03
3.23920000E+03
3.34140000E+03
3.40000000E+03

1.12210000E+03
1.10190000E+03
1.08000000E+03
1.05610000E+03
1.02800000E+03
9.98410000E+02
9.70210000E+02
9.48720000E+02
9.37570000E+02
9.35300000E+02
9.58120000E+02
9.76000000E+02
9.83200000E+02
1.00740000E+03
1.02610000E+03
1.05210000E+03
1.07700000E+03
1.07820000E+03
1.04640000E+03
9.88050000E+02
9.28920000E+02
8.80000000E+02
9.37680000E+03
1.17450000E+03
1.66760000E+03
2.07640000E+03
2.63560000E+03
2.92200000E+03
3.10500000E+03
3.25650000E+03
3.35440000E+03
3.41900000E+03

AERO(2) TABLE 1

MACH NUMBER

0.
1.0500000E+00
4.0000000E+00

1.9000000E-01
1.1200000E+00
5.0000000E+00

5.0000000E-01
1.2000000E+00
2.0000000E+01

7.5000000E-01
1.4000000E+00
5.0000000E+01

0.3000000E-01
1.7500000E+00

PITCH ANGLE OF ATTACK (DEG)

-9.0000000E+01 9.0000000E+01

AERODYNAMIC COEFFICIENT

-1.2050000E-01 -1.2050000E-01
-3.9140000E-01 -4.6170000E-01
-3.0090000E-01 -2.4090000E-01

-9.2700000E-02
-4.7100000E-01
-2.2000000E-01

-9.2700000E-02
-4.6170000E-01
-2.2000000E-01

-1.3043000E-01
-4.0150000E-01
-2.2000000E-01

-1.7059000E-01
-3.6140000E-01

-1.2050000E-01 -1.2050000E-01
-3.9140000E-01 -4.6170000E-01
-3.0090000E-01 -2.4090000E-01

-9.2700000E-02
-4.7100000E-01
-2.2000000E-01

-9.2700000E-02
-4.6170000E-01
-2.2000000E-01

-1.3043000E-01
-4.0150000E-01
-2.2000000E-01

-1.7059000E-01
-3.6140000E-01

AERO(2) TABLE 2

MACH NUMBER

0. 1.0000000E+02

PITCH ANGLE OF ATTACK (DEG)

-5.0000000E+01 9.0000000E+01

AERODYNAMIC COEFFICIENT

0. 0.

0. 0.

AERO(2) TABLE 3

MACH NUMBER

1046

0. 1.0000000E+02

PITCH ANGLE OF ATTACK (DEG)

-9.0000000E+01 9.0000000E+01

AERODYNAMIC COEFFICIENT

0. 0.

0. 0.

AERO(3) TABLE 1

MACH NUMBER

0. 1.0000000E+02

PITCH ANGLE OF ATTACK (DEG)

-9.0000000E+01 9.0000000E+01

AERODYNAMIC COEFFICIENT

0. 0.

0. 0.

AERO(3) TABLE 2

MACH NUMBER

0. 1.0000000E+02

PITCH ANGLE OF ATTACK (DEG)

-5.0000000E+01 9.0000000E+01

1045
24

0.
0.
0.

AERO(3) TABLE 3

MACH NUMBER

0. 1.0000000E+02

PITCH ANGLE OF ATTACK (DEG)

-9.0000000E+01 9.0000000E+01

AERODYNAMIC COEFFICIENT

0.
0.

PITCH RATE TABLE 1

TIME (SEC)

0. 1.4000000E+01 2.4000000E+01

PITCH RATE STEPS (DEG/SEC)

0. 2.0000000E-01

ORBITER MU = 2.77726 WEIGHT = 63043 THRUST = 94000 T/W = 1.402549
 BOOSTER MU = 3.04500 WEIGHT = 3467461 THRUST = 5007435 T/W = 1.293563
 PAYLOAD = 36000.00 LO WT DIFF = -03070.99 SLOPE = -21.1000

ORBITER MU = 2.77726 WEIGHT = 610620 THRUST = 94000 T/W = 1.539428
 BOOSTER MU = 3.04500 WEIGHT = 3370977 THRUST = 5007435 T/W = 1.327442
 PAYLOAD = 32062.99 LO WT DIFF = 5413.03 SLOPE = -22.4749

ORBITER MU = 2.77726 WEIGHT = 612057 THRUST = 94000 T/W = 1.535004
 BOOSTER MU = 3.04500 WEIGHT = 3544433 THRUST = 5007435 T/W = 1.325303

START SSSP
CASE

ITERATIONS

PAYLOAD = 32303.83 LO WT DIFF = -42.40 SLOPE = -22.4749

CONVERG ERR = -.000013

GN GN MU = 2.790235 MTVOL MU = 2.777257 1 INPUT TOL = .00050 TOL = .000451
ORBITER MU = 2.79023 WEIGHT = 613789 THRUST = 940000 T/W = 1.531471
BOOSTER MU = 3.04500 WEIGHT = 3483913 THRUST = 5007435 T/W = 1.267460
PAYLOAD = 32303.83 LO WT DIFF = -99523.16 SLOPE = -22.4749

ORBITER MU = 2.79023 WEIGHT = 587294 THRUST = 940000 T/W = 1.600561
BOOSTER MU = 3.04500 WEIGHT = 3382753 THRUST = 5007435 T/W = 1.325961
PAYLOAD = 27875.64 LO WT DIFF = 1637.25 SLOPE = -22.8446

ORBITER MU = 2.79023 WEIGHT = 587725 THRUST = 940000 T/W = 1.599389
BOOSTER MU = 3.04500 WEIGHT = 3384399 THRUST = 5007435 T/W = 1.325316
PAYLOAD = 27947.31 LO WT DIFF = -9.30 SLOPE = -22.8446

CONVERG ERR = -.000003

GN GN MU = 2.735376 MTVOL MU = 2.790235 2 INPUT TOL = .00050 TOL = .001744
ORBITER MU = 2.78538 WEIGHT = 586038 THRUST = 940000 T/W = 1.603991
BOOSTER MU = 3.04500 WEIGHT = 3377954 THRUST = 5007435 T/W = 1.327845
PAYLOAD = 27947.31 LO WT DIFF = 6436.24 SLOPE = -22.8446

ORBITER MU = 2.78538 WEIGHT = 587729 THRUST = 940000 T/W = 1.599377
BOOSTER MU = 3.04500 WEIGHT = 3384419 THRUST = 5007435 T/W = 1.325308
PAYLOAD = 28229.05 LO WT DIFF = -28.57 SLOPE = -22.8446

CONVERG ERR = -.000000

GN GN MU = 2.785460 MTVOL MU = 2.785376 3 INPUT TOL = .00071 TOL = .000838
ORBITER MU = 2.78546 WEIGHT = 587743 THRUST = 940000 T/W = 1.599338
BOOSTER MU = 3.04500 WEIGHT = 3384471 THRUST = 5007435 T/W = 1.325288
PAYLOAD = 28229.05 LO WT DIFF = -80.68 SLOPE = -22.8446

1044
6

CONVEPG ERR = -.000024

SYNTHESIS ITERATION CONVERGENCE

- PRINT CONVERGED CASE -

1 GMPEQ-3384398 MR=3.845 FLYBACK=2

WEIGHT BREAKDOWN - ORBITAL STAGE

AERODYNAMIC SURFACES
WING & WING MOUNTED CONTROL SURFACES
HORIZONTAL SURFACES
VERTICAL SURFACES
FAIRINGS, SPROUDS AND ASSOCIATED STRUCTURE

RDV STRUCTURE
STRUCTURAL FUEL CONTAINERS
STRUCTURAL OXIDIZER CONTAINERS
BASIC BODY STRUCTURE
SECONDARY STRUCTURE
THRUST STRUCTURE

INDUCED ENVIRONMENTAL PROTECTION
COVER PANELS, NON-STRUCTURAL
VEHICLE INSULATION

LAUNCH, RECOVERY AND DOCKING
LAUNCH GEAR
DEPLOYABLE AERODYNAMIC DEVICES
ALIGNING GEAR
DOCKING STRUCTURE

PROPULSION
ENGINES AND ACCESSORIES
SECONDARY ENGINES AND ACCESSORIES
FUEL CONTAINERS AND SUPPORTS (NON-STRUCTURAL)
OXIDIZER CONTAINERS AND SUPPORTS (NON-STRUCTURAL)
SECONDARY FUEL TANKAGE AND SYSTEMS
SECONDARY OXIDIZER TANKAGE AND SYSTEMS
PROPELLANT INSULATION
FUEL SYSTEM - MAIN
OXIDIZER SYSTEM - MAIN
PURGE SYSTEMS
AIRBREATHING ENGINES AND INSTALLATION
NACELLE, RODS, PYLONS, SUPPORTS
AIRBREATHING PROPULSION TANKAGE AND SYSTEMS

ORIENTATION, SEPARATION AND LIFTAGE CONTROL
SEPARATION SYSTEMS
STABILITY AND CONTROL, ENGINE GENERAL
AERODYNAMIC CONTROLS
SPATIAL ATTITUDE CONTROL SYSTEM
CONTROL PROPELLANT TANKAGE AND SYSTEMS

PRIME POWER SOURCE AND DISTRIBUTION
POWER SOURCE UNITS
POWER SOURCE TANKAGE AND SYSTEMS

POWER CONVERSION AND DISTRIBUTION
HYDRAULIC/PNEUMATIC

GUIDANCE AND NAVIGATION

INSTRUMENTATION

COMMUNICATION

KILOGRAMS

POUNDS

10805

22857

13271
6512
2274
0

6820
2954
1031
0

24405

53805

8611
2591
36270
2572
3760

3906
1175
16452
1167
1706

4887

10775

10775
0

4887
0

3306

7289

0
0
7289
0

0
0
3306
0

11639

25659

9094
0
0
0
954
476
2043
1860
3290
775
6822
355

4125
0
0
0
437
216
927
853
1492
333
3094
161

2773

6114

1175
300
1730
2377
532

533
136
745
1078
241

1286

2668

2510
150

1138
68

549

1211

1211

549

249

558

558

590

1308

1308

0

600

600

MM PERSONNEL PROVISIONS

DRY STRUCTURE

DESIGN RESERVE (CONTINGENCY)

PERSONNEL

PAYLOAD

CARGO OR UPPER STAGE MISSION EQUIPMENT/PAYLOAD

RESIDUAL PROPELLANT AND SERVICE ITEMS TANK PRESSURIZATION AND PURGE GASES

TRAPPED FUEL

TRAPPED OXIDIZER

SERVICE ITEMS RESIDUALS

RESERVE PROPELLANT AND SERVICE ITEMS

FUEL-MAIN PROPULSION

OXIDIZER-MAIN PROPULSION

POWER SOURCE PROPELLANTS

LUBRICANTS

ATTITUDE CONTROL PROPELLANTS

OPERATING WEIGHT EMPTY

IN-FLIGHT LOSSES

ICE AND FROST

FUEL VENTED

OXIDIZER VENTED

POWER SOURCE PROPELLANTS

ATTITUDE CONTROL PROPELLANTS

LUBRICANTS AND OTHER SERVICE ITEMS

FLYBACK FUEL

THRUST DECAY PROPELLANTS

PROPELLANTS

FUEL - MAIN

OXIDIZER - MAIN

FUEL - SECONDARY

OXIDIZER - SECONDARY

MASS AT FULL THRUST

PRE-IGNITION LOSSES

MM MINIMUM GROSS WEIGHT

1811

(60893)

6889

245

12004

12884

970

216

572

181

1471

210

1261

0

0

0

(82473)

2470

0

0

0

660

402

590

806

612

181034

24412

145475

1691

8456

266598

0

(266598)

1 GAREQ-3384398 MR=3.045 FLYBACK=2
VOLUMES

	CU FT	CU M
BCDY STRUCTURE	5316.49	150.46
CREW AND PASSENGER COMPARTMENTS	1500.00	42.45
CARGO COMPARTMENT	10600.00	299.96
LANDING GEAR BAYS	0.00	0.00
PROPULSION RAY WITHIN BODY	0.00	0.00
FUEL CONTAINERS	13517.59	382.55
OXYDIZER CONTAINERS	4852.45	137.32
PROPELLANT INSULATION	0.00	0.00
SECONDARY FUEL CONTAINERS	1285.57	36.38
SECONDARY OXYDIZER CONTAINERS	340.45	10.77
OTHER BODY VOLUME	31638.59	895.37
TOTAL ENTRY VEHICLE VOLUME	63091.2	1955.3

DESIGN DATA

	SQ. FT.	SQ.M.
WETTED AREAS		
GROSS BODY (SBODY)	11814.49	1097.57
FUEL TANKS (SFUTK)	4045.69	379.56
OXYDIZER TANKS (SOXTK)	8.00	0.00
PLAN AREAS		
THEORETICAL WING AREA (SWING)	1818.00	168.89
EXPOSED WING AREA (SXPOS)	1319.96	122.61
PLANFORM, INCL. ELEVONS (SPLAN)	2885.83	268.09
VERTICAL SURFACES (SVERT)	440.62	40.93
HORIZONTAL SURFACES (SHORZ)	867.50	82.45
UNIT WEIGHTS	LB/SQ.FT.	KG/SQ.M.
WING	7.70	35.64
VERTICAL SURFACES	5.16	25.19
HORIZONTAL SURFACES	7.34	35.82
WING LOADING	98.75	452.11

3A DIMENSIONAL DATA

	FEET	METERS
WING		
STRUCTURAL SPAN	112.57	34.31
ROOT CHORD LENGTH	27.15	7.06
THICKNESS RATIO		
BODY		
LENGTH (LBODY)	145.64	44.39
WIDTH (BBODY)	22.93	6.99
HEIGHT (HBODY)	29.35	8.95

THEORETICAL VELOCITY INCREMENT
FEET/SECOND METERS/SECOND

MASS RATIO DURING FLIGHT PHASE 1=	1.0000	0.0000	0.0	0.0	0.0
MASS RATIO DURING FLIGHT PHASE 2=	1.0000	0.0000	0.0	0.0	0.0
MASS RATIO DURING FLIGHT PHASE 3=	2.7855	6.0000	451.6	14884.5	4536.6
MASS RATIO DURING FLIGHT PHASE 4=	1.0156	6.0000	451.6	225.0	68.6
MASS RATIO DURING FLIGHT PHASE 5=	1.1186	5.0000	416.0	1500.0	457.2
MASS RATIO DURING FLIGHT PHASE 6=	1.0000	0.0000	0.0	0.0	0.0

POUNDS KILOGRAMS

MAXIMUM GROSS WEIGHT (MGROSS)

507747 265598

PRE-IGNITION LOSSES

9

MASS AT EMISSION

266598

FUEL
OXIYDIZFR
JETYSCH

424

END OF PRE-FLIGHT PHASE (WTO)

20597

FUEL
OXIDIZER
JETTYSON

[illegible]

WEIGHT IN ORBIT (WOBIT)
JETTYSON

95710
-3052

RETURN CONDITION (WEYURA)

92658

FUEL
OXIDIZER
JETTYSON

-1691
-8456
-1082

ENTRY WEIGHT (MENTRY)

81429

FLYBACK FUEL

- 886 -

LANDING HEIGHT (WLAND)

80622

WEIGHT COEFFICIENTS * C(II),

1	1624.00000
4	1.19700
6	.00590
10	.63700
12	.58400
13	3.07000
23	.15000
29	.10000
30	.00416
37	4802.00000
43	.50000
45	.00200
48	.00350
51	.04000
52	.04000
55	.07710
62	47.62700
63	.47300
64	608.00000
65	.32600
66	.64900
68	950.00000
70	1700.00000
72	800.00000
74	.00100
75	700.00000
76	150.00000
96	.19000
97	270.00000
103	20229.05270
114	400.00000
127	1500.00000
129	47000.00000
131	1300.00000
135	1.24000
138	.53400
153	.00200
159	.17300
156	285.00000
160	.75000
161	300.00000
164	.60000
166	.00143
168	.00400
169	800.00000
170	.75000
171	1.25000
174	.00420
176	1.00000
181	.91200
182	1.12400
183	.00610
185	1.00000
210	.03800
212	.20000
214	.01000
220	100.00000
221	.50000
225	.00110
226	.00006

227	.00050
228	.00114

VOLUME COEFFICIENTS = K(I), I = 1,30 =

[illegible]

1 5MREQ=3384390 MR=3.845 FLYBACK=2

CE000V	=	WIDTH OF BODY /	VBODY **	.3333	=	.5586	
CM000V	=	WEIGHT OF BODY /	VBODY**	.3333	=	.7154	
CL000V	=	LENGTH OF BODY /	VBODY**	.3333	=	3.5443	
CS000V	=	WETTED AREA OF BODY /	VBODY **	.6667	=	7.0167	
CF000V	=	FAIRING AREA /	BODY WETTED AREA	=	0.0000		
CF001K	=	WETTED AREA OF FUEL TANKS /	TANK VOLUME **	.6667	=	7.2000	
CF002K	=	STABILIZER AREA/THO WING AREA	=	.4467			
CS001K	=	WETTED AREA OF OXIDIZER TANKS /	TANK VOLUME **	.6667	=	0.0020	
CS002K	=	VEHICLE PLATFORM AREA /	BODY VOLUME **	.6667	=	1.7139	
CS003K	=	AREA OF EACH VERTICAL	SURFACE /	VBODY **	.6667	=	.2617
CS004K	=	ENGINE THRUST /	WTO	=	0.0000		

CTHST2 = AIRBREATHING ENGINE THRUST / WENTRY = 0.0000

NCREW = NUMBER OF CREW = 2

NEMGS = NUMBER OF ENGINES = 2

RHOFU = FUEL DENSITY = 4.40

RHOFU2 = SECONDARY PROPELLSION FUEL DENSITY = 2.90

RHOX = OXIDIZER DENSITY = 71.00

RHOX2 = SECONDARY PROPELLSION OXIDIZER DENSITY = 49.00

TTOT = MAIN PROPELLSION THRUST = CTHST2 * WCRB17 + C(129) * NEMGS =

0.000 + 587746 + 47000 * 2 = 940000

TTOT2 = AIRBREATHING ENGINE THRUST = CTHST2 * WETURN C(150) = *

0.000 + 179519 + 0 = 0

WJET(1) = 0.0

WJET(2) = 0.0

WJET(3) = 0.0

WJET(4) = 6729.2

WJET(5) = 2386.2

WJET(6) = 0.0

NR 1 CREW=3380390 MW=3.045 FLVACK=2

WEIGHT BREAKDOWN - ROOST STAGE

AERODYNAMIC SURFACES
WING + WING MOUNTED CONTROL SURFACES
HORIZONTAL SURFACES
VERTICAL SURFACES
FAIRINGS, SHROUDS AND ASSOCIATED STRUCTURE

800V STRUCTURE 137037 64994 29481

STRUCTURAL FUEL CONTAINERS
STRUCTURAL OXIDIZER CONTAINERS
BASIC BODY STRUCTURE
SECONDARY STRUCTURE
THRUST STRUCTURE

INDUCED ENVIRONMENTAL PROTECTION 32539 14759 14750

COVER PANELS, NON-STRUCTURAL
VEHICLE INSULATION

LAUNCH, RECOVERY AND DOCKING 21325 9673

LAUNCH GEAR
DEPLOYABLE AERODYNAMIC DEVICES
ALIGNING GEAR
DOCKING STRUCTURE

PROPULSION 131477 59637

ENGINES AND ACCESSORIES
SECONDARY ENGINES AND ACCESSORIES
FUEL CONTAINERS AND SUPPORTS (NON-STRUCTURAL)
OXIDIZER CONTAINERS AND SUPPORTS (NON-STRUCTURAL)
SECONDARY FUEL TANKAGE AND SYSTEMS
SECONDARY OXIDIZER TANKAGE AND SYSTEMS
PROPELLANT INSULATION
FUEL SYSTEM - MAIN
OXIDIZER SYSTEM - MAIN
PURGE SYSTEMS
AIRBRAKING ENGINES AND INSTALLATION
MAGNETS, PODS, PYLONS, SUPPORTS
AIRBRAKING PROPULSION TANKAGE AND SYSTEMS

ORIENTATION, SEPARATION AND ULLAGE CONTROL 21069 9564

SEPARATION SYSTEMS
STABILITY AND CONTROL, ENGINE RIMBAL
AERODYNAMIC CONTROLS
SPATIAL ATTITUDE CONTROL SYSTEM
CONTROL PROPELLANT TANKAGE AND SYSTEMS

PRIME POWER SOURCE AND DISTRIBUTION 2764 1263

POWER SOURCE UNITS
POWER SOURCE TANKAGE AND SYSTEMS

POWER CONVERSION AND DISTRIBUTION 3622 1643

HYDRAULIC/PNEUMATIC

GUIDANCE AND NAVIGATION 950 249

INSTRUMENTATION 2000 907

COMMUNICATION 350 8

PERSONNEL PROVISIONS	1649	748	
DRY STRUCTURE	(419412)	(190242)	
DESIGN RESERVE (CONTINGENCY)	41941	19024	
PERSONNEL	466	109	
PAYLOAD	567747	266598	
CARGO OR UPPER STAGE	567747	266598	
MISSION EQUIPMENT/PAYLOAD	0	0	
RESIDUAL PROPELLANT AND SERVICE ITEMS	9038	4100	
TANK PRESSURIZATION AND PURGE GASES	0	0	
TRAPPED FUEL	3253	1475	
TRAPPED OXIDIZER	5655	2565	
SERVICE ITEMS RESIDUALS	130	59	
RESERVE PROPELLANT AND SERVICE ITEMS	1378	621	
FUEL-MAIN PROPULSION	1378	621	
OXIDIZER-MAIN PROPULSION	0	0	
POWER SOURCE PROPELLANTS	0	0	
LUBRICANTS	0	0	
ATTITUDE CONTROL PROPELLANTS	0	1	
OPERATING WEIGHT EMPTY	(1059968)	(480794)	
IN-FLIGHT LOSSES	42807	19417	
ICE AND FROST	0	0	
FUEL VENTED	0	0	
OXIDIZER VENTED	0	0	
POWER SOURCE PROPELLANTS	159	72	
ATTITUDE CONTROL PROPELLANTS	1571	713	
LUBRICANTS AND OTHER SERVICE ITEMS	130	59	
FLYBACK FUEL	40947	18573	
THRUST DECAY PROPELLANTS	8713	3952	
PROPELLANTS	2272997	1031014	
FUEL - MAIN	324714	147286	
OXIDIZER - MAIN	1948283	883726	
FUEL - SECONDARY	0	0	
OXIDIZER - SECONDARY	0	0	
MASS AT FULL THRUST	3384486	1535177	
PRE-IGNITION LOSSES	9	9	
MAXIMUM GROSS WEIGHT	(3384486)	(1535177)	
GROSS WEIGHT OF ONE BOOSTER	2796739	1268579	

NR 1 GUREQ-338439J HP=3.845 FLYBCK=2

VOLUMES

	CU FT	CU M
ACDY STRUCTURE	14967.76	423.59
CREW AND PASSENGER COMPARTMENTS	250.00	7.07
CARGO COMPARTMENT	0.00	0.00
LANDING GEAR BAYS	0.00	0.00
PROPULSION BAY WITHIN BODY	0.00	0.00
FUEL CONTAINERS	88194.95	2495.92
OXYDIZER CONTAINERS	29006.72	820.89
PROPELLANT INSULATION	0.00	0.00
SECONDARY FUEL CONTAINERS	0.00	0.00
SECONDARY OXYDIZER CONTAINERS	0.00	0.00
OTHER BODY VOLUME	57849.32	1637.14
TOTAL ENTRY VEHICLE VOLUME	198266.4	5784.6

DESIGN DATA

WETTED AREAS

	SQ. FT.	SQ.M.
GROSS BODY (SPODY)	21692.48	2015.22
FUEL TANKS (SFUTK)	12958.07	1203.80
OXYDIZER TANKS (SOXTK)	5192.32	482.37
PLAN AREAS	SQ. FT.	SQ.M.
THEORETICAL WING AREA (SWING)	4641.38	449.73
EXPOSED WING AREA (SXPOS)	3637.43	337.92
PLANFORM, INCL. ELEVONS (SPLAN)	7327.48	680.72
VERTICAL SURFACES (SVERT)	1007.06	93.56
HORIZONTAL SURFACES (SHOPZ)	2643.00	245.53

UNIT WEIGHTS

	LB/SQ.FT.	KG/SQ.M.
WING	8.64	42.18
VERTICAL SURFACES	6.29	30.72
HORIZONTAL SURFACES	6.37	31.09
WING LOADING	103.38	507.19

DIMENSIONAL DATA

	FEET	METERS
WING		
STRUCTURAL SPAN	183.69	55.99
ROOT CHORD LENGTH	39.48	12.03
THICKNESS RATIO		.14
BODY		
LENGTH (LBODY)	215.59	65.71
WIDTH (WBODY)	32.11	9.79
HEIGHT (HBODY)	31.61	9.63

1 SHRE0-3304390 HR=3.045 FLYBACK=2

	MASS RATIO DURING FLIGHT PHASE	1=	1.0000	O/F RATIO	ISP	THEORETICAL VELOCITY INCREMENT FEET/SECOND METERS/SECOND
	MASS RATIO DURING FLIGHT PHASE	2=	1.0000	0.0000	0.0	0.0
	MASS RATIO DURING FLIGHT PHASE	3=	3.0450	0.0000	420.7	15359.3
	MASS RATIO DURING FLIGHT PHASE	4=	1.0000	0.0000	0.0	0.0
	MASS RATIO DURING FLIGHT PHASE	5=	1.0000	0.0000	0.0	0.0
	MASS RATIO DURING FLIGHT PHASE	6=	1.0000	0.0000	0.0	0.0

WEIGHT SUMMARY

POUNDS KILOGRAMS

MAXIMUM GROSS WEIGHT (NGROSS)

3304406 1535177

PRE-IGNITION LOSSES

0

MASS AT IGNITION

3304406 1535177

FUEL

0

OXIDIZER

0

JETTISON

0

END OF PRE-FLIGHT PHASE (WTO)

3304407 1535170

FUEL

-324714 -147200

OXIDIZER

-1948203 -803726

PAYLOAD ADDED/REMOVED

-587747 -266598

INITIAL SEPARATION

523743 237566

THRUST DECAY AND RESIDUALS

-19121 -8673

RETURN CONDITION (RETURN)

504622 220892

JETTISON AND EXPENDABLES

-1730 -745

INITIAL FLYBACK

502031 220100

FLYBACK FUEL

-40947 -10573

LANDING WEIGHT (WLAND)

461944 209534

WEIGHT COEFFICIENTS = C(I),

1	1624.0000
2	1.19700
3	.00550
4	.03700
5	.00400
6	2.00000
7	.10000
8	.00916
9	4250.0000
10	.50000
11	.00150
12	.00200
13	.06000
14	.06000
15	.07710
16	.47.62700
17	.47300
18	700.00000
19	.32600
20	.84900
21	558.00000
22	2000.00000
23	350.00000
24	.03095
25	420.00000
26	352.00000
27	.10000
28	230.00000
29	587747.12063
30	130.00000
31	1370.00000
32	159.00000
33	455221.2080
34	120.00000
35	1.24000
36	.53400
37	.32000
38	.17300
39	285.00000
40	.75000
41	1950.00000
42	.60000
43	.00174
44	.00400
45	.00300
46	1.00000
47	1.50000
48	1.12400
49	.00010
50	1.00000
51	45000.00000
52	.20000
53	.00864
54	40.00000
55	.50000
56	.0010
57	.00015
58	.00019
59	.00095

VOLUME COEFFICIENTS = K(I), I = 1,30 =

6	250.00000	7	0.00000	8	0.00000	9	0.00000	10	0.00000
11	0.00000	12	0.00000	13	0.00000	14	0.00000	15	0.00000
16	0.00000	17	0.00000	18	0.00000	19	0.00000	20	0.00000
21	0.00000	22	0.00000	23	0.00000	24	0.00000	25	0.00000
26	0.00000	27	0.00000	28	9306.18982	29	0.00000	30	0.00000

1 GNR20=3304390 MR=3.045 FLYRCK=2

C*BODY = WIDTH OF BODY / VBODY ** .3333 = .5563
 C*BODY = HEIGHT OF BODY / VBODY** .3333 = .5496
 C*BODY = LENGTH OF BODY / VBODY** .3333 = 3.7463
 C*BODY = WETTED AREA OF BODY / VBODY ** .6667 = 6.5574
 C*FAIR = FAIRING AREA / BODY WETTED AREA = 0.0000
 C*FUTK = WETTED AREA OF FUEL TANKS / TANK VOLUME ** .6667 = 6.3400
 C*HOR2 = STARILIZER AREA/THEO WING AREA = .5460
 C*OXTK = WETTED AREA OF OXIDIZER TANKS / TANK VOLUME ** .6667 = 5.5000
 C*PLAM = VEHICLE PLANFORM AREA / BODY VOLUME ** .6667 = 2.2190
 C*VERT = AREA OF EACH VERTICAL SURFACE / VBODY ** .6667 = .3044
 C*VRST = ENGINE THRUST / WTO = 0.0000

CTWST2 = AIRBREATHING ENGINE THRUST / WENTRY = 0.0000

NCREW = NUMBER OF CREW = 2
NENGS = NUMBER OF ENGINES = 11
RHOFU = FUEL DENSITY = 4.48
RHOFP2 = SECONDARY PROPELLSION FUEL DENSITY = 0.00
RHOX = OXIDIZER DENSITY = 71.00
RHOX2 = SECONDARY PROPELLSION OXIDIZER DENSITY = 0.00
T10Y = MAIN PROPELLSION THRUST = CTHST + WORBIT + C(129) * NENGS =
0.000 + 3364487 + 455221 * 11 = 5087435

T1CT2 = AIRBREATHING ENGINE THRUST = CTWST2 * METURN C(158) = *
0.000 + 502091 + 0 = 0

WJET(1) = 0.0
WJET(2) = 0.0
WJET(3) = 587747.1
WJET(4) = 19121.7
WJET(5) = 1730.3
WJET(6) = 0.0

SPECIFIC FUEL CONSUMPTION = .30300
CRUISE BACK VELOCITY = 290.00 FT/SEC
THRUST RATIO = .96856
L/D = 0.50000
FLYBACK RANGE = 413.17 N. MI.
BANK ANGLE DURING REENTRY = 0.00 DEG

MINIMUM ASCENT DYNAMIC PRESSURE = 499.41 PSF
TIME AT MAX Q = 65.000 SEC
ALTITUDE AT MAX Q = 29353 FT
MACH NUMBER AT MAX Q = 1.019
RELATIVE VELOCITY AT MAX Q = 1040.60 FPS
RELATIVE FLIGHT PATH ANGLE AT MAX Q = 69.25 DEG

ORBITER MU = 2.78546 WEIGHT = 587747 THRUST = 949880 T/W = 1.599327
BOOSTER MU = 3.04508 WEIGHT = 3384486 THRUST = 5007435 T/W = 1.325282
FIRST CONTROL VAR = 1.75582783E+05

NUMBER OF TRAJECTORIES FOR THIS CASE = 1 TRAJECTORIES PER INSIDE ITERATION = 1
INSIDE ITERATIONS PER INTERMEDIATE ITERATION = 1 INTERMEDIATE ITERATIONS PER OUTSIDE ITERATION = 8
SIMULATION SECTION 1 PAYLOAD = 8.

SIMULATION SECTION 2

SIMULATION SECTION 3

BURNOUT WEIGHT = 1.11148862E+06, JETTISON WEIGHT = 5.23741490E+05, SUBSEQUENT WEIGHT = 5.07747124E+05

***** SUCCESSFUL CONVERGENCE IN XHEM1 *****
FIRST CONTROL VAR = 1.75507703E+00

NUMBER OF TRAJECTORIES FOR THIS CASE = 2

INSIDE ITERATIONS PER INTERMEDIATE ITERATION = 1

SIMULATION SECTION 1

TRAJECTORIES PER INSIDE ITERATION = 2

INTERMEDIATE ITERATIONS PER OUTSIDE ITERATION = 0

PAYLOAD = 0.

ABR TIME	REL TIME	ALTITUDE	LATITUDE	LONGITUDE	HEIGHT
REL VELOCITY	REL GAMMA	REL AZIMUTH	INPTL VELOCITY	INRTL GAMMA	INRTL AZIMUTH
HEATING PAR	HEAT FLUX	RADIUS	TARGET DIST	CROSS RANGE	DOWN RANGE
ROLL ANGLE	ALPHA	/AM ANGLE	PITCH ANGLE	WEIGHT DOT	ISP
C (YAW)	C (AXIAL)	C (NORMAL)	YAW F(AERO)	AXIAL F(AERO)	NORM F(AERO)
THRUST	THRUST LOAD	PITCH RATE	FORCE (AZMTH)	FORCE (VEL)	FORCE (GAMMA)
YAW AERO LO	AXIAL AERO LO	NORM AERO LO	AXIAL LOAD	TOTAL AERO LO	TOTAL LO
CYN PRESS	ATM PRESS	VEL SOUND	MACH NUMBR	MERIDIAN GRVY	RADIAL GRVY
TOTAL GRVY	SURF RADIUS	GRAVITY LOSS	DRAQ LOSS	MISALIGN LOSS	IDEAL VEL
0.	0.	0.	2.03200000E+01	2.79600000E+02	3.3644056AE+06
0.	9.0000000E+01	3.76500000E+01	1.34227615E+03	0.	9.00000000E+01
0.	0.	2.09090910E+07	0.	0.	0.
0.	0.	0.	0.	1.14481813E+04	3.9100E+00E+02
0.	0.	0.	0.	0.	0.
4.40539745E+06	1.32528185E+00	0.	0.	4.40539745E+06	0.
0.	0.	0.	1.32528185E+00	0.	1.32528185E+00
0.	1.47510000E+01	1.13000000E+03	0.	-4.37956260E-02	3.22120900E+01
3.2212095E+01	2.09090919E+07	0.	0.	0.	0.
0.	1.0000000E+01	5.50342763E+02	2.03199925E+01	2.79459993E+02	3.27000366E+06
1.12686640E+02	9.83713360E+04	3.76500000E+01	1.34657900E+07	4.79681251E+00	0.00240054E+01
3.89618300E+03	1.41511692E+03	2.09104423E+07	0.	0.	3.60000000E+02
0.	-3.73338760E-01	0.	0.	1.14481813E+04	3.92662551E+02
0.	-1.02122000E-01	0.	0.	-7.090904126E+03	0.
4.49527209E+06	1.37469932E+00	0.	0.	4.40000597E+06	-1.92447396E+04
0.	-2.16845042E-03	0.	1.37253007E+00	2.16845042E-03	1.37253007E+00
1.43829989E+01	1.44719762E+01	1.13563335E+03	9.9157520E-02	-4.37912030E-02	1.2210943F+01
3.22104241E+01	2.09090920E+07	3.22107600E+02	2.34496371E-01	7.69475061E-03	4.34092661E+02
0.	2.00000000E+01	2.30504929E+03	2.03199892E+01	2.79459991E+02	3.15552205E+06
2.41325179E+02	0.91851002E+01	3.76500000E+01	1.36600499E+03	1.01745213E+01	8.90041979E+01
7.13164502E+04	1.51350044E+04	2.09121978E+07	0.	0.	3.60000000E+02
0.	-1.29113362E+00	0.	2.10603340E+00	1.14481813E+04	3.95343254E+02
0.	-9.27000000E-02	0.	0.	-2.11469649E+04	0.

72

-1.01347585E+05
1.4253791E+00
3.22049839E+01
0.85669044E+02

4.49667324E+06
8.91962231E-03
-4.37768044E-02
5.57614714E-02

0.
1.43377911E+00
2.13857768E-01
1.05471150E+00

3.5105556E-01
0.
1.12843777E+03
6.44180397E+02

1.42429073E+00
-8.9162231E-03
1.36008048E+01
2.00096920E+07

4.52596126E+06
0.
6.27195405E+01
3.22050136E+01

3.04104024E+06
8.90777965E+01
0.
3.99747932E+02
0.

2.79460267E+02
1.58967856E+01
0.
1.14481813E+04
-6.64672957E+04

2.83203047E+01
1.41374857E+03
0.
4.08307386E+00
6.

5.43376947E+03
3.76500000E+01
1.09153254E+07
0.

3.00000000E+01
8.59169261E+01
5.75002127E+04
0.
-9.27000000E-02

3.00000000E+01
3.88218528E+02
4.02565950E+05
0.
0.

0.
1.40316928E+00
3.21953473E+01
1.35823557E+03

4.51037745E+06
2.18567630E-02
-4.37505947E-02
1.04263539E-01

0.
1.40316388E+00
1.47515689E-01
6.64092425E+00

3.08868960E-01
1.
1.11712510E+03
9.55842630E+02

1.50502604E+00
-2.10567630E-02
1.21670090E+01
2.00098916E+07

4.57684475E+06
0.
1.48133085E+01
3.21953770E+01

2.92655817E+06
8.74970686E+01
0.
4.05703653E+02
0.

2.79461206E+02
2.15946541E+01
0.
1.14481813E+04
-1.19156824E+05

2.83213741E+01
1.49631040E+03
0.
7.33942550E+00
0.

1.01107970E+04
3.77355082E+01
2.09200013E+07
0.

4.00000000E+01
0.20605745E+01
1.46504598E+05
0.
-9.343284E-02

4.00000000E+01
5.56126583E+02
1.37754286E+06
0.
0.

0.
1.54632589E+00
3.21809467E+01
1.85536271E+03

4.52541217E+06
4.07156827E-02
-4.37125168E-02
1.04263539E-01

0.
1.54632559E+00
5.04865395E-01
1.65290031E+01

4.33653439E-01
0.
1.10153437E+03
1.28599302E+03

1.56704127E+00
-4.07156827E-02
1.82533511E+01
2.09098905E+07

4.64456899E+06
0.
2.643748E+02
3.21809764E+01

2.81207661E+06
8.47963201E+01
0.
4.12507149E+02
0.

2.79463449E+02
2.66324245E+01
0.
1.14481813E+04
-2.28039196E+05

2.83239230E+01
1.62374193E+03
0.
1.28817071E+01
0.

1.64985221E+04
3.77328113E+01
2.09263864E+07
0.

5.00000000E+01
7.71187979E+01
2.90046059E+05
0.
-1.21658481E-01

5.00000000E+01
7.46472870E+02
1.51725479E+06
0.
0.

0.
1.5913712E+00
3.21612955E+01
2.38051363E+03

4.49361745E+06
8.13772978E-02
-4.36616680E-02
1.04263539E-01

0.
1.59797121E+00
6.91479677E-01
7.57175913E+01

5.30496299E-01
0.
1.07898562E+03
1.60245569E+03

1.67934850E+00
-8.13772978E-02
0.05250977E+00
2.09098807E+07

4.72245605E+06
0.
3.8855933E+02
3.21613251E+01

2.67759400E+06
8.10769652E+01
0.
4.19431719E+02
0.

2.79467849E+02
3.01964707E+01
0.
1.14481813E+04
-6.47680472E+05

2.83293294E+01
1.79115089E+03
0.
1.06184757E+01
0.

2.46714949E+04
3.77058312E+01
2.09345543E+07
0.

6.00000000E+01
7.13818243E+01
4.61904515E+05
0.
-2.77354409E-01

6.00000000E+01
9.50845338E+02
7.29342316E+06
0.
0.

0.
1.33990878E+00

4.15484991E+05
2.40095537E-01

9.
1.53990878E+00

6.00445078E-01
0.

1.78000041E+00
-2.40095537E-01

4.80173038E+06
0.

4.05005216E+02	5.01250257E+00	1.04391112E+03	9.10657731E-01	-4.3304490E-02	8.81301790E+01
3.21362099E+01	2.09098820E+07	1.9119276E+03	7.91107920E+01	1.04263539E-01	2.93609299E+03
***	7.00000000E+01	3.43210931E+04	2.83737310E+01	2.79475203E+02	2.58311298E+06
1.18468888E+03	6.4358911E+01	3.76851187E+01	1.98342486E+03	3.12571655E+01	7.70177423E+01
1.54460006E+07	5.93134262E+05	2.79441953E+07	0.	0.	0.
0.	0.	0.	2.50641089E+01	1.14481013E+04	4.25551172E+02
0.	-4.70759268E-01	0.	0.	-1.12081032E+06	0.
4.87178699E+06	1.08601390E+00	6.7496597E-01	0.	3.75097667E+06	0.
0.	-4.33899070E-01	0.	1.45211433E+00	4.3399070E-01	1.45211433E+00
4.91810911E+02	3.83293987E+00	9.96049487E+02	1.18243382E+00	-4.35264248E-02	3.21065665E+01
3.21065960E+01	2.09098742E+07	2.21019040E+03	1.97127306E+02	1.04263539E-01	3.52654401E+03
***	8.00000000E+01	4.50722460E+04	2.83500165E+01	2.79486353E+02	2.46863117E+06
1.33766812E+03	5.78770132E+01	3.76769990E+01	2.10336733E+03	3.12564469E+01	7.2450027E+01
1.82846787E+07	6.13135880E+05	2.09549334E+07	0.	0.	3.79344614E-02
0.	0.	0.	3.21229868E+01	1.14481013E+04	4.30280312E+02
0.	-4.00822235E-01	0.	0.	-8.89396146E+05	0.
4.92881083E+06	1.99544537E+00	7.10680720E-01	0.	4.03662249E+06	0.
0.	-3.60379071E-01	0.	1.43516630E+00	3.60279071E-01	1.63516630E+00
4.58361736E+02	2.30053715E+00	9.51456674E-02	1.40591566E+00	-4.3496943E-02	3.20736238E+01
3.20736533E+01	2.09098612E+07	2.49202755E+03	3.27217086E+02	1.04263539E-01	4.15088879E+03
***	9.00000000E+01	5.70598831E+04	2.83687254E+01	2.79502772E+02	2.35414936E+06
1.63304717E+03	5.07379244E+01	3.76800039E+01	2.46525400E+03	3.05013707E+01	6.75239665E+01
2.43475173E+07	6.25360111E+05	2.09669118E+07	0.	0.	6.15729383E-02
0.	0.	0.	3.2620756E+01	1.14481013E+04	4.33528667E+02
0.	-3.62349323E-01	0.	0.	-6.71720024E+05	0.
4.98311400E+06	2.10244126E+00	7.04140632E-01	0.	4.29138597E+06	0.
0.	-2.05338236E-01	0.	1.02290302E+00	2.05338236E-01	1.82290302E+00
3.82940630E+02	1.25372532E+00	9.37609184E+02	1.74171414E+00	-4.33687469E-02	3.20369256E+01
3.20369549E+01	2.09098420E+07	2.75235086E+03	4.32662247E+02	1.04263539E-01	4.81093102E+03
***	1.00000000E+02	7.04242052E+04	2.83954013E+01	2.79526195E+02	2.23966754E+06
2.01629659E+03	4.40026879E+01	3.76938643E+01	2.87466008E+03	2.91086128E+01	6.27953642E+01
3.85651393E+07	5.76626414E+05	2.09800738E+07	0.	0.	9.52815337E-02
0.	9.	0.	4.59973121E+01	1.14481013E+04	4.35402317E+02
0.	-3.33545020E-01	0.	0.	-4.61774208E+05	0.
4.98456468E+06	2.22558241E+00	6.49291310E-01	0.	4.52279047E+06	0.
0.	-2.06179801E-01	0.	2.01940261E+00	2.06179801E-01	2.01940261E+00
2.85982934E+02	6.48224163E-01	9.62298544E+02	2.0953097E+00	-4.32846220E-02	3.15961566E+01
3.19961859E+01	2.00098146E+07	2.98782950E+03	5.10736388E+02	1.04263539E-01	5.50794897E+03

***	1.1000000E+02	1.1000000E+02	8.5123656E+04	2.04328811E+01	2.79558433E+02	2.12510573E+06
	2.4074302E+03	3.7944764E+01	3.7716792E+01	3.35120610E+03	2.71350972E+01	5.86422923E+01
	3.6845652E+07	5.18067755E+05	2.09949805E+07	0.	0.	1.41642878E-01
	0.	0.	0.	5.20523354E+01	1.14481813E+04	4.36402835E+02
	0.	-2.99674143E-01	0.	0.	-3.02147227E+05	0.
	4.9768165E+06	2.3500611E+00	5.75650259E-01	0.	4.69386928E+06	0.
	0.	-1.42174504E-01	0.	2.20861662E+08	1.42174504E-01	2.20868662E+00
	2.0273617E+02	3.22634485E-01	9.82926022E+02	2.5306642E+08	-4.31994059E-02	3.19513776E+01
	3.19514060E+01	2.09097769E+07	3.19736920E+03	5.66274173E+02	1.04263539E-01	6.24390209E+03

***	1.2000000E+02	1.2000000E+02	1.01045371E+05	2.04007245E+01	2.79601246E+02	2.01070392E+06
	3.04292614E+03	3.26466902E+01	3.77499972E+01	1.91116350E+03	2.40149846E+01	5.52044995E+01
	4.0869996E+07	4.43089570E+05	2.10107722E+07	0.	0.	2.03149219E-01
	0.	0.	0.	5.73537098E+01	1.14481813E+04	4.36910757E+02
	0.	-2.79552911E-01	0.	0.	-1.97066015E+05	0.
	5.0107938E+06	2.40762602E+00	4.99630105E-01	0.	4.04081256E+06	0.
	0.	-9.4000603E-02	0.	2.30961715E+00	9.0008063E-02	2.30931715E+00
	1.4561777E+02	1.56969446E-01	1.00301579E+03	3.03367722E+08	-4.31159660E-02	3.19029890E+01
	3.19020102E+01	2.09097260E+07	3.38179362E+03	6.04692156E+02	1.04263539E-01	7.02191690E+03

***	1.2600000E+02	1.2600000E+02	1.11106336E+05	2.05164183E+01	2.79632708E+02	1.94201522E+06
	3.43554058E+03	2.90220736E+01	3.77746902E+01	4.28590910E+03	2.33403791E+01	5.34717014E+01
	4.33750724E+07	3.92109829E+05	2.10207964E+07	0.	0.	2.48299406E-01
	0.	0.	0.	6.01770264E+01	1.14460770E+04	4.37087017E+02
	0.	-2.66742548E-01	0.	0.	-1.48243152E+05	0.
	5.00226131E+06	2.7633470E+00	4.56391900E-01	0.	4.05503016E+06	0.
	0.	-7.63346906E-02	0.	2.50000000E+08	7.63346996E-02	2.50000000E+00
	1.14001421E+02	1.01245021E-01	1.01937014E+03	3.35393631E+09	-4.30680966E-02	3.18728733E+01
	3.10725024E+01	2.09096901E+07	3.43110904E+03	6.21449150E+02	1.04263539E-01	7.51063400E+03

SIMULATION SECTION 2

ABS TIME	REL TIME	ALTITUDE	LATITUDE	LONGITUDE	WEIGHT
REL VELOCITY	REL GAMMA	REL AZIMUTH	INR/L VELOCITY	INR/L GAMMA	INR/L AZIMUTH
HEATING PAR	HEAT FLUX	RADIUS	TARGET DIST	CROSS RANGE	DOWN RANGE
ROLL ANGLE	ALPHA	YAW ANGLE	PITCH ANGLE	WEIGHT DOT	ISP
C (YAW)	C (AXIAL)	C (NORMAL)	YAW F(AERO)	AXIAL F(AERO)	NORM F(AERO)
THRUST	THRUST LOAD	PITCH RATE	FORCE (AZIMTH)	FORCE (VEL)	FORCE (GAMMA)
YAW AERO LD	AXIAL AERO LD	NORM AERO LD	AXIAL LOAD	TOTAL AERO LD	TOTAL LD
DYN PRESS	ATH PRESS	VEL SOUND	MACH NUMBER	MERIDIAN GRVY	RADIAL GRVY
TOTAL GPVY	SURF RADIUS	GRAVITY LOSS	DRAG LOSS	MISALIGN LOSS	IDEAL VFL
*** 1.26000000E+02	0.	1.1106336E+05	2.05164103E+31	2.79632700E+02	1.94201526E+06
3.41554650E+03	2.98229736E+01	3.77746902E+01	4.20598910E+03	2.33483791E+01	5.34717014E+01
4.33750724E+07	7.92109829E+05	2.10207964E+07	0.	0.	2.48299406E+01
0.	0.	0.	6.01770764E+01	1.14468770E+04	4.37087017E+02
0.	-2.66742586E-01	0.	0.	-1.48243152E+05	0.
5.00328131E+06	2.57633470E+60	0.	0.	4.85503816E+06	0.
0.	-7.63346996E-02	0.	2.50000000E+00	7.63346996E-02	2.50000000E+00
1.14001421E+02	1.01245821E-01	1.01037011E+03	3.3593631E+00	-4.30680966E-02	7.1874733E+01
3.1875024E+01	2.09096901E+07	3.48110984E+03	6.21449150E+02	1.04263539E-01	7.51063408E+03
*** 1.42000000E+02	1.60000000E+01	1.39208892E+05	2.86379334E+01	2.79740105E+02	1.76870816E+06
4.47499474E+03	2.34191018E+01	3.74581613E+01	5.35071574E+03	1.94148040E+01	5.80251908E+01
4.85316612E+07	2.55497272E+05	2.10487736E+07	0.	0.	4.02120236E-01
0.	0.	0.	6.69808982E+01	1.07611025E+04	4.37301160E+02
0.	-2.36697221E-01	0.	0.	-6.54217232E+04	0.
4.48719212E+06	2.57698882E+00	3.66456963E-01	0.	4.42177840E+06	0.
0.	-3.69884218E-02	0.	2.50000000E+00	3.69884218E-02	2.50000000E+00
5.70944296E+01	3.19709881E-02	1.06317304E+03	4.20809350E+00	-4.23553205E-02	3.17074669E+01
3.17875159E+01	2.09095647E+07	3.70870992E+03	6.48222296E+02	1.84263539E-01	8.82556919E+03
*** 1.32000000E+02	5.60000000E+01	2.898047597E+05	2.91183589E+01	2.80169056E+02	1.40480780E+06
7.31259589E+03	1.22635863E+01	3.81774098E+01	8.49556393E+03	1.09250634E+01	4.57288927E+01
5.46442255E+07	8.20211557E+04	2.11171135E+07	0.	0.	1.01194773E+00
0.	0.	0.	7.77364138E+01	8.85683984E+03	4.37393411E+02
0.	-7.28888000E-01	0.	0.	-1.19891519E+04	0.
3.52488866E+06	2.58853437E+00	2.27418079E-01	0.	3.51201951E+06	0.
0.	-6.53437164E-03	0.	2.50000000E+00	8.53437164E-03	2.50000000E+00
1.12168212E+01	2.13141577E-03	1.01206576E+03	7.22581577E+00	-4.28464644E-02	3.15809034E+01
3.15093325E+01	2.09090659E+07	4.04811679E+03	6.73490066E+02	1.04263539E-01	1.20670019E+04

000

2.22901239E+02
1.04114147E+04
5.05293264E+07
0.
0.
2.70115335E+06
0.
2.28060877E+00
3.14250951E+01

9.69012394E+01
5.11542107E+00
2.37451132E+04
0.
-2.20000000E-01
2.50219325E+00
-2.19327775E-03
1.72233025E-04
2.09002664E+07

2.68545572E+05
3.06673174E+01
2.11600120E+07
0.
0.
1.39157464E-01
0.
9.04376674E+02
4.27017902E+03

2.90702434E+01
1.12930701E+04
0.
4.40045709E+01
0.
0.
2.50000000E+00
1.14615610E+01
6.79420912E+02

2.000661620E+02
4.71511504E+00
0.
6.35839675E+03
-2.43700325E+03
2.77072155E+06
2.19327775E-03
-4.31014667E-02
1.04263539E-01

1.11140862E+06
4.39951877E+01
1.90100095E+00
4.37399456E+02
0.
0.
2.50000000E+00
3.14250655E+01
1.53620609E+04

SIMULATION SECTION 3

ABS TIME	REL TIME	ALTITUDE	LATITUDE	LONGITUDE	WEIGHT
REL VELOCITY	REL AZIMUTH				
HEATING PAR	RADIUS				
ROLL ANGLE	YAW ANGLE				
C (YAW)	C (NORMAL)				
THRUST	PITCH RATE				
YAW AERO LO	NORM AERO LO				
DYN PRESS	VEL SOUND				
TOTAL GRVTV	GRAVITY LOSS				
0.00000000E+00	0.00000000E+00	2.60545572E+05	2.98762434E+01	2.80861626E+02	1.11148862E+06
1.04114147E+04	3.86673174E+01	3.86673174E+01	1.12930701E+04	4.71511504E+00	4.39951877E+01
5.65293264E+07	2.11608120E+07	0.00000000E+00	0.00000000E+00	0.00000000E+00	1.98188095E+00
0.00000000E+00	0.00000000E+00	0.00000000E+00	1.48845789E+01	0.00000000E+00	0.00000000E+00
0.00000000E+00	0.00000000E+00	0.00000000E+00	0.00000000E+00	-2.43780325E+03	0.00000000E+00
0.00000000E+00	0.00000000E+00	0.00000000E+00	0.00000000E+00	-2.43780325E+03	0.00000000E+00
0.00000000E+00	0.00000000E+00	0.00000000E+00	-2.19327775E-03	2.19327775E-03	2.19327775E-03
2.28068077E+00	9.08376674E+02	9.08376674E+02	1.14615610E+01	-4.3101867E-02	3.14250655E+01
3.44250991E+01	4.27817902E+03	4.27817902E+03	6.79420912E+02	1.04263539E-01	1.53626689E+04
0.00000000E+00	0.00000000E+00	2.63285717E+05	2.99833577E+01	2.8092 341E+02	1.11148862E+06
1.04029613E+04	3.87095073E+01	3.87095073E+01	1.12953472E+04	4.32959821E+00	4.40322635E+01
5.65961032E+07	2.11714824E+07	0.00000000E+00	0.00000000E+00	0.00000000E+00	2.06606958E+00
0.00000000E+00	0.00000000E+00	0.00000000E+00	8.53023698E+01	0.00000000E+00	0.00000000E+00
0.00000000E+00	0.00000000E+00	0.00000000E+00	0.00000000E+00	-2.13768572E+03	0.00000000E+00
0.00000000E+00	1.39468758E-01	1.39468758E-01	0.00000000E+00	-2.13768572E+03	0.00000000E+00
0.00000000E+00	0.00000000E+00	0.00000000E+00	-1.92319173E-03	1.92319173E-03	1.92319173E-03
1.99983171E+00	1.4945988E-04	9.02885424E+02	1.15219063E+01	-4.31364194E-02	3.14169889E+01
3.14178166E+01	2.09881967E+07	4.28634204E+03	6.79619442E+02	1.04263539E-01	1.53626689E+04

RUMOUT WEIGHT = 1.11148862E+06, JETTISON WEIGHT = 5.23741498E+05, SURSEQUENT WEIGHT = 5.87747129E+05

SIMULATION SECTION 4

PAYLOAD = 0.

ANS TIME	REL TIME	ALTITUDE	LATITUDE	LONGITUDE	WEIGHT
REL VELOCITY	REL GAMMA	REL AZIMUTH	INRNL VELOCITY	INRNL GAMMA	INRNL AZIMUTH
HEATING PAR	HEAT FLUX	RADIUS	TARGET DIST	CROSS RANGE	DOWN RANGE
ROLL ANGLE	ALPHA	YAW ANGLE	PITCH ANGLE	HEIGHT DDT	TSP
C (YAW)	C (AXIAL)	C (NORMAL)	YAW F(AERO)	AXIAL F(AERO)	NORM F(AERO)
THRUST	THRUST LOAD	PITCH RATE	FORCE (AZIMTH)	FORCE (VEL)	FORCE (GAMMA)
YAW AERO LD	AXIAL AERO LD	NORM AERO LD	AXIAL LOAD	TOTAL AERO LD	TOTAL LD
DNV PRESS	ATH PRESS	VEL SOUND	MACH NUMBER	MERIDIAN GRVY	RADIAL GRVY
TOTAL GPVY	SUPP RADIUS	GRAVITY LOSS	DRAG LOSS	MISALIGN LOSS	IDEAL VEL

*** 2.25981239E+02	0.	2.63785717E+05	2.99439577E+01	2.80922341E+02	5.07747129E+05
1.84824613E+04	4.69763019E+00	3.87095833E+01	1.12853472E+04	4.32959821E+00	4.40322639E+01
5.65561032E+07	2.88041710E+04	2.11714024E+07	0.	0.	2.06806958E+00
0.	0.	0.	6.53023699E+01	0.	0.
0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.
1.99983171E+00	1.49445980E-04	9.02885424E+02	1.15219063E+01	-4.31364194E-02	3.14169369E+01
3.14170166E+01	2.09081967E+07	4.28634204E+03	6.79619442E+02	1.04263539E-01	1.53628689E+04

*** 2.29981239E+02	4.00000000E+00	2.66563166E+05	3.00314067E+01	2.81003413E+02	5.87747129E+05
1.83938280E+04	4.13963646E+00	3.87663408E+01	1.12763885E+04	3.81487561E+00	4.40817489E+01
5.66723864E+07	1.76046491E+04	2.11746886E+07	0.	0.	2.17823346E+00
0.	0.	0.	8.58603635E+01	0.	0.
0.	0.	0.	0.	0.	0.
0.	0.	1.39681021E-01	0.	0.	0.
1.69389144E+00	1.24940540E-04	8.96150681E+02	1.15974833E+01	-4.31054038E-02	3.14072772E+01
3.14073069E+01	2.09081037E+07	4.28615637E+03	6.79619442E+02	1.04263539E-01	1.53628689E+04

FIRST CONTROL VAR = -1.42688325E+01, SECOND CONTROL VAR = 9.65548799E+01

NUMBER OF TRAJECTORIES FOR THIS CASE = 1 TRAJECTORIES PER INSIDE ITERATION = 1

INSIDE ITERATIONS PER INTERMEDIATE ITERATION = 1 INTERMEDIATE ITERATIONS PER OUTSIDE ITERATION = 1

SIMULATION SECTION 5 PAYLOAD = 0.

SIMULATION SECTION 6

SIMULATION SECTION 7

***** SUCCESSFUL CONVERGENCE IN XMEM4 *****
FIRST CONTROL VAR = -1.4260035E+01, SECOND CONTROL VAR = 9.8554879E+01

NUMBER OF TRAJECTORIES FOR THIS CASE = 2 TRAJECTORIES PER INSIDE ITERATION = 2
INSIDE ITERATIONS PER INTERMEDIATE ITERATION = 1 INTERMEDIATE ITERATIONS PER OUTSIDE ITERATION = 1
SIMULATION SECTION 5 PAYLOAD = 0.

ABS TIME	REL TIME	ALTITUDE	LATITUDE	LONGITUDE	WEIGHT
REL VELOCITY	REL AZIMUTH	INRTL VELOCITY	INRTL GAMMA	INRTL AZIMUTH	
HEATING PAR	RADIUS	TARGET DIST	CROSS RANGE	ODMM RANGE	
ROLL ANGLE	YAW ANGLE	PITCH ANGLE	WEIGHT DOT	ISP	
C (YAW)	C (NORMAL)	YAW F (AERO)	AXIAL F (AERO)	NORM F (AERO)	
THRUST	PITCH RATE	FORCE (AZMTH)	FORCE (VEL)	FORCE (GAMMA)	
YAW AERO LD	NORM AERO LD	AXIAL LOAD	TOTAL AERO LD	TOTAL LD	
ATM PRESS	VEL SOUND	MACH NUMBER	MERIDIAN GRVY	RADIAL GRVY	
TOTAL GRVY	GRAVITY LOSS	DRAW LOSS	MISALIGN LOSS	IDEAL VEL	

2.24901239E+02	0.	2.65583168E+05	3.00314867E+01	2.81003413E+02	5.87747129E+05
1.03030208E+04	4.13963646E+08	3.87663408E+01	1.5E+04	3.81487561E+00	4.40017489E+01
5.66723864E+07	1.76046491E+04	2.11746869E+07	0.	0.	2.17828346E+00
0.	1.42600371E+01	0.	7.6903304E+01	2.08148804E+03	4.51598191E+02
0.	0.	0.	0.	0.	0.
9.39996234E+05	1.59932004E+08	-1.42600331E+01	0.	9.11032674E+05	2.31542702E+05
0.	0.	0.	1.59932004E+00	0.	1.59932004E+00
1.63289144E+00	1.24940508E-04	8.96150681E+02	1.15978033E+01	-4.31854038E-02	3.14072772E+01
3.14073069E+01	2.09081037E+07	4.29615663E+03	6.79619442E+02	1.04263539E-01	1.53628669E+04

2.91901239E+02	6.20000000E+01	2.93791759E+05	3.15894133E+01	2.82487153E+02	4.58694870E+05
1.38126380E+04	1.09524469E+08	3.97944539E+01	1.46904387E+04	1.02979305E+00	4.37454328E+01
5.73043367E+07	6.91828124E+03	2.12052134E+07	0.	0.	4.19093740E+00
0.	1.18965393E+01	0.	7.70082160E+01	2.08148804E+03	4.51599700E+02
0.	0.	0.	0.	0.	0.
9.39999375E+05	2.04929123E+08	8.83465689E-02	0.	9.19809541E+05	1.93776240E+05
0.	0.	0.	2.04929123E+00	0.	2.04929123E+00
5.00666035E-01	2.07205440E-03	8.594E+02	1.54856679E+01	-4.41981312E-02	3.13133235E+01
3.13133546E+01	2.09064217E+07	4.3566E+03	6.79619442E+02	9.62659204E+01	1.89649748E+04

3.03901239E+02	8.00000000E+01	3.07167815E+05	3.21261961E+01	2.83016994E+02	4.21228885E+05
1.50170136E+04	6.09840479E-01	4.01566428E+01	1.58931657E+04	5.76220245E-01	4.37678076E+01
5.74282055E+07	7.02462666E+03	2.12089989E+07	0.	0.	4.89139444E+00
0.	1.07619993E+01	0.	7.86281602E+01	2.08148804E+03	4.51599758E+02
0.	0.	0.	0.	0.	0.

9.39999496E+05
0.
4.67777870E-01
3.13009274E+01

2.23156090E+00
0.
1.67212209E-05
2.00058311E+07

9.09255119E-02
0.
9.01121675E+02
6.30919903E+03

0.
2.23156090E+00
1.6692551E+01
6.79619442E+02

9.23466139E+05
0.
-4.45784793E-02
1.20437339E+02

1.75525902E+05
2.23156090E+00
3.1300956E+01
2.02030699E+04

SIMULATION SECTION 6

1031

ABS TIME	REL TIME	ALTITUDE	LATITUDE	LONGITUDE	WEIGHT
REL VELOCITY HEATING PAR C (YAW)	REL GAMMA HEAT FLUX ALPHA C (AXIAL)	REL AZIMUTH RADIUS YAW ANGLE C (NORMAL)	INRTL VELOCITY TARGET DIST PITCH ANGLE YAW F(AERO)	INRTL GAMMA CROSS RANGE WEIGHT DOT AXIAL F(AERO)	INRTL AZIMUTH DOWN RANGE TSP NORM F(AERO)
THRUST YAW AERO LO OWN PR-55 TOTAL GRVTV	THRUST LOAD AXIAL AERO LO ATH PRESS SURF RADIUS	PITCH RATE NORM AERO LO VEL SOUND GRAVITY LOSS	FORCE (AZMTH) AXIAL LOAD MACH NUMMR DRAG LOSS	FORCE (VEL) TOTAL AERO LO MERIDIAN GRVTV MISALIGN LOSS	FORCE (GAMMA) TOTAL LO RADIAL GRVTV IDEAL VEL
3.09901239E+02	0.	3.03167815E+05	3.21261961E+01	2.83016994E+02	4.2128885E+05
1.50178136E+04	6.09840439E-01	4.01564287E+01	1.58931653E+04	5.76220245E-01	4.37678076E+01
5.74282055E+07	7.02462666E+03	2.12089989E+07	0.	0.	4.89139444E+00
0.	1.07619993E+01	0.	7.86281602E+01	2.08148004E+03	4.51599750E+02
0.	0.	0.	0.	0.	0.
9.39999496E+05	2.23156890E+00	0.	0.	9.23466139E+05	1.75525902E+05
0.	0.	0.	2.23156890E+00	0.	2.23156890E+00
4.57777870E-01	1.67212209E-05	9.04421675E+02	1.65929551E+01	-4.45704793E-02	3.13009956E+01
3.13009274E+01	2.89058311E+07	4.38919903E+03	6.79619442E+02	1.20437339E+02	2.02070699E+04
3.59901233E+02	5.00000000E+01	3.88042713E+05	3.33600407E+01	2.84798891E+02	3.17193683E+05
1.90010382E+04	-2.8629957E-02	4.13039575E+01	1.99539633E+04	-2.73786337E-02	4.41331328E+01
5.78512782E+07	1.04156244E+04	2.12119310E+07	0.	0.	7.18033326E+00
0.	6.81042888E+00	0.	8.32182019E+01	2.08148004E+03	4.51599015E+02
0.	0.	0.	0.	0.	0.
9.39999613E+05	2.96386157E+00	9.27930477E-02	0.	9.33366921E+05	1.11469562E+05
0.	0.	0.	2.96386157E+00	0.	2.96386157E+00
5.66803037E-01	1.28466521E-05	9.12073327E+02	2.09213861E+01	-4.57416250E-02	3.12881391E+01
3.12881725E+01	2.8903883E+07	4.39666610E+03	6.79619442E+02	1.59593581E+02	2.43264534E+04
3.62401239E+02	5.25000000E+01	3.88125682E+05	3.39571089E+01	2.84902205E+02	3.11953792E+05
1.93202171E+04	-3.87352844E-02	4.14337387E+01	2.01927780E+04	-3.51489688E-02	4.41641337E+01
5.78787817E+07	1.11640584E+04	2.12119035E+07	0.	0.	7.30988850E+00
0.	6.81621139E+00	0.	8.34505239E+01	2.07232453E+03	4.51599015E+02
0.	0.	0.	0.	0.	0.
9.35861376E+05	3.00000000E+00	9.29288841E-02	0.	9.29685063E+05	1.07341507E+05
0.	0.	0.	3.00000000E+00	0.	3.00000000E+00
5.77428222E-01	1.27407083E-05	9.12954614E+02	2.11785797E+01	-4.58039906E-02	3.12879870E+01
3.12881026E+01	2.89037778E+07	4.39670713E+03	6.79619442E+02	1.71233273E+02	2.45666511E+04

SIMULATION SECTION 7

1030

ABS TIME	REL VELOCITY	REL GAMMA	REL AZIMUTH	LATITUDE	LONGITUDE	WEIGHT
REL VELOCITY	REL GAMMA	REL AZIMUTH	REL VELOCITY	LONGITUDE	WEIGHT	
HEATING PAR	HEAT FLUX	RADIUS	TARGET DIST	INRVL GAMMA	INRVL AZIMUTH	DOWN RANGE
POLL ANGLE	ALPHA	YAW ANGLE	PITCH ANGLE	WEIGHT DDT	ISP	
C (YAW)	C (AXIAL)	C (NORMAL)	YAW F(AERO)	YAW F(AERO)	NORH F(AERO)	
THRUST	THRUST LOAD	PITCH RATE	FORCE (AZIMUTH)	FORCE (VEL)	FORCE (GAMMA)	
YAW AERO LO	AXIAL AERO LO	NORM AERO LO	AXIAL LOAD	TOTAL AERO LO	TOTAL LO	
DYN PRESS	ATM PRESS	VEL SOUND	MACH NUMBER	MERIDIAN GRVITY	RADIAL GRVITY	
TOTAL GRVITY	SURF RADIUS	GRAVITY LOSS	DRAW LOSS	MISALIGN LOSS	IDEAL VEL	
INCLINATION	MODAL LONG	PERIGEE ARG	ORBITAL PAR	ENERGY (FPS)2	ENERGY (KPS)2	
CIRCULAR VEL	APOGEE RAD	APOGEE VEL	TRUE ANOMALY	PERIOD	ECCENTRICITY	
PERIGEE, RAD	PERIGEE LAT	APOGEE LAT	PERIGEE ALT	APOGEE ALT		
*** 3.62401239E+02	0.	3.09125602E+05	3.39573009E+01	2.84902205E+02	3.11953792E+05	
1.93202571E+04	-3.67352044E-02	4.143737303E+01	2.81922000E+04	-3.51409608E-02	4.41641537E+01	
5.78707417E+07	1.11640564E+04	2.12119015E+07	0.	0.	7.30998050E+00	
0.	6.50620139E+00	0.	0.	2.07232453E+03	4.51599015E+02	
0.	0.	0.	0.	0.	0.	
9.35601376E+05	3.00000000E+00	0.	0.	9.29665063E+05	1.07341507E+05	
0.	0.	0.	3.00000000E+00	0.	3.00300000E+00	
5.77042022E-01	1.27007083E-05	9.12254614E+02	2.11785797E+01	-4.50039906E-02	3.12879070E+01	
3.12000206E+01	2.09037770E+07	4.39670713E+03	5.7961944PE+02	1.71233273E+02	2.45066911E+04	
5.46974302E+01	2.56421009E+02	2.23135152E+02	1.30325020E+07	-9.19504906E+00	-0.54251504E+01	
2.37607203E+04	2.12119099E+07	2.01921901E+04	1.0056005E+02	3.17208302E+03	3.85600210E-01	
9.40573013E+06	-3.39171247E+01	3.39171247E+01	-1.1490054E+07	3.00036379E+05		
*** 4.21256611E+02	5.0053714E+01	3.09094907E+05	3.65571000E+01	2.87011235E+02	2.11003779E+05	
2.49074000E+04	9.16237003E-05	4.30070167E+01	2.50530095E+04	0.85421500E-05	4.50005517E+01	
5.90262506E+07	2.13037366E+04	2.12106679E+07	0.	0.	1.00311001E+01	
0.	1.03700433E+00	0.	0.69660240E+01	1.40170052E+03	4.51599035E+02	
0.	0.	0.	0.	0.	0.	
6.33011336E+05	3.00000000E+00	9.41004315E-02	0.	6.32900201E+05	1.44210706E+04	
0.	0.	6.	3.00000000E+00	0.	3.00000000E+00	
0.59696920E-01	1.13745274E-05	9.16120372E+02	2.72709573E+01	-4.72770706E-02	3.12652909E+01	
3.12053266E+01	2.09037730E+07	4.39696158E+03	6.79619442E+02	1.06200901E+02	3.02474979E+04	
5.51002110E+01	2.56671703E+02	4.65657922E+01	2.13616592E+07	-6.50929465E+00	-6.12167955E+01	
2.57614706E+04	2.15140156E+07	2.50475343E+04	0.	5.20908490E+03	7.1.864742E-03	
2.12106679E+07	3.65571000E+01	-3.65571000E+01	3.09094905E+05	6.14042550E+05		

GTSM MU = 2.785482 WTVOL MU = 2.785488 INPUT TOL = .00050 TOL = .000000

1029

NA 1 CREW-33843:0 HR-3.845 FLYCK-2 SPACE SHUTTLE SYNTHESIS SUMMARY

VEHICLE

ORBITER

BOOSTER

WEIGHT (LB)
 PROPELLANT, ASCENT LFSS FPR 2272997 376741
 PROPELLANT, ASCENT FPR, 1370 3242
 PROPELLANT, ASCENT TOTAL 2274367 27984
 PROPELLANT, ORBIT MANEUVER 22371 22371
 PROPELLANT, TOTAL 402350 402350
 FLYBACK FUEL 1777 1777
 PAYLOAD 28229 28229
 STRUCTURE 134249 134249
 CONTINGENCY 13425 13425
 OTHER 7713 7713
 TOTAL 587747 587747
 IN ORBIT 211005 211005
 RETURN CONDITION 204276 204276
 ENTRY 170519 170519
 LANDING 177742 177742

VOLUME (FT3)
 FUEL TANK 13518 13518
 OXIDIZER TANK 4632 4632
 PROPELLANT TANKS 18370 18370
 PAY TAD 10600 10600
 OTHER 40121 40121
 TOTAL 69091 69091

GEOMETRY
 LENGTH (FT) 145.6 145.6
 BODY WETTED AREA (FT2) 11816.5 11816.5
 PCN PLANFORM AREA (FT2) 2085.8 2085.8
 THEORETICAL WING AREA (FT2) 1816.0 1816.0
 WING LOADING (PSF) 94.7 94.7
 PROPULSION
 THRUST-TO-WEIGHT 1.59933 1.59933
 NO. OF ENGINES 2 2
 SL THRUST/ENG NOM/UR 247697/ 247697/
 VAC THRUST/ENG NOM/UR 470000/ 470000/
 SL ISP NOM/UR 230.8/ 230.8/
 VAC ISP NOM/UR 451.6/ 451.6/

TRAJECTORY
 MASS RATIO 3.04500 3.04500
 CHARACTERISTIC VELOCITY (FPS) 15343 15343
 MAXIMUM DYNAMIC PRESSURE (PSF) 247697/ 247697/
 STAGNATION DYNAMIC PRESSURE (PSF) 470000/ 470000/
 STAGING VELOCITY (RELATIVE) (FPS) 230.8/ 230.8/
 STAGING ALTITUDE (FT) 451.6/ 451.6/
 STAGING FLIGHT PATH ANGLE (RELATIVE) (DEG) 25.53 25.53
 INJECTION VELOCITY (INERTIAL) (FPS) 307395 307395
 INJECTION ALTITUDE (FT) 4694 4694
 INJECTION FLIGHT PATH ANGLE (INERTIAL) (DEG) 58.11 58.11

FLYBACK RANGE (N MI) 413.2 413.2

SPACE SHUTTLE SYNTHESIS SUMMARY

1 GMEQ=3384390 MR=3.045 FLYBCK=2

VEHICLE

ORBITER

BOOSTER

WEIGHT (LPI)

PROPELLANT, ASCENT LESS FPR 2272997
 PROPELLANT, ASCENT FPR 1370
 PROPELLANT, ASCENT TOTAL 2214357
 PROPELLANT, ORBIT MANEUVER 2274367
 PROPELLANT, TOTAL 40947
 FLYBACK FUEL
 PAYL 0
 STRUCTURE 419412
 CONTINGENCY 41941
 OTHER 20072
 TOTAL 2795739

IN ORBIT 211005
 RETURN CONDITION 523743
 ENTRY 502891
 LANDING 461944

VOLUME (FT3)

FUEL TANK 88195
 OXIDIZER TANK 29097
 PROPELLANT TANKS 117202
 PAYLOAD 73067
 OTHER 190269
 TOTAL 215.6

GEOMETRY

LENGTH (FT) 215.6
 BODY WETTED AREA (FT2) 21692.4
 BODY PLANFORM AREA (FT2) 7327.5
 THEORETICAL WING AREA (FT2) 4041.0
 WING LOADING (PSF) 103.9

PROPULSION

THRUST-TO-WEIGHT 1.59933
 NO. OF ENGINES 11
 SL THRUST/ENG NOM/UR 407763/ 407763
 VAC THRUST/ENG NOM/UR 455221/ 455221
 SL ISP NOM/UR 391.0/391.0
 VAC ISP NOM/UR 437.6/437.6

TRAJECTORY

MASS RATIO 3.04500
 CHARACTERISTIC VELOCITY (FPS) 15363
 MAXIMUM DYNAMIC PRESSURE (PSF)
 STAGING DYNAMIC PRESSURE (PSF)
 STAGING VELOCITY (RELATIVE) (FPS)
 STAGING ALTITUDE (FT)
 STAGING FLIGHT PATH ANGLE (RELATIVE) (DEG)
 INJECTION VELOCITY (INERTIAL) (FPS)
 INJECTION ALTITUDE (FT)
 INJECTION FLIGHT PATH ANGLE (INERTIAL) (DEG)
 INJECTION INCLINATION (DEG)

FLYBACK RANGE (N MI) 413.2

376741
 3742
 37984
 2371
 402354
 1777
 28229
 134249
 13425
 7713
 507747
 211005
 204276
 179519
 177742
 13518
 4652
 18370
 19580
 40121
 60091
 145.6
 11814.5
 2845.8
 1818.0
 98.7
 1.59933
 2
 247697/
 470000/
 238.0/
 491.6/
 1.32528
 4485397/ 4485397
 5087435/ 5087435
 391.0/391.0
 437.6/437.6

30247
 499.4
 2
 18403
 263286
 4.698

25853
 309895
 .800
 55.11

SPACE SHUTTLE SYNTHESIS SUMMARY

MR 1 CMREQ=3384398 MR=3.845 FLYBACK=2
 DRAG LOSS AT STAGING = 679.62
 GRAVITY LOSS AT STAGING = 4286.34
 DRAG LOSS AT INJECTION = 679.62
 GRAVITY LOSS AT INJECTION = 4396.96
 MISALIGNMENT LOSS AT INJECTION = 186.29

PTR = 1.7550278E+00
 SELP(S) = -1.4260032E+01
 P10Y(S) = 9.055080E+01
 ACCELERATION TIME (SEC. 2) = 222.98
 ORBITER TURN TIME (SEC. 5.6.7) = 191.36

FLYBACK
 FID STAGE), 0 AT STAGING = 2.080

CRUISE, RANGE = 413.17
 SFC = .3000
 L/D = 6.5890
 CRUISE VELOCITY = 290.00
 DRAG = 8.00
 WFLYX = 0.00
 CRUISE FUEL = 40947.24
 FLYBACK FUEL = 40947.24
 C(1214) = .00064