Technical Report 341

Volume I

OPERATIONS RESEARCH *Incorporated*

SANTA MONICA, CALIFORNIA

PROPULSION AND SPACE VEHICLE SYSTEMS ANALYSIS PROGRAM

TENSOR I

By
J. B. King
R. J. Krane
G. L. Ordway
A. Reiff
W. E. Schuman
E. Wendorf

30 September 1965

Prepared for
The Performance Evaluation Section
Propulsion Division
George C. Marshall Space Flight Center
Huntsville, Alabama

Under Contract NAS 8-5321

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

I. INTRODUCTION

This document describes the ORI Propulsion and Space Vehicle Systems Analysis Computer Program, designated as the TENSOR I program. The purpose of this program is to provide a means for defining and evaluating propulsion and vehicle system requirements for many types of space missions. The program is logically divided into three distinct blocks, which when integrated, provide a comprehensive capability for space missions analysis.

Block I consists of generalized subroutines which simulate the steady-state performance of the various components of liquid propellant rocket engine systems. Executive routines, which simulate the steady-state performance of H-1, F-1, J-2 and RL-10 type engine systems, are also included. These subroutines provide a generalized capability to perform non-linear engine balances and rebalances. In addition, linear engine influence coefficients may be generated if desired.

In general, Block I provides the necessary subroutines to generate a non-linear mathematical model representing the steady-state performance of a liquid propellant rocket engine system. This non-linear mathematical model may be used to perform an engine balance and/or an engine rebalance. A balance is defined as the determination of the required orificing or control settings to cause a given engine design to operate at a given performance level with a given set of engine inlet conditions. A rebalance is the determination of engine performance when operating at a given set of inlet conditions and with a fully defined hardware configuration. The major difference between the balance and the rebalance is that in a balance, a given system performance level (thrust and mixture ratio) is desired, whereas in a rebalance the system performance level is determined as a function of the specified independent parameters.

Block II is a near-earth, vehicle system, trajectory simulation. This block simulates the flight of a point-mass missile in a 3-dimensional, earth-centered, inertial Cartesian coordinate system. A set of simultaneous differential equations of motion is integrated, using the Runge-Kutta-Gill method to define the trajectory. At each point in time, the missile position and velocity are given in the Cartesian coordinate system. These coordinates are transformed to altitude above sea level, latitude, longitude, velocity with respect to the atmosphere, velocity azimuth, and velocity β, (i.e., the angle between the local vertical and velocity vector). The missile weight is also found by integration. Missile thrust orientation is described by two angles, pitch and yaw. Pitch is measured with respect to the launch vertical; yaw is measured with respect to the launch azimuth at the launch site. Other parameters computed include local atmospheric properties, Mach number, angle of attack, thrust, flow rates, lift, drag, gravitational acceleration, and latitude and longitude on a rotating oblate spheroid.

The missile is described as a series of stages. For each stage, propulsion system data, weight and aerodynamic data are input. Thrust and flow rates may be described analytically as zero for coast; as a function of vacuum thrust, expansion ratio, throat area and altitude with constant flow rate; as a function of tabular vacuum thrust, expansion ratio, throat area and altitude with constant Isp; as a function of liquid engine influence coefficients; or as a combination of tabular vacuum thrust and liquid engine influence coefficients. Thrust and total flow rate may also be described empirically by tables which may be a function of any variable calculated by the program. Lift and drag are calculated from axial and normal force coefficients.

The flight profile is described as a series of sections. For each section, stage identification, thrust and flow rate option, thrust orientation option, initial or jettison weight, and section termination criteria are input. Thrust orientation can be described analytically as vertical rise, gravity turn, constant pitch and yaw rates (which may be zero for constant attitude), or guidance system steering. Thrust orientation can also be described empirically by tables of pitch angle and yaw angle or pitch angle-of-attack and yaw angle-of-attack which may be a function of any variable calculated by the program.

The trajectory simulation can be executed by various general search routines to optimize any dependent variables by perturbing any independent variables. Routines to find the independent variables given the values of the dependent variables are available for one function of one variable, two functions of two variables, or for finding the value of the independent variable corresponding to the minimum or maximum of a function.

A parameter study and a "gain" routine are available for generating a series of perturbation studies and, if desired, general partial derivatives of the gains of up to 10 dependent parameters with respect to up to 10 independent parameters. A propellant loading optimization routine is also available. These options are available with or without concurrent search routines.

Block III of the TENSOR I computer program is a lunar and interplanetary transfer simulation. This block is used to determine three-dimensional geocentric launch conditions for direct single maneuver lunar and interplanetary transfers. Although Block III is based upon an impulsive velocity requirement, the propulsion system performance may be simulated during the finite time interval required to attain this velocity.

Table 1 is a listing of the subroutines which comprise the TENSOR I computer program. Complete documentation of these subroutines is contained in this report (Volume I).

The documentation of each subroutine consists of a listing of the step-by-step programmed procedure with brief comments regarding the solution. A nomenclature list which defines and references each variable used in the subroutine and a symbolic listing of the subroutine are also included. The references given for a variable pertain to the statement numbers, in the step-by-step procedure, which contain the variable and any specification statements which may contain the variable. The abbreviations DIM, EQUIV and DP are used to denote DIMENSION, EQUIVALENCE and DOUBLE PRECISION, respectively. COMMON block names are embedded in slashes, e.g., /CONST/.

Table 2 provides a cross reference of the COMMON blocks and the subroutines in which they are used. Table 3 summarizes the flags used within Blocks II and III.

Volume II of this report contains the input data and output listings of a number of test cases which were run with the TENSOR I program.

TABLE 1. TENSOR I SUBROUTINES

MNCNT	Main TENSOR I Program control
CNTROL	Main control for Block I (non-linear engine balance/rebalance simulation)
GUIDON	Main control for Block II (powered-flight simulation)
TRAJ	Control routine for trajectory portion of Block II
SEARCH	Control deck for search routines
SRCH1	One-parameter-search routine
DSRCH1	One-parameter-search routine
SRCH2	Two-parameter-search routine
MNX	Minimization/maximization-search routine
DMNX	Minimization/maximization-search routine
SDATA	Pre-assembled nominal inputs for search routines
PSTDY	Performs equal increment perturbations of one or two parameters
GAINS	Calculates the gains of up to ten dependent variables with respect to up to ten independent variables
NEWTON	Computes derivatives for each integration interval during vehicle flight
GUIDE	Computes vehicle attitude during flight

TABLE 1. TENSOR I SUBROUTINES (cont.)

INIT	Sets up initial conditions for Block II
SUBA	Sets up section and stage working data for Blocks II and III
ENDOS	End of section convergence routine
ENGSU	Sets up working data for Subroutine ENGINE at the beginning of each stage
ENGINE	Computes instantaneous thrust and flow rates during flight
VARYD	Computes the quantity $(X_i - \bar{X}_i)/(\bar{X}_i)$ for use in influence coefficient calculations
GICEQ	Computes the instantaneous value of a dependent variable using the general influence coefficient equation
INTEG	Runge-Kutta-Gill integration routine
NTERP	Performs linear, quadratic or cubic interpolation
STDATM	Standard earth atmospheric model
HITE	Computes the height of propellant in a tank given the tank geometry and propellant volume
CROOT	Determines the real, positive root of a cubic
G1OB	Performs the coordinate transformation from the local geocentric coordinates of a rotating oblate spheroid to inertial Cartesian coordinates
G2OB	Performs the coordinate transformation from inertial Cartesian coordinates to the local geocentric coordinates of a rotating oblate spheroid
FINPUT	General input routine for Block II and Block III (lunar and interplanetary transfer simulation)
FXINP	Converts a list of floating point numbers to fixed point numbers

TABLE 1. TENSOR I SUBROUTINES (cont.)

SYMTAB	Input symbol table for Block II and Block III
BLKDTA	Presets all Block II and Block III input variables to nominal values to be used unless they are redefined by input
UNPACK	Decodes and executes logical control data input by Subroutine FINP
LOOKUP	Searches the symbol table for the Hollerith names corresponding to a parameter location
ERROR	Communicates error conditions between subroutines
DUMPS	Dumps named COMMON blocks on call
PRNTKY	Prints an output key of the Block II output
OUTPUT	Block II output routine
XPRINT	Optimal extra print line routine
PLOAD	Optimum propellant loading routine
TAN	Computes tangent of an angle
ASIN	Computes arcsine
ACOS	Computes arccosine
QUAD	Computes the value of an angle between 0 and 2π radians given its sine and cosine
PATCH	Determines three-dimensional geocentric launch coordinates for direct, single maneuver interplanetary transfers
EFEM	Provides the classical set of orbital elements for the planets Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus and Neptune at a given Julian date

TABLE 1. TENSOR I SUBROUTINES (cont.)

JULDT	Computes a double precision Julian date and corresponding Greenwich sidereal time, given the local time
TORB	Computes the position and velocity vectors of an orbiting body at a particular time, given the classical set of orbital elements
VITER	Determines a preliminary orbit from two position vectors by iteration on the true anomaly
CRPRD	Computes the component terms of the cross product of two vectors
DTPRD	Computes the dot product of two vectors
MOON	Provides lunar ephemeris data
LUNAR	Computes required central transfer angle for lunar transfer
BART	Computes the sidereal times at booster launch and re-ignition, the required direction of the thrust vector during re-ignition and the required velocity vector at the end of re-ignition for interplanetary and lunar transfers
ROT	Computes the components of a vector in an orthogonal set after rotation about one axis in another orthogonal set
H1	Executive routine for H-1 type engine system simulation
F1	Executive routine for F-1 type engine system simulation
J2	Executive routine for J-2 type engine system simulation
RL10	Executive routine for RL10-A1 type engine system simulation
CHAMBR	Computes the operating performance of a thrust chamber

TABLE 1. TENSOR I SUBROUTINES (cont.)

PUMP	Computes the operating performance of a pump
TURBIN	Computes the operating performance of a turbine
TURBEX	Computes the operating performance of a turbine exhaust system
GPROP	Computes gas generator gas properties
RES	Computes the pressure drop across a resistance through which a liquid flows
CHMBER	Computes the chamber performance for an RL10-A1 engine
PUMP1	Computes the pump performance for an RL10-A1 engine
LOX	Computes the liquid oxygen mass flow rate for an RL10-A1 engine
FUEL	Computes the hydrogen mass flow rate for an RL10-A1 engine
TRBINE	Computes turbine performance for an RL10-A1 engine
TCPROP	Computes theoretical thrust coefficient and characteristic velocity for liquid oxygen/RP-1 and liquid oxygen/liquid hydrogen propellant combinations.
PRPDEN	Computes density and vapor pressure for liquid oxygen, liquid hydrogen, and RP-1 propellants
BIVAR/UVAR	Bivariate and univariate curve fit routine used with RL10-Al engine simulation
PTEST	Error routine used with Subroutine BIVAR
EPRINT	Block I output routine
INFL	Generates a set of non-dimensional, linear engine influence coefficients

TABLE 1. TENSOR I SUBROUTINES (cont.)

STINFL	Saves nominal engine performance and influence coefficients generated by Block I, for use in Block II
SETNOM	Saves and resets nominal Block I input variables
SWITCH	Sets one variable equal to another
ESTM	Calculates the value X which yields Y = 0 for a given function Y = Y(X)
VPOLY	Evaluates a polynomial
COMON	Produces a beginning location for saving a portion of LINK 1
COMOE	Produces an ending location for saving a portion of LINK 1
.UN08.	Tape unit specification routine
BDATA	Presets certain data in COMMON blocks used by both Block I and Block II
TURMAP	Computes turbine performance using a turbine map
PU	Computes propellant utilization valve flow loss coefficient
SKIP	Skips orbital coast
COAST	Simulates orbital coast
INJEC	Stores velocity components required for orbital transfer
TRNSFR	Main control for Block III

TABLE 2. COMMON BLOCK/SUBROUTINE CROSS REFERENCE

	SUBROUTINES																								
COMMON BLOCKS	BART	BDATA	BLKDTA	CHAMBR	CHMBER	CNTROL	COAST	EFEM	ENDOS	ENGINE	ENGSU	EPRINT	F1	FUEL	GAINS	GPROP	GUIDE	GUIDON	H1	INFL	INIT	INJEC	J2	LOOKUP	LOX
ASTRO	X		X					X										X							
BCIPZE																								X	
CBAL				X	X	X						X	X	X					X	X			X		
CCHAM				X	X	X						X	X	X					X				X		X
CDEN						X						X	X						X				X		
CGPROP						X						X	X	X		X			X				X		
CHRL10					X	X						X													
CJ2IN												X											X		
CJ2OUT												X											X		
CMAIN					X	X						X	X	X					X				X		X
COMP																	X	X			X				
CONST			X						X	X	X				X		X	X			X	X			
CPA				X	X	X						X	X	X					X				X		
CPUMP					X	X						X	X	X					X				X		X
CRES						X						X	X						X				X		
CRESGG					X	X						X	X	X					X				X		
CTURB						X						X	X						X				X		
CTURBX						X						X	X						X						
CTURMP												X											X		
DATA									X	X	X						X	X			X	X			
DUMPY			X															X							
ENGINI			X							X	X							X							
ENGINO									X	X	X							X							
ENGINW									X	X	X							X			X				
FLRL10						X						X		X											
GAIND			X												X			X							
GILL									X	X							X	X			X				
H1F1J2						X						X	X						X				X		
INCON						X						X													
INFLTB		X				X					X									X					
INPUT			X						X		X						X	X			X				
IPRINT			X															X							
LXRL10						X						X		X											X
MNX1																		X							
MNX2																		X							

	SUBROUTINES																							
COMMON BLOCKS	LUNAR	NEWTON	OUTPUT	PATCH	PLOAD	PRNTKY	PUMP	PUMP1	RES	RL10	SDATA	SEARCH	SETNOM	SKIP	STINFL	SUBA	SYMTAB	TRAJ	TRBINE	TRNSFR	TURBEX	TURBIN	TURMAP	XPRINT
ASTRO	X			X													X							
BCIPZE																								
CBAL							X			X			X											
CCHAM								X		X			X						X		X			
CDEN							X		X				X											
CGPROP													X										X	
CHRL10								X		X			X											
CJ2IN													X											
CJ2OUT																								
CMAIN								X		X			X						X					
COMP		X	X													X	X							
CONST		X	X		X									X		X	X	X		X				X
CPA										X			X								X			
CPUMP							X	X		X			X						X					
CRES									X				X											
CRESGG										X			X											
CTURB								X		X			X						X		X	X	X	
CTURBX													X								X			
CTURMP													X										X	
DATA		X	X								X					X	X	X		X				
DUMPY		X															X	X						
ENGINI				X													X							
ENGINO			X														X							
ENGINW		X															X							
FLRL10								X		X			X						X					
GAIND																	X							
GILL		X	X											X		X	X	X		X				
H1F1J2													X											
INCON													X											
INFLTB													X		X									
INPUT		X	X		X						X					X	X			X				
IPRINT			X														X							
LXRL10								X		X			X											
MNX1											X	X					X							
MNX2											X	X					X							

TABLE 2. COMMON BLOCK/SUBROUTINE CROSS REFERENCE (cont.)

	SUBROUTINES																																															
COMMON BLOCKS	BART	BDATA	BLKDTA	CHAMBR	CHMBER	CNTROL	COAST	EFEM	ENDOS	ENGINE	ENGSU	EPRINT	F1	FUEL	GAINS	GPROP	GUIDE	GUIDON	H1	INFL	INIT	INJEC	J2	LOOKUP	LOX	LUNAR	NEWTON	OUTPUT	PATCH	PLOAD	PRNTKY	PUMP	PUMP1	RES	RL10	SDATA	SEARCH	SETNOM	SKIP	STINFL	SUBA	SYMTAB	TRAJ	TRBINE	TRNSFR	TURBEX	TURBIN	TURMAP	XPRINT
MNRL10					X	X						X		X											X								X		X			X						X					
OBART	X																	X								X			X													X			X				
PBART	X						X											X				X																	X			X			X				
PNSPRI						X						X																							X			X											
PPRL10						X						X																					X		X			X											
PROPL			X															X												X												X							
SAVE									X	X	X						X	X			X						X			X									X		X	X			X				
SRCH1A																		X																		X	X					X							
SRCH1B																		X																		X	X					X							
SRCH2A																		X																		X	X					X							
STAGES		X				X					X																											X											
STUDY																		X																		X	X					X							
TABLES			X							X							X	X			X						X															X							
TAPES			X												X			X									X	X		X	X											X	X		X				
TBRL10						X						X																					X		X			X						X					
TOL	X																	X																								X			X				
WFLAG									X									X			X						X	X													X	X	X						
WORKD							X		X	X	X						X	X			X	X					X	X													X	X	X		X				

TABLE 3. BLOCKS II AND III FLAG SUMMARY

FLAG	DESCRIPTION	REFERENCES	
		COMMON BLOCK	SUBROUTINES
ATIND	Indicator in section input data for attitude control option desired in that section	WORKD	GUIDE, NEWTON
FLCNV	Shut-down volume convergence flag .TRUE. = converged .FALSE. = not converged	—	ENDOS
FLDSC	Discontinuity flag. A discontinuity is defined as any time when the derivatives of the integrated variables must be re-evaluated before integration can proceed	WFLAG	ENDOS, NEWTON, OUTPUT, TRAJ
FLEOF	End-of-flight flag (all sections of trajectory simulation are completed) .TRUE. = end-of-flight sensed .FALSE. = not end-of-flight	WFLAG	INIT, OUTPUT, SUBA, TRAJ
FLEOS	End-of-section flag .TRUE. = end-of-section sensed .FALSE. = not end-of-section	WFLAG	ENDOS, OUTPUT, TRAJ

TABLE 3. BLOCKS II AND III FLAG SUMMARY (cont.)

FLAG	DESCRIPTION	REFERENCES	
		COMMON BLOCK	SUBROUTINES
FLGHD	Hold-down simulation flag .TRUE. = hold-down is being simulated .FALSE. = no hold-down	ENGINW	ENDOS, ENGINE, ENGSU, NEWTON
FLGSD	Shut-down simulation flag .TRUE. = shut-down is being simulated .FALSE. = no shut-down	ENGINW	ENDOS, ENGINE, ENGSU
FLPAR	Input flag for execution of Subroutine GAINS 0 = do not execute Subroutine GAINS ≠ 0 = execute Subroutine GAINS	GAIND	BLKDTA, GUIDON
FLPRT	Print option input flag. This is a two-part binary (octal) integer of the form XY where X is the print option control during searches and Y is the print option control during converged or straight trajectories. Each part is the sum of the desired following options: 1 — one-line burn-out print 2 — discontinuity print 3 — standard time history print	INPUT	BLKDTA, OUTPUT

TABLE 3. BLOCKS II AND III FLAG SUMMARY (cont.)

FLAG	DESCRIPTION	REFERENCES	
		COMMON BLOCK	SUBROUTINES
FLSC	Input flag for number of sections	INPUT	BLKDTA, GUIDON, SUBA
FLTP	Flag to print	WFLAG	ENDOS, OUTPUT, TRAJ
FLTR	Input flag for type of run. If this is negative, a parameter study will also be executed	INPUT	BLKDTA, GUIDON
FLTRW	Working type of run flag. FLTRW = |FLTR| except during the printing of a converged trajectory. Then it is equal to zero. If FLTRW = 0, the Y part of the print flag (FLPRT) is used for print control. If FLTRW ≠ 0, the X part of the print flag is used for print control.	WFLAG	GUIDON, OUTPUT
FPTPA	Tank pressure option input flag	ENGINW	ENGINE

TABLE 3. BLOCKS II AND III FLAG SUMMARY (cont.)

FLAG	DESCRIPTION	REFERENCES	
		COMMON BLOCK	SUBROUTINES
TARGET	Target body for lunar or interplanetary transfers 0 — Moon 1 — Mercury 2 — Venus 4 — Mars 5 — Jupiter 6 — Saturn 7 — Uranus 8 — Neptune	INPUT	PATCH, TRNSFR
THIND	Indicator in section input data for thrust option desired in that section	WORKD	COAST, ENDOS, ENGINE, OUTPUT
TYSC	Tally of the number of sections flown. When TYSC = FLSC, the last section is being flown. When TYSC > FLSC, FLEOF = .TRUE. .	WFLAG	ENDOS, INIT, NEWTON, OUTPUT, SUBA

II. SUBROUTINES MNCNT, COMON, COMOE AND UNITSP

MNCNT is the main control routine for the TENSOR I computer program. It brings in the correct block (Block I or Block II of the TENSOR I computer program) through the overlay feature of the IBSYS FORTRAN IV Loader (see Figure 1 for generalized subroutine linkage).

Since the variables that are input to Block I of the TENSOR program must be saved from case to case, Subroutines COMON and COMOE are called to produce a beginning and ending location for saving this information on tape. Block II variables are always kept in core. Subroutine COMON is also used as a supplementary entry point back into MNCNTL from CNTROL (entry point is COMONS). This entry is used only if an error is detected during processing of the engine balance/rebalance programs.

Deck UNITSP is a tape unit specification routine that assigns the first available tape unit on channel A as the scratch tape used for the saving of Block I variables, and defines the variable tape units used within the various subroutines.

Subroutines FRDU, FIOU, FXP1, FRCD, FATN, FDSC, FDAT, FDSQ, FSLITE, FXP2, FDXP, FDLG and FDX2 are FORTRAN IV System routines that are included in the various links via $INCLUDE control cards.

CALLING SEQUENCE

None. Entry is accomplished by the FORTRAN IV $ENTRY control card. Figure 2 is an overlay map for the program.

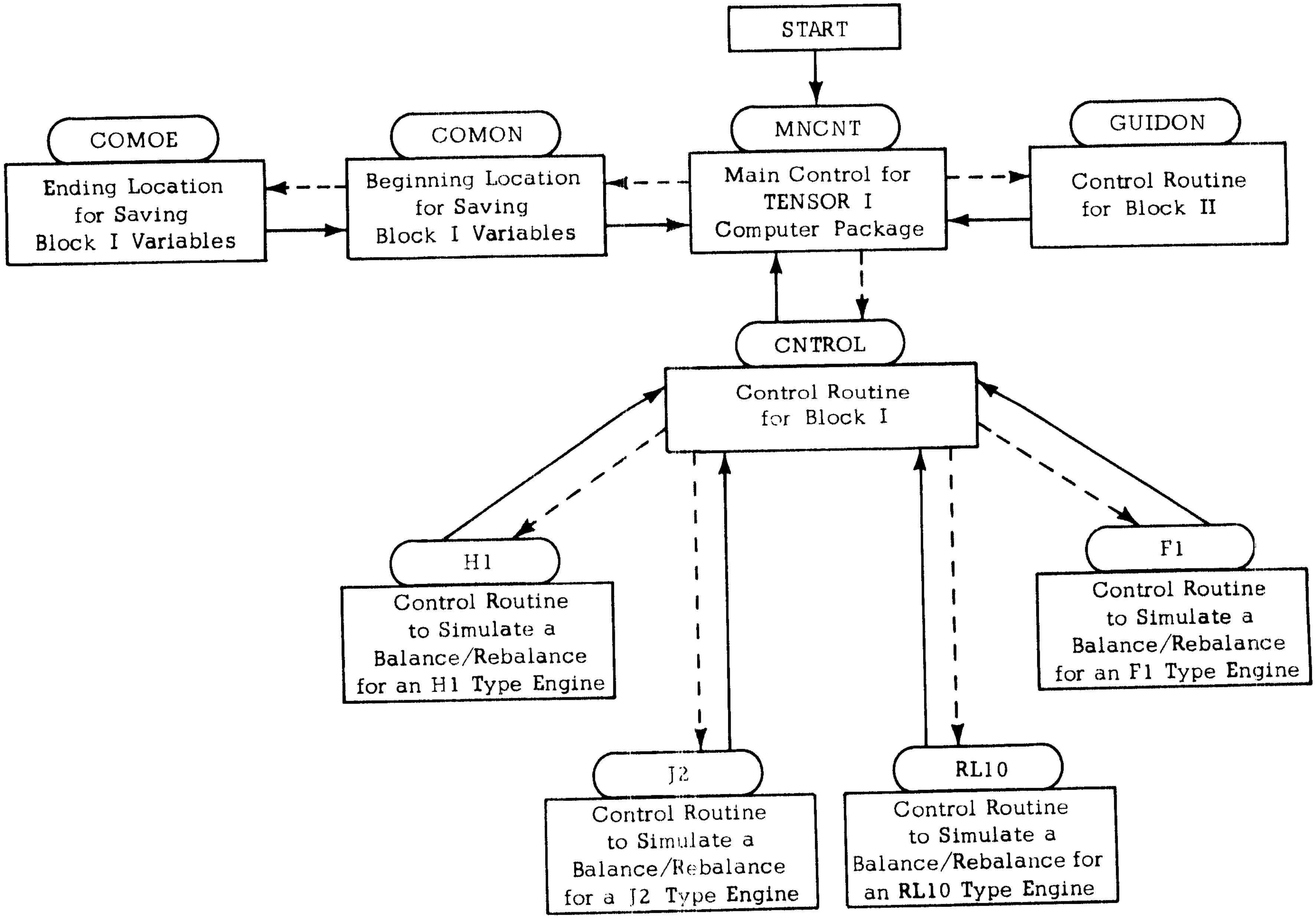
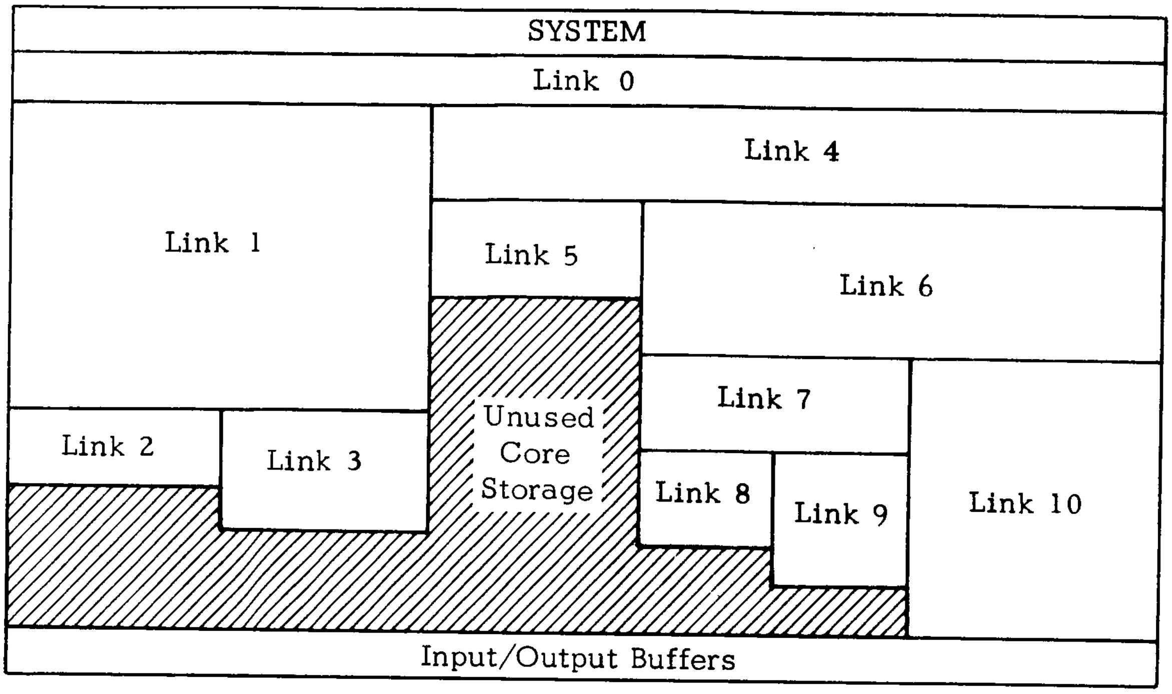


FIGURE 1. GENERALIZED SUBROUTINE LINKAGE DIAGRAM FOR TENSOR I COMPUTER PROGRAM



LINK 0 contains subroutines: MNCNT, UNITSP, BDATA, SDATA, BLKDTA, VPOLY, TCPROP, PRPDEN and FORTRAN IV system routines.

LINK 1 contains subroutines: FRDU, FIOU, COMON, SETNOM, CNTROL, INFL, STINFL, ESTM, EPRINT, H1, F1, TURBEX, CHAMBR, TURBIN, PUMP, RL10, CHMBER, LOX, FUEL, PUMP1, TRBINE, J2, COMOE

LINK 2 contains subroutines: GPROP, SWITCH, RES, PU, TURMAP

LINK 3 contains subroutines: BIVAR, UVAR, PTEST

LINK 4 contains subroutines: FXP1, SYMTAB, DUMPS, LOOKUP, UNPACK, ERROR, GUIDON

FIGURE 2. TENSOR I OVERLAY MAP

LINK 5 contains subroutines: FRCD, PRNTKY, FINPUT, FXINPT

LINK 6 contains subroutines: FATN, FDSC, FDAT, FDSQ, SEARCH, PSTDY, SRCH1, DSRCH1, SRCH2, MNX, DMNX, PLOAD, GAINS, TRAJ, G1OB, G2OB, TAN, QUAD, ACOS, ASIN

LINK 7 contains subroutines: FSLITE, INIT, SUBA, JULDT, ENGSU, BART, ROT, COAST, SKIP, INJEC, TRNSFR

LINK 8 contains subroutines: FXP2, LUNAR, MOON

LINK 9 contains subroutines: FDXP, FDLG, FDX2, EFEM, PATCH, TORB, VITER, CRPRD, DTPRD

LINK 10 contains subroutines: ENDOS, OUTPUT, XPRINT, NEWTON, GUIDE, ENGINE, HITE, CROOT, STDATM, NTERP, INTEG, VARYD, GICEQ

FIGURE 2. TENSOR I OVERLAY MAP (cont.)

```
$IBMAP  MNCNTL
        ENTRY     MNCNT
 MNCNT  CALL      C0M0N(A,MNCER,TRJF)
        CLA       A
        STA       RDT+3
        STA       RDTA+3
        ARS       18
        SUB       RDT+3
        STA       LGTH
AMNCNT  STZ       TRJF
        NZT       FLG
        TRA       C0NT
        CALL      C0M0N(A)
        CALL      .FRWT.(.UNO8.)
        CALL      .FRDB.(.UNO8.)
   RDT  CALL      .FBLI.(**,LGTH)
        TSX       .FRLR.,4
        CALL      .FRWT.(.UNO8.)
  C0NT  CALL      CNTR0L(TRJF)
 MNCER  CLA       TRJF
        TZE       C0NT
        STL       FLG
        CALL      .FRWT.(.UNO8.)
        CALL      .FWRB.(.UNO8.)
  RDTA  CALL      .FBL0.(**,LGTH)
        TSX       .FWLR.,4
        CALL      .FRWT.(.UNO8.)
        CALL      GUID0N(TRJF)
        TRA       AMNCNT
     A  PZE
   FLG  PZE
  LGTH
  TRJF  PZE
        END
```

FIGURE 1. SYMBOLIC LISTING OF SUBROUTINE MNCNT

```
$IBMAP CØMØS
       ENTRY   CØMØN
       ENTRY   CØMØNS
 CØMØN SXA     XR,4
       CLA     RETMN
       TZE     MNRET
       CALL    CØMØE(CA)
    XR AXT     **,4
       CLA     CA
       STØ*    3,4
       TRA     1,4
 MNRET CLA     4,4
       STA     RETMN
       CLA     5,4
       STA     TRJF
       TRA     CØMØN+3
CØMØNS CLA*    3,4
       STØ*    TRJF
       LAC     RETMN,4
       TRA     0,4
  TRJF PZE
 RETMN PZE
    CA PZE     CØMØN,,**
       END
```

FIGURE 4. SYMBOLIC LISTING OF SUBROUTINE COMON

```
$IBMAP COMOES   FULIST,REF
       ENTRY    CØMØE
 CØMØE CLA      Z
       STD*     3,4
       TRA      1,4
     Z PZE      0,,CØMØE
       END
```

FIGURE 5. SYMBOLIC LISTING OF SUBROUTINE COMOE

```
$IBMAP UNITSP  LIST,DECK
       ENTRY   .UN01.
.UN01. PZE     UNIT01
UNIT01 FILE    ,B(1),DEFER,INØUT,BLK=256,BIN,NØLIST,PØØL
       ENTRY   .UN02.
.UN02. PZE     UNIT02
UNIT02 FILE    ,B(1),DEFER,INØUT,BLK=256,BIN,NØLIST,PØØL
       ENTRY   .UN03.
.UN03. PZE     UNIT03
UNIT03 FILE    ,B(1),DEFER,INØUT,BLK=256,BIN,NØLIST,PØØL
       ENTRY   .UN04.
.UN04. PZE     UNIT04
UNIT04 FILE    ,B(1),DEFER,INØUT,BLK=256,BIN,NØLIST,PØØL
       ENTRY   .UN08.
.UN08. PZE     UNIT08
UNIT08 FILE    ,A(1),MØUNT,INØUT,BLK=256,BIN,NØLIST,PØØL
       ENTRY   .UN09.
.UN09. PZE     UNIT09
UNIT09 FILE    ,B(1),DEFER,INØUT,BLK=256,BIN,NØLIST,PØØL
       ENTRY   .UN10.
.UN10. PZE     UNIT10
UNIT10 FILE    ,B(1),DEFER,INØUT,BLK=256,BIN,NØLIST,PØØL
       ENTRY   .UN11.
.UN11. PZE     UNIT11
UNIT11 FILE    ,B(1),DEFER,INØUT,BLK=256,BIN,NØLIST,PØØL
       END
```

FIGURE 6. SYMBOLIC LISTING OF SUBROUTINE UNITSP

III. BLOCKS II AND III INPUT DATA SUMMARY

The trajectory simulation program reads, converts, and stores input via a special FORTRAN input routine called FINPUT which requires a particular input format. The majority of the data is numerical and is usually converted to floating point real numbers. A few items, however, must be converted as binary integers or decimal integers denoted by a "B" or an "I" prefix respectively. Also used is a non-numeric input option denoted as symbolic input and indicated by an "A" prefix. The symbolic input option assumes a 12-character value divided into 4 subfields, AAAAA T DDDDD P, where internal to the trajectory program:

AAAAA	is used to specify a variable or array,
T	is an integer specifying the type of conversion,
DDDDD	is used to specify a routine to be executed <u>or</u> the subscript of the array in AAAAA if the subscript is not one,
P	may be used to indicate the sign of the variable slope.

AAAAA will be referred to as the address, T as the tag, DDDDD as the decrement, and P as the prefix portion of the value.

The input for Blocks II and III can be divided into 3 categories: general trajectory data, miscellaneous optional data, and search control data.

GENERAL TRAJECTORY DATA

General trajectory data can be subdivided into 5 types: flags, headers and initial conditions; section data; stage data; engine data; and tabular data.

Flags, Headers and Initial Conditions

Three flags are required:

FLTR = flag for type of run

FLSC = flag for number of sections

FLPRT = flag for print options

FLPRT is a 2-part binary number (i.e., input with a "B" prefix) of the form xy where x is the print control during searching, and y is the print control for converged and normal trajectories. x and y are each the sum of the following numbers corresponding to a different print option:

1 = burnout print on optional tape

2 = discontinuity print

4 = time history print

There are 15 headers available for print-out. Headers 1 - 12 print out at the end of sections 1 - 12. Headers 13 - 15 print out at the beginning of a trajectory print-out.

Initial conditions of the integrated variables are specified as follows:

TI	initial time	seconds
VI	initial relative velocity	ft/sec
HI	initial altitude above sea level	ft
BETAI	initial β (angle of velocity vector with local vertical)	degrees
AZI	initial or firing azimuth	degrees
LATI	initial latitude	degrees
LONGI	initial longitude	degrees
WI	initial weight	lbs
WPL	additional payload weight	lbs
BETA0	velocity vector kick at end of section one	degrees
DATE	date array: year, month, day, hour, minute, second, and time zone	
SIDETI	initial local sidereal time if DATE not input	degrees
TARGT	Target body (A-prefix conversion code)	
TRANS	Transfer time	days for planets hours for moon

Section Data

The section data is an array of 12 sections each composed of 20 entries. The 20 entries in any section are defined as follows:

1.	Section ending parameter (A-prefix conversion code) If this is input as a zero (0), the section will end on the relative section time indicated in the next location.	
2.	Value of parameter to end section	
3.	Section initial weight (> 0) or jettison weight (< 0)	lbs
4.	Stage indicator	
5.	Thrust option indicator	
6.	Attitude control option indicator	
7.	Section integration interval	sec
8.	Section print interval	sec
9.	Pitch rate, attitude option 2	deg/sec
10.	Yaw rate, attitude option 2	deg/sec
11.	Instantaneous pitch kick, attitude option 2	deg
12.	Instantaneous yaw kick, attitude option 2	deg
13.	Velocity to go in x direction, attitude option 5	ft/sec
14.	Velocity to go in y direction, attitude option 5	ft/sec
15.	Velocity to go in z direction, attitude option 5	ft/sec
16.	Section order number	
17.	Parameter for section skip (A-prefix conversion code)	

18. Test value for section skip

19. Optional routine to execute before moving stage data (A-prefix conversion code)

20. Optional routine to execute after moving stage data (A-prefix conversion code)

Stage Data

The stage data is an array of 6 stages each composed of 20 entries. The 20 entries in any stage are defined as follows:

1.	Reference area	ft^2
2.	Stage initial weight ($>$ 0) or stage jettison weight ($<$ 0)	lb
3.	Stage initial oxidizer weight	lb
4.	Stage initial fuel weight	lb
5.	Stage monopropellant weight	lb
6.	Stage burnout weight. If any section in this stage ends on W of 0, this weight is used	lb
7.	Vacuum thrust, thrust option 1	lb
8.	Flow rate, thrust option 1	lb/sec
9.	Expansion ratio, thrust options 1, 3, 5	
10.	Throat area, thrust options 1, 3, 5	in^2

11.	Isp, thrust options 3, 5	sec
12.	Thrust multiplier, thrust options 2, 3	
13.	Flow rate multiplier, thrust option 2	
14.	Pitch multiplier, attitude options 3, 4	
15.	Yaw multiplier, attitude options 3, 4	
16.	Normal force multiplier	
17.	Axial force multiplier	
18.	Not used	
19.	Not used	
20.	Not used	

Engine Data

The engine data consists of a number of arrays of data necessary only for thrust options 4 and 5; i indicates the stage from 1 to 6.

YBBRi = specific values of dependent engine parameter

1.	$\bar{\bar{P}}_c$	thrust chamber nozzle stagnation pressure	psia
2.	$\bar{\bar{\dot{W}}}_{oc}$	thrust chamber oxidizer flow rate	lb/sec
3.	$\bar{\bar{\dot{W}}}_{fc}$	thrust chamber fuel flow rate	lb/sec

4.	$\bar{\bar{\dot{W}}}_{og}$	gas generator oxidizer flow rate	lb/sec
5.	$\bar{\bar{\dot{W}}}_{fg}$	gas generator fuel flow rate	lb/sec

YBAR i = nominal values of dependent engine parameters

1.	$\bar{P}_c$	thrust chamber nozzle stagnation pressure	psia
2.	$\bar{\dot{W}}_{oc}$	thrust chamber oxidizer flow rate	lb/sec
3.	$\bar{\dot{W}}_{fc}$	thrust chamber fuel flow rate	lb/sec
4.	$\bar{\dot{W}}_{og}$	gas generator oxidizer flow rate	lb/sec
5.	$\bar{\dot{W}}_{fg}$	gas generator fuel flow rate	lb/sec

XNOM i = nominal values of independent engine parameters

1.	$\bar{P}_o$	oxidizer pump inlet pressure	psia
2.	$\bar{P}_f$	fuel pump inlet pressure	psia
3.	$\bar{T}_o$	oxidizer pump inlet temperature	^{o}R
4.	$\bar{T}_f$	fuel pump inlet temperature	^{o}R
5.	$\bar{F}$	nominal thrust	lb
6.	$\bar{\eta}_c^*$	c* correction factor	

Influence Coefficients

KPC i = P_c influence coefficients

1. $(\partial P_c / \partial P_o)(\bar{P}_o / \bar{P}_c)$
2. $(\partial P_c / \partial P_f)(\bar{P}_f / \bar{P}_c)$
3. $(\partial P_c / \partial T_o)(\bar{T}_o / \bar{P}_c)$
4. $(\partial P_c / \partial T_f)(\bar{T}_f / \bar{P}_c)$
5. $(\partial P_c / \partial F)(\bar{F} / \bar{P}_c)$
6. $(\partial P_c / \partial \eta_c^*)(\bar{\eta}_c^* / \bar{P}_c)$

KWOC i = $\dot{W}_{oc}$ influence coefficients

1. $(\partial \dot{W}_{oc} / \partial P_o)(\bar{P}_o / \bar{\dot{W}}_{oc})$
2. $(\partial \dot{W}_{oc} / \partial P_f)(\bar{P}_f / \bar{\dot{W}}_{oc})$
3. $(\partial \dot{W}_{oc} / \partial T_o)(\bar{T}_f / \bar{\dot{W}}_{oc})$
4. $(\partial \dot{W}_{oc} / \partial T_f)(\bar{T}_f / \bar{\dot{W}}_{oc})$
5. $(\partial \dot{W}_{oc} / \partial F)(\bar{F} / \bar{\dot{W}}_{oc})$
6. $(\partial \dot{W}_{oc} / \partial \eta_c^*)(\bar{\eta}_c^* / \bar{\dot{W}}_{oc})$

KWFCi = $\dot{W}_{fc}$ influence coefficients

1. $(\partial \dot{W}_{fc}/\partial P_o)(\overline{P}_o/\overline{\dot{W}}_{fc})$

2. $(\partial \dot{W}_{fc}/\partial P_f)(\overline{P}_f/\overline{\dot{W}}_{fc})$

3. $(\partial \dot{W}_{fc}/\partial T_o)(\overline{T}_o/\overline{\dot{W}}_{fc})$

4. $(\partial \dot{W}_{fc}/\partial T_f)(\overline{T}_f/\overline{\dot{W}}_{fc})$

5. $(\partial \dot{W}_{fc}/\partial F)(\overline{F}/\overline{\dot{W}}_{fc})$

6. $(\partial \dot{W}_{fc}/\partial \eta_c^*)(\overline{\eta}_c^*/\overline{\dot{W}}_{fc})$

KWOGi = $\dot{W}_{og}$ influence coefficients

1. $(\partial \dot{W}_{og}/\partial P_o)(\overline{P}_o/\overline{\dot{W}}_{og})$

2. $(\partial \dot{W}_{og}/\partial P_f)(\overline{P}_f/\overline{\dot{W}}_{og})$

3. $(\partial \dot{W}_{og}/\partial T_o)(\overline{T}_o/\overline{\dot{W}}_{og})$

4. $(\partial \dot{W}_{og}/\partial T_f)(\overline{T}_f/\overline{\dot{W}}_{og})$

5. $(\partial \dot{W}_{og}/\partial F)(\overline{F}/\overline{\dot{W}}_{og})$

6. $(\partial \dot{W}_{og}/\partial \eta_c^*)(\overline{\eta}_c^*/\overline{\dot{W}}_{og})$

KWFG i = $\dot{W}_{fg}$ influence coefficients

1. $(\partial \dot{W}_{fg} / \partial P_o)(\bar{P}_o / \bar{\dot{W}}_{fg})$
2. $(\partial \dot{W}_{fg} / \partial P_f)(\bar{P}_f / \bar{\dot{W}}_{fg})$
3. $(\partial \dot{W}_{fg} / \partial T_o)(\bar{T}_o / \bar{\dot{W}}_{fg})$
4. $(\partial \dot{W}_{fg} / \partial T_f)(\bar{T}_f / \bar{\dot{W}}_{fg})$
5. $(\partial \dot{W}_{fg} / \partial F)(\bar{F} / \bar{\dot{W}}_{fg})$
6. $(\partial \dot{W}_{fg} / \partial \eta_c^*)(\bar{\eta}_c^* / \bar{\dot{W}}_{fg})$

Pump Inlet Temperature Polynomial Coefficients

CTOXi = oxidizer temperature coefficients

1. $a_0(x^0)$
2. $a_1(x^1)$
3. $a_2(x^2)$
4. $a_3(x^3)$
5. $a_4(x^4)$
6. $a_5(x^5)$

CTFLi = fuel temperature coefficients

1. $b_0(x^0)$
2. $b_1(x^1)$
3. $b_2(x^2)$
4. $b_3(x^3)$
5. $b_4(x^4)$
6. $b_5(x^5)$

Propellant Tank Data

TNKSi

1. propellant type indicator

   1 = Lox/RP1
   2 = Lox/LH_2

2. reference temperature - RP1 °F
3. reference specific gravity - RP1
4. tank pressure option

   0 = constant
   $\neq 0$ = stepped

5.	constant oxidizer tank top pressure	psig
6.	constant fuel tank top pressure	psig
7.	stage time to step initial oxidizer tank pressure	sec
8.	initial oxidizer tank pressure	psia
9.	stage time to reach final oxidizer tank pressure	sec
10.	final oxidizer tank pressure	psia
11.	1st-degree term oxidizer pressure rise polynomial coefficient	
12.	2nd-degree term oxidizer pressure rise polynomial coefficient	
13.	stage time to step initial fuel tank pressure	sec
14.	initial fuel tank pressure	psia
15.	stage time to reach final fuel tank pressure	sec
16.	final fuel tank pressure	psia
17.	1st-degree term fuel pressure polynomial coefficient	
18.	2nd-degree term fuel pressure polynomial coefficient	
19.	length of cylindrical part of oxidizer tank	ft
20.	radius of oxidizer tank	ft

21.	type of oxidizer tank upper header, 1 - 4	
22.	type of oxidizer tank lower header, 1 - 4	
23.	length of cylindrical part of fuel tank	ft
24.	radius of fuel tank	ft
25.	type of fuel tank upper header, 1 - 4	
26.	type of fuel tank lower header, 1 - 4	
27.	not used	
28.	not used	
29.	not used	
30.	not used	

Miscellaneous Engine Data

MISCi

1.	number of engines	
2.	turbine exhaust thrust	lb
3.	optimum thrust	lb
4.	uprated thrust multiplier indicator 0.0 or 1.0	
5.	thrust build-up interval	sec
6.	oxidizer shutdown volume	ft^3

7.	fuel shutdown volume	ft^3
8.	convergence tolerance for shutdown volumes	ft^3
9.	thrust decay interval	sec
10.	miscellaneous stage lube, etc., flow rate	lb/sec
11.	miscellaneous stage oxidizer flow rate	lb/sec
12.	miscellaneous stage fuel flow rate	lb/sec
13.	miscellaneous engine oxidizer flow rate	lb/sec
14.	miscellaneous engine fuel flow rate	lb/sec
15.	oxidizer feed line hydraulic resistance	sec^2/ft^5
16.	fuel feed line hydraulic resistance	sec^2/ft^5
17.	thrust chamber nozzle throat area	in^2
18.	thrust chamber nozzle expansion ratio	
19.	nozzle efficiency	
20.	combustion efficiency	
21.	initial oxidizer temperature	^oR
22.	oxidizer temperature change scale factor	

23.	initial fuel temperature	$^{\circ}$R
24.	fuel temperature change scale factor	
25.	volume of oxidizer below bulkhead	ft^3
26.	volume of fuel below bulkhead	ft^3
27.	height of oxidizer column	ft
28.	height of fuel column	ft
29.	parameter for lift-off, stage 1	
30.	value of parameter for lift-off	

Tabular Data

All tabular data is entered in separate arrays of argument values and function values. The argument values are ordered in increasing values with the exception of the maximum value which is entered as the first and last item. The function values are arranged corresponding to their arguments. All tables are a maximum of 30 entries.

It is also possible to specify the degree of interpolation desired (from 1 to 3) and the variable to be used for the argument for most tables. If not input, a linear interpolation will be assumed using TIME for the argument variable.

Normal force coefficient, axial force coefficient, pitch angle or pitch angle of attack, yaw angle or yaw angle of attack, thrust, and total flow rate may be specified in tables. The table names, argument option indicators, and interpolation degree indicators are related as follows:

Argument Table	Function Table	Argument Indicator	Interpolation Indicator
MACH i	CSBN i	None	JCSBN (i)
MACH i	CSBA i	None	JCSBA (i)
TPIT i	PIT i	LPIT (i)	JPIT (i)
TYAW i	YAW i	LYAW (i)	JYAW (i)
TTRS i	TRUS i	LTRUS (i)	JTRUS (i)
TWDT i	WDOT i	LWDOT (i)	JWDOT (i)

MISCELLANEOUS OPTIONAL DATA

Tape Specifications

Tape specifications must be input as integers as they are not fixed by the program before use. If non-standard FORTRAN tape units are specified, the appropriate tape control cards must be included with the program deck. A unit specification of 0 indicates on-line, and negative indicates both on and off-line for output. The preassembled values will be used until changed by input:

ITAPE = input tape unit (5)

OTAPE = standard output tape unit (6)

STAPE = summary output tape unit (6)

Extra Print Line

PRTON is an array of Dimension 6 where input is of the A-prefix type. The parameters in PRTON will occur on STAPE just after the one-line burnout print.

PRTOF is an array of Dimension 6 where input is of the A-prefix type. The parameters in PRTOF will occur along with the standard trajectory printout.

Dump Option Data

Ten dump arrays of Dimension 10 are available for dumping COMMON blocks at various times during execution. The elements of the array are the A-prefix type where the address specifies the start of the block, the decrement the length and the tag the type of dump. The COMMON block names are in the symbol table, giving the location and length of the block.

The types of dumps available are:

0 = octal

1 = real

2 = integer

3 = Hollerith

The dumps are written on OTAPE. The dump arrays are as follows:

DUMP0	=	executed after reading input	In GUIDON
DUMP1	=	executed after a straight trajectory	
DUMP2	=	executed in case of main error	
DUMP3	=	executed after setting up stage and section data	In TRAJ
DUMP4	=	executed at discontinuities	
DUMP5	=	executed after each integration step	
DUMP6	=	executed at end of flight	
DUMP7	=	executed in case of trajectory error	
DUMP8	=	executed at end of derivative calculations	In NEWTON
DUMP9	=	executed in case of trajectory error before aborting integration	

SEARCH CONTROL DATA

A-prefix items denoted by (A).

a. GAINS routine input:

FLPAR = non-zero to execute GAINS
IVAR = independent variables (up to 10) (A)
DVAR = dependent variables (up to 10) (A)

b. Parameter study input:

PSTDY

1. parameter 1 - first location (A)
2. parameter 1 - second location (A)
3. parameter 1 - third location (A)
4. parameter 1 delta
5. number of times to increment parameter 1
6. parameter 2 - first location (A)
7. parameter 2 - second location (A)
8. parameter 2 - third location (A)
9. parameter 2 - delta
10. number of times to increment parameter 2

c. One-parameter search input:

1PRS1

1. independent variable - location 1 (A)
2. independent variable - location 2 (A)
3. independent variable - location 3 (A)
4. dependent variable (A)

5. desired value of dependent variable
6. % increment - independent variable
7. convergence tolerance
8. slope of dependent variable at solution
9. max iteration
10. max consecutive errors
11. max imaginary solutions
12. error scale factor

d. Two-parameter search input:

2PRS1

1. independent variable 1 - location 1 (A)
2. independent variable 1 - location 2 (A)
3. independent variable 1 - location 3 (A)
4. independent variable 2 (A)
5. dependent variable 1 (A)
6. dependent variable 2 (A)
7. desired value - dependent variable 1
8. desired value - dependent variable 2
9. % increment - independent variable 1
10. % increment - independent variable 2
11. convergence tolerance - variable 1
12. convergence tolerance - variable 2
13. Jacobian counter
14. scale factor for point 0
15. scale factor for points 1, 2
16. error stop point 0

17. error stop point 1
18. error stop point 2
19. damping scale factor
20. number of times to damp
21. max iterations

e. Min-max search input:

MNMX1

1. independent variable - location 1 (A)
2. independent variable - location 2 (A)
3. independent variable - location 3 (A)
4. dependent variable (A)
5. min-max indication - 1.0 min, + 1.0 max
6. % increment
7. convergence tolerance
8. max iterations
9. max errors
10. error scale factor

f. Second min-max input block for double min-max search:

MNMX2 — see MNMX1 for definitions.

g. Second one-parameter search input block for double one-parameter search:

1PRS2 — see 1PRS1 for definitions.

h. Propellant loading input:

PROPD

1. max iterations
2. max consecutive errors
3. option indicator

   0 = missile weight varies with loading

   $\neq 0$ = payload weight varies for constant total weight

4. number of stages for weight build-up

   < 0 = use max propellant weight on nominal trajectory

5. $\neq 0$ to load stage 1

   < 0 stage does not end on depletion

6. structure weight stage 1
7. oxidizer ullage volume
8. fuel ullage volume
9. same as 5 for stage 2
10. same as 6 for stage 2
11. same as 7 for stage 2
12. same as 8 for stage 2
13. same as 5 for stage 3
14. same as 6 for stage 3

15. same as 7 for stage 3

16. same as 8 for stage 3

17. same as 5 for stage 4

18. same as 6 for stage 4

19. same as 7 for stage 4

20. same as 8 for stage 4

21. same as 5 for stage 5

22. same as 6 for stage 5

23. same as 7 for stage 5

24. same as 8 for stage 5

25. same as 5 for stage 6

26. same as 6 for stage 6

27. same as 7 for stage 6

28. same as 8 for stage 6

i. Values of FLTR for various options

0 = straight trajectory

1 = one-parameter search

2 = two-parameter search

3 = min-max search

4 = not defined

5 = propellant loading

6 = double min-max search

7 = min-max/one-parameter search

8 = min-max/two-parameter search

9 = not defined

10 = min-max/propellant loading

11 = double one-parameter search

12 = one-parameter/two-parameter search

13 = one-parameter/min-max search

14 = not defined

15 = one-parameter/propellant loading

16 = propellant loading/double min-max search

17 = propellant loading/one-parameter search

18 = propellant loading/two-parameter search

19 = propellant loading/min-max search

20 = not defined

21 = two-parameter/one-parameter search

22 = not defined

23 = two-parameter/min-max search

24 = not defined

25 = two-parameter/propellant loading search

26 = one-parameter/two-parameter/propellant loading

27 = one-parameter/min-max/propellant loading

28 = not defined

29 = two-parameter/one-parameter/propellant loading

30 = two-parameter/min-max/propellant loading

31 = min-max/one-parameter/propellant loading

32 = min-max/two-parameter/propellant loading

33 = double min-max/propellant loading

34 = not defined

35 = propellant loading/one-parameter/two-parameter

36 = propellant loading/one-parameter/min-max

37 = propellant loading/two-parameter/one-parameter

38 = propellant loading/two-parameter/min-max

39 = propellant loading/min-max/one-parameter

40 = propellant loading/min-max/two-parameter

< + 0 = parameter study also

Figure 7 cross references the required input data as a function of the desired thrust and attitude options. Figure 8 cross references the required search data as a function of the desired type of run. Sample input data sheets are shown in the Appendix.

			OPTIONS													DESCRIPTION
			ALL	THIND						ATIND						
				0	1	2	3	4	5	0	1	2	3	4	5	
INPUT DATA	FLAGS	FLTR	X													Flag for type of run
		FLSC	X													Number of sections
		FLPRT	X													Print option
	INITIAL CONDITIONS	TI	X													Initial time
		VI	X													Initial velocity
		HI	X													Initial Altitude
		BETAI	X													Initial β
		AZI	X													Initial Azimuth
		LATI	X													Initial latitude
		LONGI	X													Initial longitude
		WI	X													Initial weight
		WPL	X													Payload weight
		BETA0	X													Kickover after vertical rise
		DATE	O													Launch date
		SIDET	O													Initial sidereal time
		TARGT	O													Target body
		TRANS	O													Transfer time
	SECTION DATA	SEX	X													Parameter to end section
		2	X													Value to end section
		3	O													Initial or jettison weight
		4	X													Stage indicator
		5	X													Thrust indicator
		6	X													Attitude indicator
		7	X													Integration interval
		8	X													Print interval
		9										X				Pitch rate
		10										X				Yaw rate
		11										X				Pitch kick
		12										X				Yaw kick
		13													X	X component of velocity to go
		14													X	Y component of velocity to go
		15													X	Z component of velocity to go
		16	O													Section number
		17	O													Parameter for section skip
		18	O													Value for section skip
		19	O													Routine to execute before moving stage data
		20	O													Routine to execute after moving stage data

X — denotes required data

O — denotes optional data

FIGURE 7. INPUT DATA/THRUST AND ATTITUDE OPTION CROSS REFERENCE

		OPTIONS													DESCRIPTION
			THIND						ATIND						
		ALL	0	1	2	3	4	5	0	1	2	3	4	5	
INPUT DATA	STAGE DATA	STAG	X												Reference Area
		2	O												Initial or jettison weight
		3					X	X							Initial oxidizer weight
		4					X	X							Initial fuel weight
		5	O												Monopropellant weight
		6	O												Burnout weight
		7			X										Vacuum thrust
		8			X										Flow rate
		9			X		X		X						Expansion ratio
		10			X		X		X						Throat area
		11					X		X						Specific impulse
		12				O	O								Thrust multiplier
		13				O									Flow rate multiplier
		14										O	O		Pitch multiplier
		15										O	O		Yaw multiplier
		16	O												Normal force multiplier
		17	O												Axial force multiplier
	ENGINE DATA	YBBR					X	X							Specific values of dependent variables
		YBAR					X	X							Nominal values of dependent variables
		XNOM					X	X							Nominal values of independent variables
		KPC					X	X							Chamber pressure influence coefficients
		KWOC					X	X							Chamber oxidizer flow rate influence coeff.
		KWFC					X	X							Chamber fuel flow rate influence coeff.
		KWOG					X	X							Gas gen. oxidizer flow rate influence coeff.
		KWFG					X	X							Gas gen. fuel flow rate influence coeff.
		CTOX					X	X							Oxidizer temperature coefficients
		CTFL					X	X							Fuel temperature coefficients
		TNKS					X	X							Propellant tank data
		MISC					X	X							Miscellaneous data
	TABULAR DATA	MACH	X												Mach number
		CSBN	X												Normal force coefficient
		CSBA	X												Axial force coefficient
		TPIT										X	X		Pitch argument
		PIT										X	X		Pitch function
		TYAW										X	X		Yaw argument
		YAW										X	X		Yaw function
		TTRS				X	X		X						Thrust argument
		TRUS				X	X		X						Thrust
		TWDT				X									Flow rate argument
		WDOT				X									Flow rate

X — denotes required data

O — denotes optional data

FIGURE 7. INPUT DATA/THRUST AND ATTITUDE OPTION CROSS REFERENCE (cont.)

INPUT DATA	TYPE OF RUN FLAG, FLTR																																								
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
NONE	X																																								
1PRS1		X						X				X	X	X		X		X				X					X	X		X		X				X	X	X		X	
2PRS1			X						X				X						X			X		X		X	X			X	X		X			X		X	X		X
MNMX1				X			X	X	X		X			X			X			X				X				X			X	X	X	X			X		X	X	X
MNMX2							X										X																	X							
1PRS2												X																													
PROPD						X					X					X	X	X	X	X						X	X	X		X	X	X	X	X		X	X	X	X	X	X

NOTE: A negative value of FLTR will also yield a parameter study. If this is desired, the data array PSTDY must also be input.

FLTR = 4, 9, 14, 20, 22, 24, 28 or 34 not defined

FIGURE 8 . SEARCH DATA/FLTR CROSS REFERENCE

IV. SUBROUTINE GUIDON

Subroutine GUIDON is the main control for Block II. Figure 9 shows the generalized Block II subroutine linkage.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL GUIDON (CFLAG)

Subroutine GUIDON contains COMMON blocks STUDY, SRCH1A, SRCH2A, MNX1, MNX2, SRCH1B, PROPL, TAPES, CONST, INPUT, IPRINT, GAIND, WFLAG, WORKD, SAVE, GILL, DATA, COMP, TABLES, ENGINI, ENGINO, ENGINW and DUMPY.

SOLUTION METHOD

1. If LLIST $\neq$ 0, go to 6

   If LLIST = 0, go to 2

Preset interpolation argument option to time:

2. LTIME = LOC(TIME)
3. Do 4, I = 1, 24
4. LLIST(I) = LTIME

Write the print key:

```
5.  CALL PRNTKY
```

Set input error flag FALSE:

```
6.  ERROR = .FALSE.
```

Read in input for one case:

```
7.  CALL FINK (ITAPE, OTAPE, ERROR, BCIPZE, HEAD)
```

Check for input error:

```
8.  If ERROR is true, go to 48
    If ERROR is not true, go to 9
```

Check for execution:

```
9.  If CFLAG < 0, go to 48
    If CFLAG ≥ 0, go to 10
```

Convert FLTR, FLSC, stage indicators, thrust indicators, attitude indicators and section numbers to fixed point:

```
10.  CALL FXINP (51, FLTR, FLSC, TARGET, SECTS(4, 1), SECTS(5, 1),
                SECTS(6, 1), SECTS(16, 1), SECTS(4, 2),
                SECTS(5, 2), SECTS(6, 2),  SECTS(16, 2),
                SECTS(4, 3), SECTS(5, 3),  SECTS(6, 3),
                SECTS(16, 3), SECTS(4, 4), SECTS(5, 4),
                SECTS(6, 4), SECTS(16, 4), SECTS(4, 5),
                SECTS(5, 5), SECTS(6, 5),  SECTS(16,5),
                SECTS(4, 6), SECTS(5, 6),  SECTS(6, 6),
```

```
                    SECTS(16, 6), SECTS(4, 7),  SECTS(5, 7),
                    SECTS(6, 7),  SECTS(16, 7), SECTS(4, 8),
                    SECTS(5, 8),  SECTS(6, 8),  SECTS(16, 8),
                    SECTS(4, 9),  SECTS(5, 9),  SECTS(6, 9),
                    SECTS(16, 9), SECTS(4, 10), SECTS(5, 10),
                    SECTS(6, 10), SECTS(16, 10), SECTS(4, 11),
                    SECTS(5, 11), SECTS(6, 11), SECTS(16, 11),
                    SECTS(4, 12), SECTS(5, 12), SECTS(6, 12),
                    SECTS(16, 12)
```

Convert propellant indicators and tank pressure flags to fixed point:

```
11.  CALL FXINP (12, TANKS(1, 1), TANKS(4, 1), TANKS(1, 2),
                 TANKS(4, 2), TANKS(1, 3), TANKS(4, 3),
                 TANKS(1, 4), TANKS(4, 4), TANKS(1, 5),
                 TANKS(4, 5), TANKS(1, 6), TANKS(4, 6)
```

Convert interpolation type indicators to fixed point:

```
12.  CALL FXINQ (36, JCSBN(1))
```

Convert miscellaneous search routine indicators to fixed point:

```
13.  CALL FXINP (20, NDELP1, NDELP2, MXIT1, MXERF, MXERI,
                 JACOB, ES0, ES1, ES2, MXIT2, MNXIT, MNXER,
                 MNXIT2, MNXER2, MXIT12, MXERF2, MXERI2,
                 PROPD(1), PROPD(2), PROPD(4))
```

Dump blocks listed in DUMP0 vector if desired:

```
14.  CALL DUMPS (DUMP0, 10, OTAPE)
```

Set working flag for type-of-run:

```
15.  FLTRW  =  | FLTR |
```

```
Check for parameter study:

16.  If ISIGN (1, FLTR) $>$ 0,  go to 21
     If ISIGN (1, FLTR) $\leq$ 0,  go to 17

Set flag for initial entry to Subroutine GAINS:

17.  KFLAG = -1

Check for GAINS execution:

18.  If FLPAR $\neq$ 0,  CALL GAINS (KFLAG)
     If FLPAR = 0,  go to 19
19.  CALL PSTUDY (KFLAG)

Check for completion of parameter study:

20.  If KFLAG $\neq$ 0,  go to 48
     If KFLAG = 0,  go to 21

Determine type-of-run:

21.  If FLTRW = 0,  go to 30
     If FLTRW $\neq$ 0,  go to 22

Fly nominal trajectory with printing:

22.  IFLTRW = FLTRW
23.  FLTRW = 0
```

```
24.  CALL TRAJ

25.  FLTRW = IFLTRW

Execute desired search routines:

26.  CALL MFLTR(FLTRW)

Check error flag:

27.  CALL CHKERR(Z)

28.  If Z ≠ 0, go to 38
     If Z = 0, go to 29

29.  Go to 36

Fly straight trajectory:

30.  CALL TRAJ

Dump blocks listed in DUMP1 vector if desired:

31.  CALL DUMPS (DUMP1, 10, OTAPE)

Check error flag:

32.  CALL CHKERR(Z)

33.  If Z ≠ 0, go to 38
     If Z = 0, go to 34

Check for parameter study:
```

```
34.  If ISIGN(1, FLTR) $<$ 0,  go to 18
     If ISIGN(1, FLTR) $\geq$ 0,  go to 35
35.  Go to 48
36.  FLTRW = 0
37.  Go to 30

Set output tape assignment:

38.  I = $|$ OTAPE $|$

Dump blocks listed in DUMP2 vector if desired:

39.  CALL DUMPS (DUMP2, 10, OTAPE)

Determine output mode(s):

40.  If I = 0,  go to 43
     If I $\neq$ 0,  go to 41
41.  Write error flag on unit I

Check for on-line print too.

42.  If OTAPE $>$ 0,  go to 44
     If OTAPE $\leq$ 0,  go to 43
43.  Print error flag on the on-line printer

Write output:
```

```
44.  CALL OUTPUT

Set error flag to zero:

45.  CALL SETERR(0)

Test for parameter study:

46.  If ISIGN (1, FLTR) $< 0$, go to 18
     If ISIGN (1, FLTR) $\geq 0$, go to 47

47.  Go to 48

48.  Return
```

SUBROUTINE GUIDON NOMENCLATURE

Symbol	Description	Units	Reference
BCIPZE	Entry point to Subroutine SYMTAB	—	EXTERNAL, 7
CFLAG	Error communication flag between Block I and Block II	—	CALL, 9
DUMP0	Input array of COMMON blocks	—	/DUMPY/, 14
DUMP1	Input array of COMMON blocks	—	/DUMPY/, 31
DUMP2	Input array of COMMON blocks	—	/DUMPY/, 39
ERROR	Input error flag	—	LOGICAL, 6, 7, 8
ES0	Error stop used in Subroutine SRCH2	—	/SRCH2A/, 13
ES1	Error stop used in Subroutine SRCH2	—	/SRCH2A/, 13
ES2	Error stop used in Subroutine SRCH2	—	/SRCH2A/, 13
FLPAR	Flag to execute GAINS routine	—	INTEGER, /GAIND/, 18
FLSC	Flag for number of sections	—	INTEGER, /INPUT/, 10
FLTR	Flag for type-of-run	—	INTEGER, /INPUT/, 10, 15, 16, 34, 46
FLTRW	Flag for type-of-run working	—	INTEGER, /WFLAG/, 15, 21, 22, 23, 25, 26, 36

SUBROUTINE GUIDON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
HEAD	Array of Hollerith headers for output	—	/INPUT/, 7
IFLTRW	Saved value of FLTRW	—	22, 25
I	Output tape	—	38, 40, 41
ITAPE	Input tape assinment	—	/TAPES/, 7
JACOB	Jacobian counter used in Subroutine SRCH2	—	/SRCH2A/, 13
JCSBN	Array of interpolation type indicators	—	/TABLES/, 12
KFLAG	Complete/not complete parameter study flag	—	17, 18, 19, 20
LLIST	Array of interpolation argument options	—	DIM, EQUIV, 1, 4
LTIME	Temporary storage for location of TIME	—	2, 4
MNXER2	Maximum allowable number of consecutive errors. Used in Subroutine DMNX	—	/MNX2/, 13
MNXER	Maximum allowable number of consecutive errors. Used in Subroutine MNX	—	/MNX1/, 13
MNXIT2	Maximum allowable number of iterations. Used in Subroutine DMNX	—	/MNX2/, 13

SUBROUTINE GUIDON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
MNXIT	Maximum allowable number of iterations. Used in Subroutine MNX	—	/MNX1/, 13
MXERF2	Maximum allowable number of consecutive errors during evaluation of dependent variable. Used in Subroutine DSRCH1	—	/SRCH1B/, 13
MXERF	Maximum allowable number of consecutive errors during evaluation of dependent variable. Used in Subroutine SRCH1	—	/SRCH1A/, 13
MXERI2	Maximum allowable number of consecutive imaginary solutions. Used in Subroutine DSRCH1	—	/SRCH1B/, 13
MXERI	Maximum allowable number of consecutive imaginary solutions. Used in Subroutine SRCH1	—	/SRCH1A/, 13
MXIT12	Maximum allowable number of iterations. Used in Subroutine DSRCH1	—	/SRCH1B/, 13
MXIT1	Maximum allowable number of iterations. Used in Subroutine SRCH1	—	/SRCH1A/, 13
MXIT2	Maximum allowable number of iterations. Used in Subroutine SRCH2	—	/SRCH2A/, 13

SUBROUTINE GUIDON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
NDELP1	Number of times to increment parameter 1	—	/STUDY/, 13
NDELP2	Number of times to increment parameter 2	—	/STUDY/, 13
OTAPE	Output tape assignment	—	INTEGER, /TAPES/, 7, 14, 31, 38, 39, 42
PROPD	Array of propellant loading input data	—	/PROPL/, 13
SECTS	Array of input section data	—	/INPUT/, 10
TANKS	Input array	—	/ENGINI/, 11
TARGET	Target body	—	/INPUT/, INTEGER, 10
TIME	Range time	sec	/GILL/, 2
Z	Error flag	—	27, 28, 32, 33, 41, 43

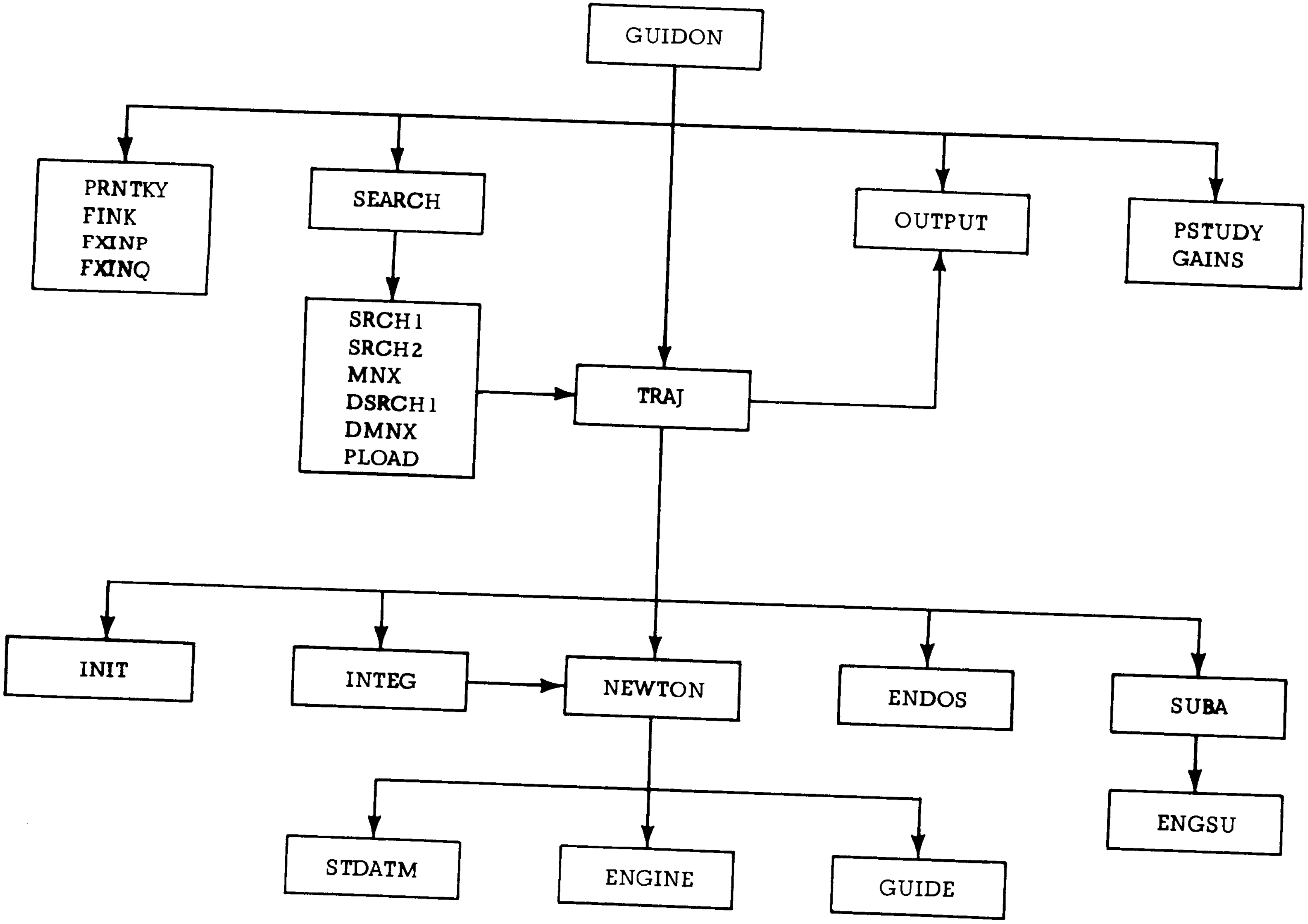


FIGURE 9. GENERALIZED BLOCK II SUBROUTINE LINKAGE

```
$IBFTC GUIDØN  FULIST,REF
      SUBRØUTINE GUIDØN(CFLAG)
      CØMMØN /STUDY/ P1ADD1,P1ADD2,P1ADD3,DELP1,NDELP1,
     $               P2ADD1,P2ADD2,P2ADD3,DELP2,NDELP2
      CØMMØN /SRCH1A/ XADD1,XADD2,XADD3,YADD,YSTAR,DELTAX,EPSI1,SLØPE,
     $               MXIT1,MXERF,MXERI,SCALE
      CØMMØN /SRCH2A/ X1ADD1,X1ADD2,X1ADD3,X2ADD,Y1ADD,Y2ADD,Y1STAR,
     $               Y2STAR,DELX1,DELX2,EPSI21,EPSI22,JACØB,K,L,ES0,ES1,
     $               ES2,KDAMP,NDAMP,MXIT2
      CØMMØN /MNX1/ VADD1,VADD2,VADD3,ZADD,MNXIND,DELV,MNXEP,MNXIT,
     $               MNXER,MNXSCL
      CØMMØN /MNX2/ VADD12,VADD22,VADD32,ZADD2,MNXIN2,DELV2,MNXEP2,
     $               MNXIT2,MNXER2,MNXSC2
      CØMMØN /SRCH1B/ XADD12,XADD22,XADD32,YADD2,YSTAR2,DELTX2,EPSI12,
     $               SLØPE2,MXIT12,MXERF2,MXERI2,SCALE2
      CØMMØN /PRØPL/ PRØPD(28)
      CØMMØN/TAPES/ITAPE,ØTAPE,STAPE
      INTEGER  ØTAPE,STAPE
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $               TAGS(7),INFIN,PI,PI02,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL  INFIN
      CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,
     $               DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $               SECTS(20,12),STAGS(20,6),HEAD(11,15)
      REAL  LATI,LØNGI
      INTEGER  TARGET
      INTEGER  FLTR,FLSC,FLPRT
      CØMMØN /IPRINT/ PRNTØN(6),PRNTØF(6)
      CØMMØN /GAIND/ FLPAR,IVAR(10),DVAR(10)
      INTEGER  FLPAR,DVAR
      CØMMØN/WFLAG/FLEØF,FLEØS,FLTP,FLDSC,TYSC,FLTRW
      LØGICAL FLEØF,FLEØS,FLTP
      INTEGER  FLDSC,TYSC,FLTRW
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $               PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $               PADSK,VESK,PADRT1,PADRT2,
     $            SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $              ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL  ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $               AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $               WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $            W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $            DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $            WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
```

FIGURE 10. SYMBOLIC LISTING OF SUBROUTINE GUIDON

```
$                   PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
$                   GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
$                   QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
$                   CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
$                   ESUBP,ESUBK,ERATIØ
 REAL   LAT,LØNG,MACH,LIFT
 CØMMØN/CØMP/THRSTX,THRSTY,THRSTZ,GRAVX,GRAVY,GRAVZ,
$                   DRAGX,DRAGY,DRAGZ,LIFTX,LIFTY,LIFTZ,
$                   VRX,VRY,VRZ
 REAL   LIFTX,LIFTY,LIFTZ
 CØMMØN/TABLES/TMACH(30,6),TCSBN(30,6),TCSBA(30,6),
$                   TTPIT(30,6),TPIT(30,6),TTYAW(30,6),TYAW(30,6),
$                   TTTRS(30,6),TTRUS(30,6),TTWDT(30,6),TWDØT(30,6),
$                   JCSBN(6),JCSBA(6),JPIT(6),JYAW(6),JTRUS(6),JWDØT(6),
$                   LPIT(6),LYAW(6),LTRUS(6),LWDØT(6),
$                   KMACH(6),KPIT(6),KYAW(6),KTRUS(6),KWDØT(6),
$                   IMACH(6),IPIT(6),IYAW(6),ITRUS(6),IWDØT(6)
 CØMMØN/ENGINI/ YBBAR(5,6),YBAR(5,6),XBAR(6,6),KPC(6,6),KWØC(6,6),
$              KWFC(6,6),KWØG(6,6),KWFG(6,6),CTØX(6,6),CTFL(6,6),
$              MISC(30,6),TANKS(30,6)
 REAL   KPC,KWØC,KWFC,KWØG,KWFG,MISC
 CØMMØN /ENGINØ/ PØ,PF,TØ,TF,CSTR,RHØØ,RHØF,PTØ,PTF,VØ,VF,HØ,HF,
$               WØCDT,WFCDT,MRC,PC,WØGDT,WFGDT,CVTH,CFTH,CSUBF,FCHMBR,
$               FENGIN,WØEDT,WFEDT,MRE,WEDØT,ISPE,WØPDT,WFPDT,
$               FS,WØSDT,WFSDT,MRS,WSDØT,ISPS
 REAL   MRC,MRE,MRS,ISPE,ISPS
 CØMMØN/ENGINW/PCBBR,WØCBB,WFCBB,WØGBB,WFGBB,
$                   PCBAR,WØCBR,WFCBR,WØGBR,WFGBR,
$                   PØNØM,PFNØM,TØNØM,TFNØM,FØPNØM,ETANØM,
$                   PIND,TREF,DREF,FPTPA,PTØG,PTFG,TSØ1,PTØ1,TSØ2,PTØ2,
$                   C1Ø,C2Ø,TSF1,PTF1,TSF2,PTF2,C1F,C2F,
$                   LØ,RØ,UHØ,LHØ,LF,RF,UHF,LHF(5),
$                   NØFE,FTURB,FØPT,KURF,DTBUP,VØCØFF,VFCØFF,EPVØL,
$                   DTDCY,WMSDT,WMØSD,WMFSD,WMØED,WMFED,RESØ,RESF,
$                   ASUBT,EPSI,ETACF,ETACS,TØI,CTØ,TFI,CTF,
$                   VØBB,VFBB,HØCØL,HFCØL,PARAML,VPARL,
$                   WKPC(6),WKWØC(6),WKWFC(6),WKWØG(6),WKWFG(6),
$                   TBLDUP,TEND,FLGHD,FLGSD,LSTAGE,CSTRN
 LØGICAL   FLGHD,FLGSD
 REAL   LØ,LF,NØFE,KURF
 INTEGER   PIND,FPTPA,UHØ,UHF
 CØMMØN /DUMPY/ DUMP0(10),DUMP1(10),DUMP2(10),DUMP3(10),DUMP4(10),
$                   DUMP5(10),DUMP6(10),DUMP7(10),DUMP8(10),DUMP9(10)
 CØMMØN/ASTRØ/KSUBS,KSUBE,ECLIP,KINVE
 REAL   KSUBS,KSUBE,KINVE
 CØMMØN /TØL/ AZB,DLATB,DLØNB,DLØNL,RAT,RMAS,TIMB
 CØMMØN /ØBART/ PLAT,PSIDT,VMF,TRANG,VMB
 CØMMØN/PBART/RXB,RYB,RZB,RXBD,RYBD,RZBD,
$                   RXR,RYR,RZR,RDXR,RDYR,RDZR,
$                   TX,TY,TZ,PITR,YAWR,SØX,SØY,SØZ,VR,
```

FIGURE 10. SYMBOLIC LISTING OF SUBROUTINE GUIDON (cont.)

```
     $             THEL,TCST
C
      LØGICAL  ERRØR
      DIMENSIØN  LLIST(24)
      EQUIVALENCE  (LLIST(1),LPIT(1))
      EXTERNAL  BCIPZE
      IF (LLIST .NE. 0)  GØ TØ 10
      LTIME = LØC(TIME)
      DØ 5  I=1,24
    5 LLIST(I) = LTIME
      CALL PRNTKY
C          READ INPUT
   10 ERRØR = .FALSE.
   20 CALL FINK(ITAPE,ØTAPE,ERRØR,BCIPZE,HEAD)
      IF (ERRØR)  GØ TØ 200
      IF (CFLAG .LT. 0.0)  GØ TØ 200
      CALL FXINP(51,FLTR,FLSC,TARGET,
     $             SECTS(4,1),SECTS(5,1),SECTS(6,1),SECTS(16,1),
     $             SECTS(4,2),SECTS(5,2),SECTS(6,2),SECTS(16,2),
     $             SECTS(4,3),SECTS(5,3),SECTS(6,3),SECTS(16,3),
     $             SECTS(4,4),SECTS(5,4),SECTS(6,4),SECTS(16,4),
     $             SECTS(4,5),SECTS(5,5),SECTS(6,5),SECTS(16,5),
     $             SECTS(4,6),SECTS(5,6),SECTS(6,6),SECTS(16,6),
     $             SECTS(4,7),SECTS(5,7),SECTS(6,7),SECTS(16,7),
     $             SECTS(4,8),SECTS(5,8),SECTS(6,8),SECTS(16,8),
     $             SECTS(4,9),SECTS(5,9),SECTS(6,9),SECTS(16,9),
     $             SECTS(4,10),SECTS(5,10),SECTS(6,10),SECTS(16,10),
     $             SECTS(4,11),SECTS(5,11),SECTS(6,11),SECTS(16,11),
     $             SECTS(4,12),SECTS(5,12),SECTS(6,12),SECTS(16,12))
      CALL FXINP(12,TANKS(1,1),TANKS(4,1),TANKS(1,2),TANKS(4,2),
     $              TANKS(1,3),TANKS(4,3),TANKS(1,4),TANKS(4,4),
     $              TANKS(1,5),TANKS(4,5),TANKS(1,6),TANKS(4,6))
      CALL FXINQ(36,JCSBN(1))
      CALL FXINP(20,NDELP1,NDELP2,MXIT1,MXERF,MXERI,JACØB,ESO,ES1,ES2,
     $        MXIT2,MNXIT,MNXER,MNXIT2,MNXER2,MXIT12,MXERF2,MXERI2,
     $        PRØPD(1),PRØPD(2),PRØPD(4))
C
      CALL DUMPS(DUMPO,10,ØTAPE)
      FLTRW = IABS(FLTR)
C          PARAMETER STUDY
      IF (ISIGN(1,FLTR) .GT. 0)  GØ TØ 50
      KFLAG = -1
   40 IF (FLPAR .NE. 0)  CALL GAINS(KFLAG)
      CALL PSTUDY(KFLAG)
      IF (KFLAG .NE. 0)  GØ TØ 200
   50 IF (FLTRW .EQ. 0)  GØ TØ 100
C        PRINT NØMINAL TRAJECTØRY
      IFLTRW = FLTRW
      FLTRW = 0
      CALL TRAJ
```

FIGURE 10. SYMBOLIC LISTING OF SUBROUTINE GUIDON (cont.)

```
      FLTRW = IFLTRW
C        EXECUTE SEARCH
      CALL MFLTR(FLTRW)
      CALL CHKERR(Z)
      IF (Z .NE. 0.0)  GØ TØ 160
      GØ TØ 150
C           STRAIGHT TRAJECTØRY
  100 CALL TRAJ
      CALL DUMPS(DUMP1,10,ØTAPE)
      CALL CHKERR(Z)
      IF (Z .NE. 0.0)  GØ TØ 160
      IF (ISIGN(1,FLTR) .LT. 0)  GØ TØ 40
      GØ TØ 200
C           FLY CØNVERGED TRAJECTØRY FØR PRINTING
  150 FLTRW = 0
      GØ TØ 100
C           ERRØR
  160 I = IABS(ØTAPE)
      CALL DUMPS(DUMP2,10,ØTAPE)
      IF (I .EQ. 0)  GØ TØ 161
      WRITE(I,1) Z
      IF (ØTAPE .GT. 0)  GØ TØ 162
  161 PRINT 1, Z
  162 CALL ØUTPUT
      CALL SETERR(0)
      IF (ISIGN(1,FLTR) .LT. 0)  GØ TØ 40
      GØ TØ 200
    1 FØRMAT(1H1,20X,21H******  ERRØR  ******,20X,A6)
C
  200 RETURN
      END
```

FIGURE 10. SYMBOLIC LISTING OF SUBROUTINE GUIDON (cont.)

V. SUBROUTINE SEARCH

SEARCH is the control region for the various search routines. It uses the input data to set up the calls to the desired search routines and is responsible for storing the independent variables in more than one location, executing any optional routines to calculate the dependent variables, converting the dependent variables to the proper units, error communication, and looking up the names of the variables for print-out by the search routines.

There are 2 entry points. This routine calls SETERR, CHKERR, UNPACK, LOOKUP, TRAJ, PSTDY, SRCH1, SRCH2, MNX, DSRCH1, DMNX and PLOAD.

1. Entry PSTUDY

PSTUDY sets up the call to PSTDY, the parameter study routine. If there is no parameter specified in P1ADD1, return is made to the calling program without executing PSTDY.

2. Entry MFLTR

MFLTR calls the correct type-of-run control area based on the value of the indicator (I) in the calling sequence — CALL MFLTR(I). The various type-of-run controls are as follows:

a. Control FLTR1

FLTR1 sets up the call to SRCH1, the one-parameter search routine. TRAJ1 is the function generator which executes the trajectory simulation and any conversions or optional routines.

b. Control FLTR2

FLTR2 sets up the call to SRCH2, the two-parameter search routine. TRAJ2 is the function generator which executes the trajectory simulation and any conversions or optional routines.

c. Control FLTR3

FLTR3 sets up the call to MNX, the min-max search routine. TRAJ3 is the function generator which executes the trajectory simulation and any conversions or optional routines.

d. Control FLTR5

FLTR5 calls PLOAD, the propellant loading routine specifying TRAJ to be the function generator.

e. Control FLTR6

FLTR6 sets up the call to DMNX, the second min-max search routine. TRAJ6 is the function generator which executes FLTR3 and any conversions or optional routines. The result is a double min-max search.

f. Control FLTR7

FLTR7 modifies TRAJ3 to call FLTR1 instead of the trajectory simulation. FLTR7 then calls FLTR3 which will now execute a min-max search driving a one-parameter search. Upon return to FLTR7, TRAJ3 is restored to call the normal trajectory simulation.

g. Control FLTR8

FLTR8 modifies TRAJ3 to call FLTR2 instead of the trajectory simulation. FLTR8 then calls FLTR3 which will now execute a min-max search driving a two-parameter search. Upon return to FLTR8, TRAJ3 is restored to call the normal trajectory simulation.

h. Control FLTR10

FLTR10 modifies TRAJ3 to call FLTR5, instead of the trajectory simulation. FLTR10 then calls FLTR3 which will now execute a min-max search driving a propellant loading optimization. Upon return to FLTR10, TRAJ3 is restored to call the normal trajectory simulation.

i. Control FLTR11

FLTR11 sets up the call to DSRCH1, the second one-parameter search routine. TRAJ11 is the function generator which executes FLTR1 and any conversions or optional routines. The result is a double one-parameter search.

j. Control FLTR12

FLTR12 modifies TRAJ1 to call FLTR2 instead of the trajectory simulation. FLTR12 then calls FLTR1 which will now execute a one-parameter search driving a two-parameter search. Upon return to FLTR12, TRAJ1 is restored to call the normal trajectory simulation.

k. Control FLTR13

FLTR13 modifies TRAJ1 to call FLTR3 instead of the trajectory simulation. FLTR13 then calls FLTR1 which will now execute a one-parameter search driving a min-max search. Upon return to FLTR13, TRAJ1 is restored to call the normal trajectory simulation.

m. Control FLTR15

FLTR15 modifies TRAJ1 to call FLTR5 instead of the trajectory simulation. FLTR15 then calls FLTR1 which will now execute a one-parameter search driving a propellant loading optimization. Upon return to FLTR15, TRAJ1 is restored to call the normal trajectory simulation.

n. Control FLTR16

FLTR16 calls PLOAD, the propellant loading routine specifying FLTR6 to be the function generator. The result is the propellant loading optimization driving a double min-max search.

o. Control FLTR17

FLTR17 calls PLOAD, the propellant loading routine specifying FLTR1 to be the function generator. The result is the propellant loading optimization driving a one-parameter search.

p. Control FLTR18

FLTR18 calls PLOAD, the propellant loading routine specifying FLTR2 to be the function generator. The result is the propellant loading optimization driving a two-parameter search.

q. Control FLTR19

FLTR19 calls PLOAD, the propellant loading routine specifying FLTR3 to be the function generator. The result is the propellant loading optimization driving a min-max search.

r. Control FLTR21

FLTR21 modifies TRAJ2 to call FLTR1 instead of the trajectory simulation. FLTR21 then calls FLTR2 which will now execute a two-parameter search driving a one-parameter search. Upon return to FLTR21, TRAJ2 is restored to call the normal trajectory simulation.

s. Control FLTR23

FLTR23 modifies TRAJ2 to call FLTR3 instead of the trajectory simulation. FLTR23 then calls FLTR2 which will now execute a two-parameter search driving a min-max search. Upon return to FLTR23, TRAJ2 is restored to call the normal trajectory simulation.

t. Control FLTR25

FLTR25 modifies TRAJ2 to call FLTR5 instead of the trajectory simulation. FLTR25 then calls FLTR2 which will now execute a two-parameter search driving a propellant loading optimization. Upon return to FLTR25, TRAJ2 is restored to call the normal trajectory simulation.

u. Control FLTR26

FLTR26 modifies TRAJ2 to call FLTR5 instead of the trajectory simulation. FLTR26 then calls FLTR12 which will now execute a one-parameter search driving a two-parameter search driving a propellant loading optimization. Upon return to FLTR26, TRAJ2 is restored to call the normal trajectory simulation.

v. Control FLTR27

FLTR27 modifies TRAJ3 to call FLTR5 instead of the trajectory simulation. FLTR27 then calls FLTR13 which will now execute a one-parameter search driving a min-max search driving a propellant loading optimization. Upon return to FLTR27, TRAJ3 is restored to call the normal trajectory simulation.

w. Control FLTR29

FLTR29 modifies TRAJ1 to call FLTR5 instead of the trajectory simulation. FLTR29 then calls FLTR21 which will now execute a two-parameter search driving a one-parameter search driving a propellant loading optimization. Upon return to FLTR29, TRAJ1 is restored to call the normal trajectory simulation.

x. Control FLTR30

FLTR30 modifies TRAJ3 to call FLTR5 instead of the trajectory simulation. FLTR30 then calls FLTR23 which will now execute a two-parameter search driving a min-max search driving a propellant loading optimization. Upon return to FLTR30, TRAJ3 is restored to call the normal trajectory simulation.

y. Control FLTR31

FLTR31 modifies TRAJ1 to call FLTR5 instead of the trajectory simulation. FLTR31 then calls FLTR7 which will now execute a min-max search driving a one-parameter search driving a propellant loading optimization. Upon return to FLTR31, TRAJ1 is restored to call the normal trajectory simulation.

z. Control FLTR32

FLTR32 modifies TRAJ2 to call FLTR5 instead of the trajectory simulation. FLTR32 then calls FLTR8 which will now execute a min-max search driving a two-parameter search driving a propellant loading optimization. Upon return to FLTR32, TRAJ2 is restored to call the normal trajectory simulation.

aa. Control FLTR33

FLTR33 modifies TRAJ3 to call FLTR5 instead of the trajectory simulation. FLTR33 then calls FLTR6 which will now execute a double-min-max search driving a propellant loading optimization. Upon return to FLTR33, TRAJ3 is restored to call the normal trajectory simulation.

bb. Control FLTR35

FLTR35 calls PLOAD, the propellant loading routine, specifying FLTR12 as the function generator. The result is the propellant loading optimization driving a one-parameter search driving a two-parameter search.

cc. Control FLTR36

FLTR36 calls PLOAD, the propellant loading routine, specifying FLTR13 as the function generator. The result is the propellant loading optimization driving a one-parameter search driving a min-max search.

dd. Control FLTR37

FLTR37 calls PLOAD, the propellant loading routine, specifying FLTR21 as the function generator. The result is the propellant loading optimization driving a two-parameter search driving a one-parameter search.

ee. FLTR38

FLTR38 calls PLOAD, the propellant loading routine specifying FLTR23 as the function generator. The result is the propellant loading optimization driving a two-parameter search driving a min-max search.

ff. FLTR39

FLTR39 calls PLOAD, the propellant loading routine specifying FLTR7 as the function generator. The result is the propellant loading optimization driving a min-max search driving a one-parameter search.

gg. FLTR40

FLTR40 calls PLOAD, the propellant loading routine specifying FLTR8 as the function generator. The result is the propellant loading optimization driving a min-max search driving a two-parameter search.

```
$IBMAP SEARCH
*
PRØG    EQU     1500
ØTAPE   EQU     TAPES+1
STAPE   EQU     TAPES+2
TAGS    EQU     CØNST+9
*
ADD     PZE
TAG     PZE
DEC     PZE
PRE     PZE
FLAG    PZE                                ERRØR FLAG
*
*
*       PARAMETER STUDY CØNTRØL
*
PSTUDY  SAVE
        CLA     3,4
        STA     CALL+12                    CØMPLETIØN FLAG
        NZT     P1ADD1
        TRA     PERRØR                     ERRØR - NØ INPUT
        CALL    LØC(=1)
        NZT     P2ADD1
        TRA     *+5
        CALL    LØC(=2)
        STA     CALL+3                     NØ ØF PARAMETERS
        CALL    UNPACK(P1ADD1,ADD)
        CLA     ADD
        STA     CALL+4                     PARAMETER 1 LØCATIØN
        CLA     P1ADD1
        STA     ADD
        CALL    LØØKUP(ADD,P1NAME)   PARAMETER 1 NAME
        LXD     P1ADD1,4
        TXL     *+2,4,0
        TXL     *+2,4,PRØG
        AXT     1,4
        SXA     P1NAME+1,4                 PARAMETER 1 SUBSCRIPT
        CALL    UNPACK(P2ADD1,ADD)
        CLA     ADD
        STA     CALL+8                     PARAMETER 2 LØCATIØN
        CLA     P2ADD1
        STA     ADD
        CALL    LØØKUP(ADD,P2NAME)   PARAMETER 2 NAME
        LXD     P2ADD1,4
        TXL     *+2,4,0
        TXL     *+2,4,PRØG
        AXT     1,4
        SXA     P2NAME+1,4                 PARAMETER 2 SUBSCRIPT
CALL    CALL    PSTDY(**,**,DELP1,NDELP1,P1NAME,**,DELP2,NDELP2,P2NAME,
        ETC     **,ØTAPE)
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH

```
        NZT     P1ADD2
        TRA     *+8
        CALL    UNPACK(P1ADD2,ADD)
        CLA*    CALL+4
        STØ*    ADD
        NZT     P1ADD3
        TRA     *+8
        CALL    UNPACK(P1ADD3,ADD)
        CLA*    CALL+4
        STØ*    ADD
        NZT     P2ADD2
        TRA     *+8
        CALL    UNPACK(P2ADD2,ADD)
        CLA*    CALL+8
        STØ*    ADD
        NZT     P2ADD3
        TRA     *+8
        CALL    UNPACK(P2ADD3,ADD)
        CLA*    CALL+8
        STØ*    ADD
        RETURN  PSTUDY
*
ERRØR   STL*    CALL+12
        RETURN  PSTUDY
*
P1NAME  BCI     1,
        PZE
P2NAME  BCI     1,
        PZE
*
STUDY   CØNTRL  STUDY
        USE     STUDY
        EVEN
P1ADD1  BSS     1                         PARAMETER 1 - LØCATIØN 1
P1ADD2  BSS     1                         PARAMETER 1 - LØCATIØN 2
P1ADD3  BSS     1                         PARAMETER 1 - LØCATIØN 3
DELP1   BSS     1                         PARAMETER 1 DELTA
NDELP1  BSS     1                         NØ ØF PERTURBATIØNS - PARAMETER 1
P2ADD1  BSS     1                         PARAMETER 2 - LØCATIØN 1
P2ADD2  BSS     1                         PARAMETER 2 - LØCATIØN 2
P2ADD3  BSS     1                         PARAMETER 2 - LØCATIØN 3
DELP2   BSS     1                         PARAMETER 2 DELTA
NDELP2  BSS     1                         NØ ØF PERTURBATIØNS - PARAMETER 2
*
        USE     PREVIØUS
*
*     MAIN CØNTRØL FØR SEARCHES
*
MFLTR   SAVE
        CLA*    3,4
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
        PAC     ,4
        TXL     *+7,4,-NFLTR-1
        CLA     LIST,4
        TZE     *+5
        STA     *+1
        CALL    **
        RETURN  MFLTR
*
LIST    PZE     TRAJ
        PZE     FLTR1                   ØNE-PARAMETER SEARCH
        PZE     FLTR2                   TWØ-PARAMETER SEARCH
        PZE     FLTR3                   MIN-MAX SEARCH
        PZE
        PZE     FLTR5                   PRØPELLANT LØADING
        PZE     FLTR6                   DØUBLE MIN-MAX SEARCH
        PZE     FLTR7                   MIN-MAX/ØNE-PARAMETER SEARCH
        PZE     FLTR8                   MIN-MAX/TWØ-PARAMETER SEARCH
        PZE
        PZE     FLTR10                  MIN-MAX/PRØPELLANT LØADING SEARCH
        PZE     FLTR11                  DØUBLE ØNE-PARAMETER SEARCH
        PZE     FLTR12                  ØNE-PARAMETER/TWØ-PARAMETER SEARCH
        PZE     FLTR13                  ØNE-PARAMETER/MIN-MAX SEARCH
        PZE
        PZE     FLTR15                  ØNE-PARAMETER/PRØPELLANT LØADING SEARCH
        PZE     FLTR16                  PRØPELLANT LØADING/DØUBLE MIN-MAX
        PZE     FLTR17                  PRØPELLANT LØADING/ØNE PARAMETER SEARCH
        PZE     FLTR18                  PRØPELLANT LØADING/TWØ-PARAMTER SEARCH
        PZE     FLTR19                  PRØPELLANT LØADING/MIN-MAX SEARCH
        PZE
        PZE     FLTR21                  TWØ-PARAMETER/ØNE-PARAMETER SEARCH
        PZE
        PZE     FLTR23                  TWØ-PARAMETER/MIN-MAX SEARCH
        PZE
        PZE     FLTR25                  TWØ-PARAMETER/PRØPELLANT LØADING SEARCH
        PZE     FLTR26                  1P/2P/PL
        PZE     FLTR27                  1P/MM/PL
        PZE
        PZE     FLTR29                  2P/1P/PL
        PZE     FLTR30                  2P/MM/PL
        PZE     FLTR31                  MM/1P/PL
        PZE     FLTR32                  MM/2P/PL
        PZE     FLTR33                  MM/MM/PL
        PZE
        PZE     FLTR35                  PL/1P/2P
        PZE     FLTR36                  PL/1P/MM
        PZE     FLTR37                  PL/2P/1P
        PZE     FLTR38                  PL/2P/MM
        PZE     FLTR39                  PL/MM/1P
        PZE     FLTR40                  PL/MM/2P
NFLTR   EQU     40
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
*
*
*     FLTR = 1    ØNE-PARAMETER SEARCH CØNTRØL
*
FLTR1  SAVE
       CALL    UNPACK(XADD1,ADD)
       CLA     ADD
       STA     CALL1+6           VARIABLE LØCATIØN
       CLA     XADD1
       STA     ADD
       CALL    LØØKUP(ADD,XNAME)  VARIABLE NAME
       LXD     XADD1,4
       TXL     *+2,4,0
       TXL     *+2,4,PRØG
       AXT     1,4
       SXA     XNAME+1,4         VARIABLE SUBSCRIPT
       CLA     YADD
       STA     ADD
       CALL    LØØKUP(ADD,YNAME)  FUNCTIØN NAME
       LXD     YADD,4
       TXL     *+2,4,0
       TXL     *+2,4,PRØG
       AXT     1,4
       SXA     YNAME+1,4         FUNCTIØN SUBSCRIPT
CALL1  CALL    SRCH1(TRAJ1,EFLAG,XNAME,**,YNAME,YVALUE,YSTAR,DELTAX,
       ETC     EPSI1,SLØPE,MXIT1,MXERF,MXERI,SCALE,STAPE)
       CALL    SETERR(EFLAG)
       RETURN  FLTR1
*
*     FUNCTIØN GENERATØR CØNTRØL FØR ØNE-PARAMETER SEARCH
*
TRAJ1  SAVE
       NZT     XADD2
       TRA     *+8
       CALL    UNPACK(XADD2,ADD)
       CLA*    CALL1+6
       STØ*    ADD
       NZT     XADD3
       TRA     *+8
       CALL    UNPACK(XADD3,ADD)
       CLA*    CALL1+6
       STØ*    ADD
TRAJ1Z CALL    TRAJ
       CALL    CHKERR(EFLAG)
       ZET     EFLAG
       RETURN  TRAJ1             ERRØR
       CALL    UNPACK(YADD,ADD)
       CLA     ADD
       STA     TRAJ1B
       CLA     TAG
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
        STA     TRAJ1A
        CLA     DEC
        TZE     TRAJ1A
        STA     *+1
        CALL    **
        CALL    CHKERR(EFLAG)
        ZET     EFLAG
        RETURN  TRAJ1               ERRØR
TRAJ1A  AXC     **,4
        TXL     *+3,4,0
        LDQ     TAGS-1,4
        TRA     *+2
        LDQ     =1.0
TRAJ1B  FMP     **
        STØ     YVALUE
        RETURN  TRAJ1
*
XNAME   BCI     1,
        PZE
YNAME   BCI     1,
        PZE
YVALUE  PZE
*
SRCH1A  CØNTRL  SRCH1A
        USE     SRCH1A
        EVEN
XADD1   BSS     1                   INDEPENDENT VARIABLE
XADD2   BSS     1
XADD3   BSS     1
YADD    BSS     1                   DEPENDENT VARIABLE
YSTAR   BSS     1                   DESIRED VALUE
DELTAX  BSS     1                   PER-CENT INCREMENT
EPSI1   BSS     1                   CØNVERGENCE TØLERANCE
SLØPE   BSS     1                   SLØPE
MXIT1   BSS     1                   MAX ITER
MXERF   BSS     1                   MAX ERRØR
MXERI   BSS     1                   MAX IMAG
SCALE   BSS     1                   ERRØR SCALE FACTØR
*
        USE     PREVIØUS
*
*      FLTR = 2   TWØ-PARAMETER SEARCH CØNTRØL
*
FLTR2   SAVE
        CALL    UNPACK(X1ADD1,ADD)
        CLA     ADD
        STA     CALL2+6             VARIABLE 1 LØCATIØN
        CLA     X1ADD1
        STA     ADD
        CALL    LØØKUP(ADD,X1NAME)  VARIABLE 1 NAME
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
       LXD     X1ADD1,4
       TXL     *+2,4,0
       TXL     *+2,4,PRØG
       AXT     1,4
       SXA     X1NAME+1,4          VARIABLE 1 SUBSCRIPT
       CALL    UNPACK(X2ADD,ADD)
       CLA     ADD
       STA     CALL2+8             VARIABLE 2 LØCATIØN
       CLA     X1ADD1
       STA     ADD
       CLA     X2ADD
       STA     ADD
       CALL    LØØKUP(ADD,X2NAME)  VARIABLE 2 NAME
       LXD     X2ADD,4
       TXL     *+2,4,0
       TXL     *+2,4,PRØG
       AXT     1,4
       SXA     X2NAME+1,4          VARIABLE 2 SUBSCRIPT
       CLA     Y1ADD
       STA     ADD
       CALL    LØØKUP(ADD,Y1NAME)  FUNCTIØN 1 NAME
       LXD     Y1ADD,4
       TXL     *+2,4,0
       TXL     *+2,4,PRØG
       AXT     1,4
       SXA     Y1NAME+1,4          FUNCTIØN 1 SUBSCRIPT
       CLA     Y2ADD
       STA     ADD
       CALL    LØØKUP(ADD,Y2NAME)  FUNCTIØN 2 NAME
       LXD     Y2ADD,4
       TXL     *+2,4,0
       TXL     *+2,4,PRØG
       AXT     1,4
       SXA     Y2NAME+1,4          FUNCTIØN 2 SUBSCRIPT
CALL2  CALL    SRCH2(TRAJ2,EFLAG,X1NAME,**,X2NAME,**,Y1NAME,Y1VALU,
       ETC     Y2NAME,Y2VALU,Y1STAR,Y2STAR,DELX1,DELX2,EPSI21,EPSI22,
       ETC     MXIT2,JACØB,K,L,ES0,ES1,ES2,KDAMP,NDAMP,STAPE)
       CALL    SETERR(EFLAG)
       RETURN  FLTR2
*
*      FUNCTIØN GENERATØR CØNTRØL FØR TWØ-PARAMETER SEARCH
*
TRAJ2  SAVE
       NZT     X1ADD2
       TRA     *+8
       CALL    UNPACK(X1ADD2,ADD)
       CLA*    CALL2+6
       STØ*    ADD
       NZT     X1ADD3
       TRA     *+8
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
       CALL     UNPACK(X1ADD3,ADD)
       CLA*     CALL2+6
       STØ*     ADD
TRAJ2Z CALL     TRAJ
       CALL     CHKERR(EFLAG)
       ZET      EFLAG
       RETURN   TRAJ2                  ERRØR
       CALL     UNPACK(Y1ADD,ADD)
       CLA      ADD
       STA      TRAJ2B
       CLA      TAG
       STA      TRAJ2A
       CLA      DEC
       TZE      TRAJ2A
       STA      *+1
       CALL     **
TRAJ2A AXC      **,4
       TXL      *+3,4,0
       LDQ      TAGS-1,4
       TRA      *+2
       LDQ      =1.0
TRAJ2B FMP      **
       STØ      Y1VALU
       CALL     CHKERR(EFLAG)
       ZET      EFLAG
       RETURN   TRAJ2                  ERRØR
       CALL     UNPACK(Y2ADD,ADD)
       CLA      ADD
       STA      TRAJ2D
       CLA      TAG
       STA      TRAJ2C
       CLA      DEC
       TZE      TRAJ2C
       STA      *+1
       CALL     **
       CALL     CHKERR(EFLAG)
       ZET      EFLAG
       RETURN   TRAJ2                  ERRØR
TRAJ2C AXC      **,4
       TXL      *+3,4,0
       LDQ      TAGS-1,4
       TRA      *+2
       LDQ      =1.0
TRAJ2D FMP      **
       STØ      Y2VALU
       RETURN   TRAJ2
*
X1NAME BCI      1,
       ZF
X2NAME BCI      1,
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
         PZE
X1NAME   BCI      1,
         PZE
X2NAME   BCI      1,
         PZE
X1VALU   PZE
X2VALU   PZE
*
*
SRCH2A   CØNTRL   SRCH2A
         USE      SRCH2A
         EVEN
X1ADD1   BSS      1                     INDEP. VAR. 1
X1ADD2   BSS      1
X1ADD3   BSS      1
X2ADD    BSS      1                      NDEP. VAR. 2
Y1ADD    BSS      1                     DEP. VAR. 1
Y2ADD    BSS      1                     DEP. VAR. 2
Y1STAR   BSS      1                     DESIRED VALUE - Y1
Y2STAR   BSS      1                     DESIRED VALUE - Y2
DELX1    BSS      1                     PER-CENT INCREMENT - X1
DELX2    BSS      1                     PER-CENT INCREMENT - X2
EPS121   BSS      1                     CØNVERGENCE TØLERANCES
EPS12    BSS      1
JACTR    BSS      1                     JACØBIAN CØUNTER
S        BSS      1                     SCALE FACTØRS
T        BSS      1
ES0      BSS      1                     ERRØR STØPS
ES1      BSS      1
ES2      BSS      1
KDAMP    BSS      1                     DAMPING FACTØRS
LDAMP    BSS      1
MXIT2    BSS      1                     MAX ITER
*
         USE      PREVIØUS
*
*       FLTR = 3   MIN-MAX SEARCH CØNTRØL
*
FLTR3   SAVE
        CALL     UNPACK(VADD1,ADD)
        CLA      ADD
        STA      CALL3+6            VARIABLE LØCATIØN
        CLA      VADD1
        STA      ADD
        CALL     LØØKUP(ADD,VNAME)  VARIABLE NAME
        LXD      VADD1,4
        TXL      *+2,4,0
        TXL      *+2,4,PRØG
        AXT      1,4
        SXA      VNAME+1,4          VARIABLE SUBSCRIPT
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
         CLA      ZADD
         STA      ADD
         CALL     LØØKUP(ADD,ZNAME)   FUNCTIØN NAME
         IXD      ZADD,4
         TXL      *+2,4,0
         TXL      *+2,4,PRØG
         AXT      1,4
         SXA      ZNAME+1,4          FUNCTIØN SUBSCRIPT
CALL3    CALL     MNX(TRAJ3,EFLAG,VNAME,**,ZNAME,ZVALUE,MNXIND,DELV,MNXEP,
         ETC      MNXIT,MNXER,MNXSCL,STAPE)
         CALL     SETERR(EFLAG)
         RETURN   FLTR3
*
*      FUNCTIØN GENERATØR CØNTRØL FØR MIN-MAX SEARCH
*
TRAJ3    SAVE
         NZT      VADD2
         TRA      *+8
         CALL     UNPACK(VADD2,ADD)
         CLA*     CALL3+6
         STØ*     ADD
         NZT      VADD3
         TRA      *+8
         CALL     UNPACK(VADD3,ADD)
         CLA*     CALL3+6
         STØ*     ADD
TRAJ3Z   CALL     TRAJ
         CALL     CHKERR(EFLAG)
         ZET      EFLAG
         RETURN   TRAJ3               ERRØR
         CALL     UNPACK(ZADD,ADD)
         CLA      ADD
         STA      TRAJ3B
         CLA      TAG
         STA      TRAJ3A
         CLA      DEC
         TZE      TRAJ3A
         STA      *+1
         CALL     **
         CALL     CHKERR(EFLAG)
         ZET      EFLAG
         RETURN   TRAJ3               ERRØR
TRAJ3A   AXC      **,4
         TXL      *+3,4,0
         LDQ      TAGS-1,4
         TRA      *+2
         LDQ      =1.0
TRAJ3B   FMP      **
         STØ      ZVALUE
         RETURN   TRAJ3
*
ZVALUE   PZE
*
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
VNAME   BCI      1,
        PZE
ZNAME   BCI      1,
        PZE
*
MNX1    CØNTRL   MNX1
        USE      MNX1
        EVEN
VADD1   BSS      1                     INDEPENDNET VARIABLE
VADD2   BSS      1
VADD3   BSS      1
ZADD    BSS      1                     DEPENDENT VARIABLE
MNXIND  BSS      1                     ØPTIØN
DELV    BSS      1                     PER-CENT INCREMENT
MNXEP   BSS      1                     CØNVERGENCE TØLERANCE
MNXIT   BSS      1                     MAX ITER
MNXER   BSS      1                     MAX ERRØR
MNXSCL  BSS      1                     ERRØR SCALE FACTØR
*
        USE      PREVIØUS
*
*       FLTR = 5   PRØPELLANT LØADING ØPTIMIZATIØN CØNTRØL
*
FLTR5   SAVE
        CALL     PLØAD(TRAJ)
        RETURN   FLTR5
*
*       FLTR = 6   DØUBLE MIN-MAX SEARCH CØNTRØL
*
FLTR6   SAVE
        CALL     UNPACK(VADD12,ADD)
        CLA      ADD
        STA      CALL6+6           VARIABLE LØCATIØN
        CLA      VADD12
        STA      ADD
        CALL     LØØKUP(ADD,VNAME2)  VARIABLE NAME
        LXD      VADD12,4
        TXL      *+2,4,0
        TXL      *+2,4,PRØG
        AXT      1,4
        SXA      VNAME2+1,4        VARIABLE SUBSCRIPT
        CLA      ZADD2
        STA      ADD
        CALL     LØØKUP(ADD,ZNAME2)  FUNCTIØN NAME
        LXD      ZADD2,4
        TXL      *+2,4,0
        TXL      *+2,4,PRØG
        AXT      1,4
        SXA      ZNAME2+1,4
CALL6   CALL     DMNX(TRAJ6,EFLAG,VNAME2,**,ZNAME2,ZVALU2,MNXIN2,DELV2,
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
        ETC     MNXEP2,MNXIT2,MNXER2,MNXSC2,STAPE)
        CALL    SETERR(EFLAG)
        RETURN  FLTR6
*
*      FUNCTIØN GENERATØR CØNTRØL FØR DØUBLE MIN-MAX SEARCH
*
TRAJ6   SAVE
        NZT     VADD22
        TRA     *+8
        CALL    UNPACK(VADD22,ADD)
        CLA*    CALL6+6
        STØ*    ADD
        NZT     VADD32
        TRA     *+8
        CALL    UNPACK(VADD32,ADD)
        CLA*    CALL6+6
        STØ*    ADD
        CALL    FLTR3
        CALL    CHKERR(EFLAG)
        ZET     EFLAG
        RETURN  TRAJ6                   ERRØR
        CALL    UNPACK(ZADD2,ADD)
        CLA     ADD
        STA     TRAJ6B
        CLA     TAG
        STA     TRAJ6A
        CLA     DEC
        TZE     TRAJ6A
        STA     *+1
        CALL    **
        CALL    CHKERR(EFLAG)
        ZET     EFLAG
        RETURN  TRAJ6                   ERRØR
TRAJ6A  AXC     **,4
        TXL     *+3,4,0
        LDQ     TAGS-1,4
        TRA     *+2
        LDQ     =1.0
TRAJ6B  FMP     **
        STØ     ZVALU2
        RETURN  TRAJ6
*
ZVALU2  PZE
*
VNAM2   BCI      1,
          PZE
ZNAME2  BCI      1,
          PZE
*
MNX2    CØNTRL  MNX2
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
         TRA      MNX2
         EVEN
VALU12   BSS      1
VALU22   BSS      1
VALU32   BSS      1
ZA[illegible] BSS 1
[illegible] BSS   1
DEL[illegible] BSS 1
MNX[illegible]Z2 BSS 1
MNXIT2   BSS      1
MNXIL2   BSS      1
MNXSC2   BSS      1
*
         USE      PREVIOUS
*
*        FILTR = 7   MIN-MAX -- ØNE-PARAMETER SEARCH CONTRØL
*
FILTR7   SAVE
         CLA      LIST+1
         STA      TRAJ3Z
         CALL     FILTR3
         CLA      LIST
         STA      TRAJ3Z
         RETURN   FILTR7
*
*        FILTR = 8   MIN-MAX -- TWØ-PARAMETER SEARCH CØNTRØL
*
FILTR8   SAVE
         CLA      LIST+2
         STA      TRAJ3Z
         CALL     FILTR3
         CLA      LIST
         STA      TRAJ3Z
         RETURN   FILTR8

*        FILTR = 10  MIN-MAX -- PRØPELLANT LØADING CØNTRØL
*
FILTR10  SAVE
         CLA      LIST+3
         STA      TRAJ3Z
         CALL     FILTR3
         CLA      LIST
         STA      TRAJ3Z
         RETURN   FILTR10
*
*
*        FILTR = 11   DØUBLE ØNE-PARAMETER SEARCH
*
FILTR11  SAVE
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
       CALL    UNPACK(XADD12,ADD)
       CLA     ADD
       STA     CALL11+6           VARIABLE LØCATIØN
       CLA     XADD12
       STA     ADD
       CALL    LØØKUP(ADD,XNAME2)  VARIABLE NAME
       LXD     XADD12,4
       TXL     *+2,4,0
       TXL     *+2,4,PRØG
       AXT     1,4
       SXA     XNAME2+1,4         VARIABLE SUBSCRIPT
       CLA     YADD2
       STA     ADD
       CALL    LØØKUP(ADD,YNAME2)
       LXD     YADD2,4
       TXL     *+2,4,0
       TXL     *+2,4,PRØG
       AXT     1,4
       SXA     YNAME2+1,4
CALL11 CALL    DSRCH1(TRAJ11,EFLAG,XNAME2,**,YNAME2,YVALU2,YSTAR2,
       ETC     DELTX2,EPSI12,SLØPE2,MXIT12,MXERF2,MXERI2,SCALE2,
       ETC     STAPE)
       CALL    SETERR(EFLAG)
       RETURN  FLTR11
*
*      FUNCTIØN GENERATØR CØNTRØL FØR DØUBLE ØNE-PARAMETER SEARCH
*
TRAJ11 SAVE
       NZT     XADD22
       TRA     *+8
       CALL    UNPACK(XADD22,ADD)
       CLA*    CALL11+6
       STØ*    ADD
       NZT     XADD32
       TRA     *+8
       CALL    UNPACK(XADD32,ADD)
       CLA*    CALL11+6
       STØ*    ADD
       CALL    FLTR1
       CALL    CHKERR(EFLAG)
       ZET     EFLAG
       RETURN  FLTR11             ERRØR
       CALL    UNPACK(YADD2,ADD)
       CLA     ADD
       STA     TRJ11B
       CLA     TAG
       STA     TRJ11A
       CLA     DEC
       TZE     TRJ11A
       STA     *+1
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
        CALL    **
        CALL    CHKERR(EFLAG)
        ZET     EFLAG
        RETURN  TRAJ11                  ERRØR
TRJ11A  AXC     **,4
        TXL     *+3,4,0
        LDQ     TAGS-1,4
        TRA     *+2
        LDQ     =1.0
TRJ11B  FMP     **
        STØ     YVALU2
        RETURN  TRAJ11
*
XNAME2  BCI     1,
        PZE
YNAME2  BCI     1,
        PZE
YVALU2  PZE
*
*
*
SRCH1B  CØNTRL  SRCH1B

        USE     SRCH1B
        EVEN
XADD12  BSS     1
XADD22  BSS     1
XADD32  BSS     1
YADD2   BSS     1
YSTAR2  BSS     1
DELTX2  BSS     1
EPSI12  BSS     1
SLØPE2  BSS     1
MXIT12  BSS     1
MXERF2  BSS     1
MXERI2  BSS     1
SCALE2  BSS     1
*
*
        USE     PREVIØUS
*
*       FLTR = 12    ØNE-PARAMETER - TWØ-PARAMETER SEARCH
*
FLTR12  SAVE
        CLA     LIST+2
        STA     TRAJ1Z
        CALL    FLTR1
        CLA     LIST
        STA     TRAJ1Z
        RETURN  FLTR12
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
*
*
*     FLTR = 13   0NE-PARAMETER - MIN-MAX SEARCH
*
FLTR13 SAVE
       CLA      LIST+3
       STA      TRAJ1Z
       CALL     FLTR1
       CLA      LIST
       STA      TRAJ1Z
       RETURN   FLTR13
*
*     FLTR = 15  0NE-PARAMTER -- PR0PELLANT L0ADING C0NTR0L
*
FLTR15 SAVE
       CLA      LIST+5
       STA      TRAJ1Z
       CALL     FLTR1
       CLA      LIST
       STA      TRAJ1Z
       RETURN   FLTR15
*
*
*     FLTR = 16   PR0PELLANT L0ADING/D0UBLE MIN-MAX SEARCH
*
FLTR16 SAVE
       CALL     PL0AD(FLTR6)
       RETURN   FLTR16
*
*
*     FLTR = 17   PR0PELLANT L0ADING/0NE-PARAMETER SEARCH
*
FLTR17 SAVE
       CALL     PL0AD(FLTR1)
       RETURN   FLTR17
*
*
*     FLTR = 18   PR0PELLANT L0ADING/TW0-PARAMETER SEARCH
*
FLTR18 SAVE
       CALL     PL0AD(FLTR2)
       RETURN   FLTR18
*
*
*     FLTR = 19   PR0PELLANT L0ADING/MIN-MAX SEARCH
*
FLTR19 SAVE
       CALL     PL0AD(FLTR3)
       RETURN   FLTR19
*
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
*
*      FLTR = 21   TW0-PARAMETER/0NE-PARAMETER SEARCH
*
FLTR21 SAVE
       CLA     LIST+1
       STA     TRAJ2Z
       CALL    FLTR2
       CLA     LIST
       STA     TRAJ2Z
       RETURN  FLTR21
*
*
*      FLTR = 23   TW0-PARAMETER/MIN-MAX SEARCH
*
FLTR23 SAVE
       CLA     LIST+3
       STA     TRAJ2Z
       CALL    FLTR2
       CLA     LIST
       STA     TRAJ2Z
       RETURN  FLTR23
*
*
*      FLTR = 25   TW0-PARAMETER/PR0PELLANT L0ADING SEARCH
*
FLTR25 SAVE
       CLA     LIST+5
       STA     TRAJ2Z
       CALL    FLTR2
       CLA     LIST
       STA     TRAJ2Z
       RETURN  FLTR25
*
*
*      FLTR = 26   0NE-PARAMETER/TW0-PARAMETER/PR0PELLANT L0ADING SEARCH
*
FLTR26 SAVE
       CLA     LIST+5
       STA     TRAJ2Z
       CALL    FLTR12
       CLA     LIST
       STA     TRAJ2Z
       RETURN  FLTR26
*
*
*      FLTR = 27   0NE-PARAMETER/MIN-MAX/PR0PELLANT L0ACING SEARCH
*
FLTR27 SAVE
       CLA     LIST+5
       STA     TRAJ3Z
       CALL    FLTR13
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
       CLA     LIST
       STA     TRAJ3Z
       RETURN  FLTR27
*
*
*      FLTR = 29   TWØ-PARAMETER/ØNE-PARAMETER/PRØPELLANT LØADING SEARCH
*
FLTR29 SAVE
       CLA     LIST+5
       STA     TRAJ1Z
       CALL    FLTR21
       CLA     LIST
       STA     TRAJ1Z
       RETURN  FLTR29
*
*
*      FLTR = 30   TWØ-PARAMETER/MIN-MAX/PRØPELLANT LØADING SEARCH
*
FLTR30 SAVE
       CLA     LIST+5
       STA     TRAJ3Z
       CALL    FLTR23
       CLA     LIST
       STA     TRAJ3Z
       RETURN  FLTR30
*
*
*      FLTR = 31   MIN-MAX/ØNE-PARAMETER/PRØPELLANT LØADING SEARCH
*
FLTR31 SAVE
       CLA     LIST+5
       STA     TRAJ1Z
       CALL    FLTR7
       CLA     LIST
       STA     TRAJ1Z
       RETURN  FLTR31
*
*
*      FLTR = 32   MIN-MAX/TWØ-PARAMETER/PRØPELLANT LØADING SEARCH
*
FLTR32 SAVE
       CLA     LIST+5
       STA     TRAJ2Z
       CALL    FLTR8
       CLA     LIST
       STA     TRAJ2Z
       RETURN  FLTR32
*
*
*      FLTR = 33   DØUBLE MIN-MAX/PRØPELLANT LØADING SEARCH
*
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

```
FLTR33 SAVE
       CLA      LIST+5
       STA      TRAJ3Z
       CALL     FLTR6
       CLA      LIST
       STA      TRAJ3Z
       RETURN   FLTR33
*
*
*      FLTR = 35    PRØPELLANT LØADING/ØNE-PARAMETER/TWØ-PARAMETER SEARCH
*
FLTR35 SAVE
       CALL     PLØAD(FLTR12)
       RETURN   FLTR35
*
*
*      FLTR = 36    PRØPELLANT LØADING/ØNE-PARAMETER/MIN-MAX SEARCH
*
FLTR36 SAVE
       CALL     PLØAD(FLTR13)
       RETURN   FLTR36
*
*
*      FLTR = 37    PRØPELLANT LØADING/TWØ-PARAMETER/ØNE-PARAMETER SEARCH
*
FLTR37 SAVE
       CALL     PLØAD(FLTR21)
       RETURN   FLTR37
*
*
*      FLTR = 38    PRØPELLANT LØADING/TWØ-PARAMETER/MIN-MAX SEARCH
*
FLTR38 SAVE
       CALL     PLØAD(FLTR23)
       RETURN   FLTR38
*
*
*      FLTR = 39    PRØPELLANT LØADING/MIN-MAX/ØNE-PARAMETER SEARCH
*
FLTR39 SAVE
       CALL     PLØAD(FLTR7)
       RETURN   FLTR39
*
*
*      FLTR = 40    PRØPELLANT LØADING/MIN-MAX/TWØ-PARAMETER SEARCH
*
FLTR40 SAVE
       CALL     PLØAD(FLTR8)
       RETURN   FLTR40
*
       END
```

FIGURE 11. SYMBOLIC LISTING OF SUBROUTINE SEARCH (cont.)

VI. SUBROUTINE SRCH1

The purpose of this subroutine is to converge on the value of an independent variable which matches a given value of a dependent variable.

Linear or quadratic interpolation or extrapolation may be used to predict the correct value of the independent variable. If a predicted value of an independent variable fails to yield a "valid" value for the dependent variable, this previous prediction is scaled by the factor SCALE.

CALLING SEQUENCE

The subroutine calling sequence is:

```
CALL SRCH1 (FUNCG, ERROR, XBCI, XADD, UBCI, UADD,
            USTAR, DELTAX, DELTAU, M, MXIT, MXERF,
            MXERI, SCALE, TAPE)
```

SOLUTION METHOD

Set indicators and save initial value of independent variable.

1. ERROR = 0

2. FLAG1 = 0

```
3.   FLAG2  = 0
4.   FLAG3  = 0
5.   SRCH1V = XADD
```

Determine output mode and write headings:

```
6.   ITAPE = $|TAPE|$
7.   If ITAPE = 0, go to 10
     If ITAPE $\neq$ 0, go to 8
8.   Write headings on output tape
9.   If TAPE $>$ 0, go to 11
     If TAPE $\leq$ 0, go to 10
10.  Write headings on on-line printer
```

Evaluate dependent variable:

```
11.  CALL FUNCG
```

Test if error flag was set during evaluation of dependent variable:

```
12.  If ERROR $\neq$ 0, go to 71
     If ERROR = 0, go to 13
```

Reset function generator error indicator to zero and save last three values of dependent variable:

```
13.  FLAG3 = 0
```

14. U2 = U1

15. U1 = U0

16. U0 = UADD

Write output:

17. N = 20

18. Go to 88

Test for convergence:

19. If U0 = U1, go to 83

If U0 $\neq$ U1, go to 20

Compute difference between current value of dependent variable and desired value:

20. UDELTA = U0 - USTAR

Test value of slope indicator, M. If M = 0, linear interpolation will be used. If M $<$ 0, the function has a negative slope in the region of interest. If M $>$ 0, the function has a positive slope in the region of interest:

21. If M = 0, go to 23

If M $\neq$ 0, go to 22

Test value of FLAG1. FLAG1 is an indicator used to denote the number of times this subroutine was entered:

```
22. If FLAG1 $\leq$ 2, go to 27
    If FLAG1 $>$ 2, go to 23
```

Test value of USTAR:

```
23. If USTAR $\neq$ 0, go to 25
    If USTAR $=$ 0, go to 24
```

Compare current value of dependent variable with desired value. If convergence criterion is not satisfied, re-evaluate independent variable:

```
24. If DELTAU $<$ $|$U0$|$, go to 27
    If DELTAU $\geq$ $|$U0$|$, go to 26
25. If $|$(DELTAU)(USTAR)$|$ $<$ $|$UDELTA$|$, go to 27
    If $|$(DELTAU)(USTAR)$|$ $\geq$ $|$UDELTA$|$, go to 26
26. ERROR = 0
    Return
```

Computations for Re-evaluation of Independent Variable

```
27. X2 = X1
28. X1 = X0
29. X0 = XADD
30. FLAG1 = FLAG 1 + 1
```

Determine number of times subroutine has been entered:

31. If FLAG1 $<$ 2, go to 32

If FLAG1 $=$ 2, go to 42

If FLAG1 $>$ 2, go to 45

First Pass

Increment independent variable:

32. XDELTA = (X0)(DELTAX)/(100)

33. X = X0 + XDELTA

34. XADD = X

Test for maximum number of iterations:

35. If FLAG1 $<$ MXIT, go to 11

If FLAG1 $\geq$ MXIT, go to 36

If maximum number of iterations has been exceeded, print error message and restore initial value of independent variable:

36. If ITAPE $=$ 0, go to 39

If ITAPE $\neq$ 0, go to 37

37. Write error message on output tape

38. If TAPE $>$ 0, go to 40

If TAPE $\leq$ 0, go to 39

39. Write error message on on-line printer

40. XADD = SRCH1V

```
41.  ERROR = HSRCH1
     Return
```

Second Pass

```
42.  OLDA = 0
43.  XDELTA = (UDELTA)(X1 - X0)/(U0 - U1)
44.  Go to 33
```

Third Pass or Greater

If M = 0, or the last two values of the independent or dependent variables are too close together, use linear interpolation:

```
45.  If M = 0, go to 42
     If M ≠ 0, go to 46
46.  If | X1 - X0 | < .001, go to 42
     If | X1 - X0 | ≥ .001, go to 47
47.  If | U1 - U0 | < .001, go to 42
     If | U1 - U0 | ≥ .001, go to 48
```

Calculate quadratic:

```
48.  DX1 = X1 - X0
49.  DX2 = X2 - X1
50.  DU1 = U1 - U0
51.  DU2 = U2 - U1
```

52. A = ((DU2)/(DX2) - (DU1)/(DX1))/(DX1 + DX2)

53. If OLDA = 0, go to 55

If ODLA $\neq$ 0, go to 54

54. If (A)(OLDA) $\leq$ 0, go to 42

If (A)(OLDA) $>$ 0, go to 55

55. OLDA = A

56. B = -(A)(X1 + X2) + (DU2)/(DX2)

57. C = U0 - USTAR + (X0)((-A)(X0) - B)

58. Z = B^2 - (4)(A)(C)

Check for imaginary solution:

59. If Z $<$ 0, go to 63

If Z $\geq$ 0, go to 60

60. X = -(B)/((2)(A)) + SIGN(($Z^{.5}$)/((2)(A)), (M)(A))

SIGN is a Fortran function which transfers the sign of the second argument to the first argument.

61. FLAG2 = 0

62. Go to 34

<u>Imaginary Solution</u>

63. If ITAPE = 0, go to 66

If ITAPE $\neq$ 0, go to 64

```
64.  Write error message on output tape
65.  If TAPE > 0, go to 67
     If TAPE ≤ 0, go to 66
66.  Write error message on on-line printer
67.  FLAG2 = FLAG2 + 1
68.  If FLAG2 > MXERI, go to 40
     If FLAG2 ≤ MXERI, go to 69
69.  X = -(B)/((2)(A))
70.  Go to 34
```

<u>Error Control</u>

```
71.  If ITAPE = 0, go to 74
     If ITAPE ≠ 0, go to 72
72.  Write error message on output tape
73.  If TAPE > 0, go to 75
     If TAPE ≤ 0, go to 74
74.  Write error message on on-line printer
75.  N = 820
76.  Go to 88
77.  FLAG3 = FLAG3 + 1
78.  If FLAG3 ≥ MXERF, go to 40
     If FLAG3 < MXERF, go to 79
```

Scale change in independent variable:

79. $X = X0 + (XADD - X0)(SCALE)$

80. If FLAG1 $= 0$, $X = (XADD)(SCALE)$

If FLAG1 $\neq 0$, $X = X0 + (XADD - X0)(SCALE)$

81. ERROR $= 0$

82. Go to 34

Zero perturbation error:

83. If ITAPE $= 0$, go to 86

If ITAPE $\neq 0$, go to 84

84. Write error message on output tape

85. If TAPE > 0, go to 87

If TAPE ≤ 0, go to 86

86. Write error message on on-line printer

87. Go to 40

Output

88. If ITAPE $= 0$, go to 91

If ITAPE $\neq 0$, go to 89

89. Write output on output tape

```
90. If TAPE $> 0$, go to 92
    If TAPE $\leq 0$, go to 91
91. Write output on on-line printer
92. If N $= 20$, go to 19
    If N $= 820$, go to 77
```

SUBROUTINE SRCH1 NOMENCLATURE

Symbol	Description	Units	Reference
A	Coefficient of quadratic equation	—	52, 54, 55, 56, 57, 58 60, 69
B	Coefficient of quadratic equation	—	56, 57, 58, 60, 69
C	Coefficient of quadratic equation	—	57, 58
DELTAU	Convergence tolerance	—	CALL, 24, 25
DELTAX	Per cent to increment independent variable on first pass	—	CALL, 32
DU1	—	—	50, 52
DU2	—	—	51, 52, 56
DX1	—	—	48, 52
DX2	—	—	49, 52, 56
ERROR	Error communication flag	—	CALL, 1, 12, 26, 41, 81
FLAG1	Indicator denoting number of times subroutine has been entered	—	INTEGER, 2, 22, 30, 31, 35, 80
FLAG2	Indicator denoting number of consecutive imaginary solutions	—	INTEGER, 3, 61, 67, 68
FLAG3	Indicator denoting number of consecutive errors obtained during evaluation of dependent variable	—	INTEGER, 4, 13, 77, 78

SUBROUTINE SRCH1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
FUNCG	Name of subroutine which evaluates dependent variable	—	CALL, 11
HSRCH1	—	—	DATA, 41
ITAPE	Absolute value of TAPE	—	6, 7, 36, 63, 71, 83, 88
M	Slope indicator. If M = 0, linear interpolation will be used. If M < 0, function has negative slope; if M > 0, function has positive slope in region of interest	—	CALL, REAL, 21, 45, 60
MXERF	Maximum allowable number of consecutive errors during evaluation of dependent variable	—	CALL, 78
MXERI	Maximum allowable number of consecutive imaginary solutions	—	CALL, 68
MXIT	Maximum allowable number of iterations	—	CALL, 35
N	—	—	17, 75, 92
OLDA	Last value of A	—	42, 53, 54, 55
SCALE	Scale factor for incrementing change in independent variable in case of error	—	CALL, 79, 80

SUBROUTINE SRCH1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
SRCH1V	Initial value of independent variable	—	5, 40
TAPE	Tape unit to write search data on	—	CALL, INTEGER, 6, 9, 38, 65, 73, 85, 90
U0	Current value of dependent variable	—	15, 16, 19, 20, 24, 43, 47, 50, 57
U1	Last value of dependent variable	—	14, 15, 19, 43, 47, 50, 51
U2	Next to last value of dependent variable	—	14, 51
UADD	Location of dependent variable	—	CALL, 16
UBCI	Name of dependent variable used for output	—	CALL, DIM
UDELTA	Difference between current value and desired value of dependent variable	—	20, 25, 43
USTAR	Desired value of dependent variable	—	CALL, 20, 23, 25, 57
X0	Current value of independent variable	—	28, 29, 32, 33, 43, 46, 48, 57, 79
X1	Last value of independent variable	—	27, 28, 43, 46, 48, 49, 56
X2	Next to last value of independent variable	—	27, 33, 49, 56

SUBROUTINE SRCH1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
X	New value of independent variable	—	33, 34, 60, 69, 79, 80
XADD	Location of independent variable	—	CALL, 5, 29, 34, 40, 79, 80
XBCI	Name of independent variable used for output	—	CALL, DIM
XDELTA	Change in independent variable	—	32, 33, 43
Z	$B^2 - (4)(A)(C)$	—	58, 59, 60

```
C        TEST FOR CONVERGENCE
   20 IF (UO .EQ. U1)  GO TO 850
      UDELTA = UO - USTAR
      IF (M .EQ. 0.0)  GO TO 30
      IF (FLAG1 .LE. 2)  GO TO 50
   30 IF (USTAR .NE. 0.0)  GO TO 40
      IF ((DELTAU - ABS(UO)) .LT. 0.0)  GO TO 50
      GO TO 45
   40 IF ((ABS(DELTAU * USTAR) - ABS(UDELTA)) .LT. 0.0)  GO TO 50
C        CONVERGED RETURN
   45 ERROR = 0.0
      RETURN
C
   50 X2 = X1
      X1 = XO
      XO = XADD
      FLAG1 = FLAG1 + 1
      IF (FLAG1 - 2)  100,200,300
C        FIRST PASS
  100 XDELTA = XO * DELTAX/100.0
  101 X = XO + XDELTA
C        STORE NEW VARIABLE AND TEST FOR MAX ITERATIONS
  110 XADD = X
      IF (FLAG1 .LT. MXIT )  GO TO 10
C        ERROR RETURN  --  RESTORE INITIAL VALUE OF VARIABLE
      IF (ITAPE .EQ. 0)  GO TO 115
      WRITE (ITAPE,1)
      IF (TAPE .GT. 0)  GO TO 150
  115 PRINT 1
    1 FORMAT(44H      MAX ITERATIONS -- ONE PARAMETER SEARCH)
  150 XADD = SRCH1V
      ERROR = HSRCH1
      RETURN
C
C        SECOND PASS
  200 OLDA = 0.0
      XDELTA = UDELTA * (X1 - XO)/(UO - U1)
      GO TO 101
C
C        THIRD OR GREATER PASS
  300 IF (M .EQ. 0.0)  GO TO 200
      IF (ABS(X1 - XO) .LT. .001)  GO TO 200
      IF (ABS(U1 - UO) .LT. .001)  GO TO 200
C        CALCULATE QUADRATIC
      DX1 = X1 - XO
      DX2 = X2 - X1
      DU1 = U1 - UO
      DU2 = U2 - U1
      A = ((DU2/DX2) - (DU1/DX1))/(DX1 + DX2)
      IF (OLDA .EQ. 0.0)  GO TO 310
C        CHECK FOR INFLECTION POINT
```

FIGURE 12. SYMBOLIC LISTING OF SUBROUTINE SRCH1 (cont.)

```
$IBFTC SRCH1    FULIST,REF
      SUBRØUTINE SRCH1(FUNCG,ERRØR,XBCI,XADD,UBCI,UADD,USTAR,DELTAX,
     $             DELTAU,M,MXIT,MXERF,MXERI,SCALE,TAPE)
C        ØNE PARAMETER SEARCH RØUTINE
C                FUNCG   - RØUTINE TØ EVALUATE FUNCTIØN
C                ERRØR   - ERRØR CØMMUNICATIØN FLAG
C                XBCI    - NAME ØF X
C                XADD    - LØCATIØN ØF VARIABLE
C                UBCI    - NAME ØF U
C                UADD    - LØCATIØN ØF FUNCTIØN
C                USTAR   - DESIRED VALUE ØF FUNCTIØN
C                DELTAX  - PER CENT INCREMENT
C                DELTAU  - CØNVERGENCE TØLERANCE
C                M       - SLØPE INDICATØR
C                MXIT    - MAX ITERATIØNS
C                MXERF   - MAX NØ ØF CØNSECUTIVE ERRØRS
C                MXERI   - MAX NØ ØF CØNSECUTIVE IMAGINARY SØLUTIØNS
C                SCALE   - SCALE FACTØR FØR DELTA IF ERRØR
C                TAPE    - TAPE UNIT TØ WRITE SEARCH DATA ØN
C
      REAL M
      INTEGER  TAPE
      DIMENSIØN  XBCI(2),UBCI(2)
      DATA  HSRCH1/5HSRCH1/
      INTEGER  FLAG1,FLAG2,FLAG3
C        SET FLAGS AND SAVE INITIAL VALUE ØF VARIABLE
      ERRØR = 0.0
      FLAG1 = 0
      FLAG2 = 0
      FLAG3 = 0
      SRCH1V = XADD
      ITAPE = IABS(TAPE)
      IF (ITAPE .EQ. 0)  GØ TØ 9
      WRITE (ITAPE,5)  UBCI,UBCI,XBCI
      IF (TAPE .GT. 0)  GØ TØ 10
    9 PRINT 5,  UBCI,UBCI,XBCI
    5 FØRMAT(1H0,31H       ØNE-PARAMETER SEARCH DATA/
     $        1H ,A6,1H(,I4,1H),3H(*),5X,2(A6,1H(,I4,1H),4X))
C
   10 CALL FUNCG
C        CHECK FØR ERRØR CØNDITIØN
      IF (ERRØR .NE. 0.0)  GØ TØ 800
C         RESET ERRØR TALLY
      FLAG3 = 0
      U2 = U1
      U1 = U0
      U0 = UADD
C        PRINT
      ASSIGN 20 TØ N
      GØ TØ 900
```

FIGURE 12. SYMBOLIC LISTING OF SUBROUTINE SRCH1

```
      IF ((A * ØLDA) .LE. 0.0)  GØ TØ 200
  310 ØLDA = A
      B = -A * (X1 + X2) + (DU2/DX2)
      C = U0 - USTAR + X0 * (-A * X0 - B)
      Z = B**2 - 4.0 * A * C
C         CHECK FØR IMAGINARY SØLUTIØN
      IF (Z .LT. 0.0)  GØ TØ 700
      X = -B/(2.0*A) + SIGN(SQRT(Z)/(2.0*A),M*A)
C          RESET IMAGINARY SØLUTIØN TALLY
      FLAG2 = 0
      GØ TØ 110
C
C         IMAGINARY SØLUTIØN
  700 IF (ITAPE .EQ. 0)  GØ TØ 705
      WRITE (ITAPE,2)
      IF (TAPE .GT. 0)  GØ TØ 710
  705 PRINT 2
    2 FØRMAT(24H      IMAGINARY SØLUTIØN)
  710 FLAG2 = FLAG2+1
      IF (FLAG2 .GT. MXERI)  GØ TØ 150
      X = -B/(2.0*A)
      GØ TØ 110
C
C         ERRØR CØNTRØL
  800 IF (ITAPE .EQ. 0)  GØ TØ 805
      WRITE (ITAPE,3)
      IF (TAPE .GT. 0)  GØ TØ 810
  805 PRINT 3
    3 FØRMAT(30H      FUNCTIØN GENERATØR ERRØR)
  810 ASSIGN 820 TØ N
      GØ TØ 900
  820 FLAG3 = FLAG3 + 1
      IF (FLAG3 .GE. MXERF)  GØ TØ 150
C         SCALE DELTA
      X = X0 + (XADD - X0) * SCALE
      IF (FLAG1 .EQ. 0)  X = XADD * SCALE
C          TURN ØFF ERRØR FLAG
      ERRØR = 0.0
      GØ TØ 110
C
C         ZERØ PERTURBATIØN -- ERRØR
  850 IF (ITAPE .EQ. 0)  GØ TØ 855
      WRITE (ITAPE,6)
      IF (TAPE .GT. 0)  GØ TØ 860
  855 PRINT 6
    6 FØRMAT(21H      NØ PERTURBATIØN)
  860 GØ TØ 150
C
C         WRITE SEARCH DATA ØN TAPE
  900 IF (ITAPE .EQ. 0)  GØ TØ 905
```

FIGURE 12. SYMBOLIC LISTING OF SUBROUTINE SRCH1 (cont.)

```
    WRITE (ITAPE,4)  USTAR,UO,XADD
    IF (TAPE .GT. 0)  GØ TØ 910
905 PRINT 4,  USTAR,UO,XADD
  4 FØRMAT(1H ,3E16.8)
910 GØ TØ N, (20,820)
    END
```

FIGURE 12. SYMBOLIC LISTING OF SUBROUTINE SRCH1 (cont.)

VII. SUBROUTINE DSRCH1

Subroutine DSRCH1 is identical to Subroutine SRCH1. It exists in order to allow for more than one single-parameter-search to be accomplished. That is, a separate deck is required for each single-parameter-search desired.

VIII. SUBROUTINE SRCH2

The purpose of this subroutine is to converge on two independent variables given the values of two functions. Each function may be dependent upon one or both of the independent variables.

CALLING SEQUENCE

The calling sequence is:

CALL SRCH2 (FUNCG, ERROR, X1BCI, X1ADD, X2BCI, X2ADD, U1BCI, U1ADD, U2BCI, U2ADD, U1STAR, U2STAR, DELTX1, DELTX2, DELTU1, DELTU2, MXIT2, JACOB, K, L, ES0, ES1, ES2, KDAMP2, NDAMP2, TAPE)

SOLUTION METHOD

Initialize variables:

1. ERROR = 0
2. V1A(1) = 0
3. V2A(1) = 0
4. V1B(3) = 0

```
5.  V2B(2) = 0
6.  FLAGS(1) = 0
7.  FLAGS(2) = 0
8.  FLAGS(3) = 0

Determine output mode and write headings:

9.  ITAPE = |TAPE|
10. If ITAPE = 0, go to 13
    If ITAPE ≠ 0, go to 11
11. Write headings on output tape
12. If TAPE > 0, go to 14
    If TAPE ≤ 0, go to 13
13. Write headings on the on-line printer

Initialize variables:

14. X1SAVE = X1ADD
15. V1A(2) = X1SAVE
16. V1A(3) = X1SAVE
17. V1B(1) = X1SAVE
18. V1(1) = X1SAVE
19. V1(3) = X1SAVE
20. X2SAVE = X2ADD
```

21. V2A(2) = X2SAVE

22. V2A(3) = X2SAVE

23. V2B(1) = X2SAVE

24. V2(1) = X2SAVE

25. V2(2) = X2SAVE

26. V1B(2) = (DELTX1)(V1(1))/(100)

27. V2B(3) = (DELTX2)(V2(1))/(100)

28. ESTOP(1) = ES0

29. ESTOP(2) = ES1

30. ESTOP(3) = ES2

31. SCALE(1) = K

32. SCALE(2) = L

33. SCALE(3) = L

34. V1(2) = V1(1) + V1B(2)

35. V2(3) = V2(1) + V2B(3)

36. DUMMY = 0

37. FLAG1 = 0

38. JFLAG = JACOB

39. I = 0

40. If JFLAG $<$ 0, I = 2

If JFLAG $\geq$ 0, I = 0

```
41.  J = 1

42.  X1ADD = V1(J)

43.  X2ADD = V2(J)
```

Increment iteration counter:

```
44.  FLAG1 = FLAG1 + 1
```

Test for maximum number of iterations:

```
45.  If FLAG1 $\le$ MXIT2,  go to 59

     If FLAG1 $>$ MXIT2,  go to 46
```

If maximum number of iterations has been exceeded, print error message and restore initial values of independent variables:

```
46.  If ITAPE = 0,  go to 49

     If ITAPE $\neq$ 0,  go to 47

47.  Write error message on output tape

48.  If TAPE $>$ 0,  go to 50

     If TAPE $\le$ 0,  go to 49

49.  Write error message on the on-line printer

50.  X1ADD = X1SAVE

51.  X2ADD = X2SAVE

52.  ERROR = HSRCH2

53.  Return
```

Singular matrix error message:

```
54.  If ITAPE $= 0$, go to 57
     If ITAPE $\neq 0$, go to 55
55.  Write error message on the output tape
56.  If TAPE $> 0$, go to 50
     If TAPE $\leq 0$, go to 57
57.  Write error message on the on-line printer
58.  Go to 50
```

Evaluate dependent variables:

```
59.  CALL FUNCG
```

Test if error flag was set during evaluation of dependent variables:

```
60.  If ERROR $\neq 0$, go to 148
     If ERROR $= 0$, go to 61
61.  $F1(J) = U1ADD$
62.  $F2(J) = U2ADD$
```

Write output:

```
63.  $N = 130$
64.  Go to 177
```

Test value of first dependent variable for convergence:

```
65.  If U1STAR ≠ 0,  go to 68
     If U1STAR = 0,  go to 66
66.  T = DELTU1 - |F1(J)|
67.  Go to 69
68.  T = |(U1STAR)(DELTU1)| - |U1STAR - F1(J)|
69.  If T < 0,  go to 76
     If T ≥ 0,  go to 70
```

Test value of second dependent variable for convergence:

```
70.  If U2STAR ≠ 0,  go to 73
     If U2STAR = 0,  go to 71
71.  T = DELTU2 - |F2(J)|
72.  Go to 74
73.  T = |(U2STAR)(DELTU2)| - |U2STAR - F2(J)|
74.  If T < 0,  go to 76
     If T ≥ 0,  go to 75
75.  Return
```

Re-evaluate independent variables:

```
76.  J = J + 1
```

```
77.  I = I + 1
78.  If I ≤ 2, go to 42
     If I > 2, go to 79
79.  If JFLAG < 0, go to 82
     If JFLAG = 0, go to 137
     If JFLAG > 0, go to 80
80.  JFLAG = JFLAG - 1
81.  Go to 85
82.  JFLAG = JFLAG + 1
83.  If JFLAG = 0, JFLAG = 1
     If JFLAG ≠ 0, go to 84
84.  Go to 93
85.  If V1B(2) = 0, go to 54
     If V1B(2) ≠ 0, go to 86
86.  MATRIX(1) = (F1(2) - F1(1))/(V1B(2))
87.  If V2B(3) = 0, go to 54
     If V2B(3) ≠ 0, go to 88
88.  MATRIX(2) = (F1(3) - F1(1))/(V2B(3))
89.  MATRIX(3) = (F2(2) - F2(1))/(V1B(2))
90.  MATRIX(4) = (F2(3) - F2(1))/(V2B(3))
```

```
91.  DETER = (MATRIX(1))(MATRIX(4)) -
             (MATRIX(2))(MATRIX(3))
92.  If DETER = 0, go to 54
     If DETER ≠ 0, go to 93
93.  MATRIX(5) = U1STAR - F1(1)
94.  MATRIX(6) = U2STAR - F2(1)
95.  If DETER = 0, go to 54
     If DETER ≠ 0, go to 96
96.  DX1 = ((MATRIX(4))(MATRIX(5)) -
          (MATRIX(2))(MATRIX(6)))/(DETER)
97.  DX2 = ((MATRIX(1))(MATRIX(6)) -
          (MATRIX(3))(MATRIX(5)))/(DETER)
98.  If KDAMP2 = 1, go to 105
     If KDAMP2 ≠ 1, go to 99
99.  If NDAMP2 = 0, go to 105
     If NDAMP2 ≠ 0, go to 100
100. If NDAMP2 = DUMMY, go to 105
     If NDAMP2 ≠ DUMMY, go to 101
101. DAMP = KDAMP2 + (1.0 - KDAMP2)(DUMMY)/(NDAMP2)
102. DX1 = (DX1)(DAMP)
103. DX2 = (DX2)(DAMP)
```

104. DUMMY = DUMMY + 1.0

105. V1A(1) = V1(1)

106. V2A(1) = V2(1)

107. FLAGS(1) = 0

108. If JFLAG ≠ 0, go to 120

If JFLAG = 0, go to 109

109. V1(3) = V1(2)

110. V1(2) = V1(1)

111. V2(3) = V2(2)

112. V2(2) = V2(1)

113. F1(3) = F1(2)

114. F1(2) = F1(1)

115. F2(3) = F2(2)

116. F2(2) = F2(1)

117. V1(1) = V1(1) + DX1

118. V2(1) = V2(1) + DX2

119. Go to 39

120. V1(1) = V1(1) + DX1

121. V1A(2) = V1(1)

122. V1A(3) = V1(1)

123. V1(3) = V1(1)

```
124.  V2(1) = V2(1) + DX2
125.  V2(2) = V2(1)
126.  V2A(2) = V2(1)
127.  V2A(3) = V2(1)
128.  V1B(2) = (V1B(2))(SCALE(2))
129.  V1(2) = V1(1) + V1B(2)
130.  V2B(3) = (V2B(3))(SCALE(3))
131.  V2(3) = V2(1) + V2B(3)
132.  V1B(3) = 0
133.  V2B(2) = 0
134.  FLAGS(2) = 0
135.  FLAGS(3) = 0
136.  Go to 39
137.  Z(1) = V1(1) - V1(2)
138.  Z(2) = V1(2) - V1(3)
139.  Z(3) = V2(1) - V2(2)
140.  Z(4) = V2(2) - V2(3)
141.  ZDETER = (Z(1))(Z(4)) - (Z(2))(Z(3))
142.  If ZDETER = 0, go to 50
      If ZDETER ≠ 0, go to 143
```

```
143.  MATRIX(1) = ((Z(4))(F1(1) - F1(2)) - (Z(3))
                  (F1(2) - F1(3)))/(MATRIX(1))

144.  MATRIX(2) = ((Z(1))(F1(2) - F1(3)) - (Z(2))
                  (F1(1) - F1(2)))/(MATRIX(2))

145.  MATRIX(3) = ((Z(4))(F2(1) - F2(2)) - (Z(3))
                  (F2(2) - F2(3)))/(MATRIX(3))

146.  MATRIX(4) = ((Z(1))(F2(2) - F2(3)) - (Z(2))
                  (F2(1) - F2(2)))/(MATRIX(4))

147.  Go to 91
```

Set error indicator to zero and write error message:

```
148.  ERROR = 0

149.  If ITAPE = 0, go to 152

      If ITAPE ≠ 0, go to 150

150.  Write error message on output tape

151.  If TAPE > 0, go to 153

      If TAPE ≤ 0, go to 152

152.  Write error message on the on-line printer

153.  N = 815

154.  Go to 177

155.  FLAGS(J) = FLAGS(J) + 1
```

```
156. If FLAGS(J) ≥ ESTOP(J), go to 50
     If FLAGS(J) < ESTOP(J), go to 157
157. V1B(J) = (V1(J) - V1A(J))(SCALE(J))
158. V2B(J) = (V2(J) - V2A(J))(SCALE(J))
159. V1(J) = V1B(J) + V1A(J)
160. V2(J) = V2B(J) + V2A(J)
161. If J > 1, go to 42
     If J ≤ 1, go to 162
162. V1A(2) = V1(1)
163. V1A(3) = V1(1)
164. V1(3) = V1(1)
165. If V1A(1) ≠ 0, go to 168
     If V1A(1) = 0, go to 166
166. V1(2) = V1(1) + (V1(1))(DELTX1)/(100)
167. Go to 169
168. V1(2) = V1(1) + V1B(2)
169. V2A(2) = V2(1)
170. V2A(3) = V2(1)
171. V2(2) = V2(1)
172. If V2A(1) ≠ 0, go to 175
     If V2A(1) = 0, go to 173
```

```
173.  V2(3) = V2(1) + (V2(1))(DELTX2)/(100)

174.  Go to 176

175.  V2(3) = V2(1) + V2B(3)

176.  Go to 42
```

Output:

```
177.  If ITAPE = 0, go to 180

      If ITAPE ≠ 0, go to 178

178.  Write output on output tape

179.  If TAPE > 0, go to 181

      If TAPE ≤ 0, go to 180

180.  Write output on the on-line printer

181.  If N = 130, go to 65

      If N = 815, go to 155
```

SUBROUTINE SRCH2 NOMENCLATURE

Symbol	Description	Units	Reference
DAMP	Calculated damping factor	—	101, 102, 103
DELTU1	Convergence tolerance for U1ADD	—	CALL, 66, 68
DELTU2	Convergence tolerance for U2ADD	—	CALL, 71, 73
DELTX1	Per cent increment for X1ADD	—	CALL, 26, 166
DELTX2	Per cent increment for X2ADD	—	CALL, 27, 173
DETER	Determinant used in solving the matrix	—	92, 95, 96, 97
DUMMY	Number of times damped	—	36, 100, 101, 104
DX1	Predicted change for first independent variable	—	96, 102, 117, 120
DX2	Predicted change for second independent variable	—	97, 103, 118, 124
ERROR	Error communication flag	—	CALL, 1, 52, 60, 148
ES0	Error stop for point zero	—	CALL, INTEGER, 28
ES1	Error stop for point 1	—	CALL, INTEGER, 29
ES2	Error stop for point 2	—	CALL, INTEGER, 30
ESTOP	Saves the counters for error conditions	—	DIM, INTEGER, 28, 29, 30, 156

SUBROUTINE SRCH2 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
F1	Value of the first dependent variable	—	DIM, 61, 66, 68, 86, 88, 113, 114, 143, 144
F2	Value of the second dependent variable	—	DIM, 62, 71, 73, 89, 90, 115, 116, 145, 146
FLAG1	Iteration counter	—	INTEGER, 37, 44, 45
FLAGS	Error condition counters	—	DIM, INTEGER, 6, 7, 8, 107, 134, 135, 155, 156
FUNCG	Name of subroutine which evaluates dependent variable	—	CALL, 59
HSRCH2	—	—	DATA, 52
I	Variable index	—	39, 77, 78
ITAPE	Absolute value of TAPE	—	9, 10, 46, 54, 149, 177
J	Variable index	—	41, 76, 161
JACOB	Jacobian counter	—	CALL, 38
JFLAG	Jacobian counter	—	38, 40, 44, 45, 79, 80, 82, 83, 108
K	Scale factor for point zero	—	CALL, REAL, 31
KDAMP2	Damping factor	—	CALL, REAL, 98, 101
L	Scale factor for points one and two	—	CALL, REAL, 32, 33
MATRIX	Calculated values of the partial matrix	—	DIM, REAL, 86, 88, 89, 90, 91, 93, 94, 96, 97, 143, 144, 145, 146

SUBROUTINE SRCH2 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
MXIT2	Maximum allowable number of iterations	—	CALL, 45
N	—	—	63, 153, 181
NDAMP2	Damping factor	—	CALL, REAL, 99, 100, 101
SCALE	Scale factors	—	DIM, 31, 32, 33, 128, 130, 157, 158
T	Temporary storage	—	66, 68, 69, 71, 73, 74
TAPE	Tape unit to write search data on	—	INTEGER, CALL, 9, 12, 48, 56, 151, 179
U1ADD	Location of first dependent variable	—	CALL, 61
U2ADD	Location of second dependent variable	—	CALL, 62
U1BCI	Name of first dependent variable	—	CALL, DIM, 11, 13
U2BCI	Name of second dependent variable	—	CALL, DIM, 11, 13
U1STAR	Desired value of first dependent variable	—	CALL, 65, 68, 93, 178, 180
U2STAR	Desired value of second dependent variable	—	CALL, 70, 73, 94, 178, 180
V1	Current value of the first independent variable	—	DIM, 18, 19, 26, 34, 42, 105, 109, 110, 117, 120, 121, 122, 123, 129, 137, 138, 157, 159, 162, 163, 164, 166, 168

SUBROUTINE SRCH2 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
V2	Current value of the second independent variable	—	DIM, 24, 25, 27, 35, 43, 106, 111, 112, 118, 124, 125, 126, 127, 131, 139, 140, 158, 160, 169, 170, 171, 173, 175
V1A	Storage for first independent variable	—	DIM, 2, 15, 16, 105, 121, 122, 157, 159, 162, 163, 165
V2A	Storage for second independent variable	—	DIM, 3, 21, 22, 106, 126, 127, 158, 160, 169, 170, 172
V1B	Storage for first independent variable	—	DIM, 4, 17, 26, 34, 85, 86, 89, 128, 129, 132, 157, 159, 168
V2B	Storage for second independent variable	—	DIM, 5, 23, 27, 35, 87, 88, 90, 130, 131, 133, 158, 160, 175
X1ADD	Location of the first independent variable	—	CALL, 14, 42, 50
X2ADD	Location of the second independent variable	—	CALL, 20, 43, 51
X1BCI	Name of the first independent variable	—	CALL, DIM, 11, 13
X2BCI	Name of the second independent variable	—	CALL, DIM, 11, 13

SUBROUTINE SRCH2 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
X1SAVE	Initial value of the first independent variable	—	14, 15, 16, 17, 18, 19, 50
X2SAVE	Initial value of the second independent variable	—	20, 21, 22, 23, 24, 25, 51
Z	—	—	DIM, 137, 138, 139, 140, 141, 143, 144, 145, 146
ZDETER	—	—	141, 142

```
$IBFTC SRCH2   FULIST,REF
      SUBRØUTINE SRCH2(FUNCG,ERRØR,X1BCI,X1ADD,X2BCI,X2ADD,U1BCI,U1ADD,
     $        U2BCI,U2ADD,U1STAR,U2STAR,DELTX1,DELTX2,DELTU1,DELTU2,
     $        MXIT2,JACØB,K,L,ESO,ES1,ES2,KDAMP2,NDAMP2,TAPE)
C        FUNCG   - RØUTINE TØ EVALUATE FUNCTIØN
C        ERRØR   - ERRØR CØMMUNICATIØN FLAG
C        X1BCI   - X1 NAME
C        X1ADD   - X1 LØCATIØN
C        X2BCI   - X2 NAME
C        X2ADD   - X2 LØCATIØN
C        U1BCI   - U1 NAME
C        U1ADD   - U1 LØCATIØN
C        U2BCI   - U2 NAME
C        U2ADD   - U2 LØCATIØN
C        U1STAR  - DESIRED VALUE ØF U1
C        U2STAR  - DESIRED VALUE ØF U2
C        DELTX1  - PER CENT INCREMENT FØR X1
C        DELTX2  - PER CENT INCREMENT FØR X2
C        DELTU1  - CØNVERGENCE TØLERANCE FØR U1
C        DELTU2  - CØNVERGENCE TØLERANCE FØR U2
C        MXIT2   - MAX ITERATIØNS
C        JACØB   - JACØBIAN CØUNTER
C        K       - SCALE FACTØR FØR PØINT 0
C        L       - SCALE FACTØR FØR PØINTS 1,2
C        ESØ     - ERRØR STØP FØR PØINT 0
C        ES1     - ERRØR STØP FØR PØINT 1
C        ES2     - ERRØR STØP FØR PØINT 2
C        KDAMP2  - DAMPING FACTØR
C        NDAMP2  - DAMPING FACTØR
C        TAPE    - TAPE TØ WRITE SEARCH DATA ØN
      REAL  K,L,KDAMP2,NDAMP2
      INTEGER  ESO,ES1,ES2
      DIMENSIØN  X1BCI(2),X2BCI(2),U1BCI(2),U2BCI(2)
C
      DIMENSIØN  V1A(3),V2A(3),V1B(3),V2B(3),V1(3),V2(3),F1(3),F2(3),
     $           FLAGS(3),ESTØP(3),SCALE(3),MATRIX(6) ,Z(4)
      REAL MATRIX
      INTEGER    FLAG1,FLAGS,ESTØP ,TAPE
      DATA  HSRCH2/5HSRCH2/
      ERRØR = 0.0
      V1A(1) = 0.0
      V2A(1) = 0.0
      V1B(3) = 0.0
      V2B(2) = 0.0
      FLAGS(1) = 0.0
      FLAGS(2) = 0.0
      FLAGS(3) = 0.0
      ITAPE = IABS(TAPE)
      IF (ITAPE .EQ. 0)  GØ TØ 6
      WRITE (ITAPE,5)  U1BCI,U2BCI,X1BCI,X2BCI
```

FIGURE 13. SYMBOLIC LISTING OF SUBROUTINE SRCH2

```
      IF (TAPE .GT. 0)  GØ TØ 10
    6 PRINT 5,  U1BCI,U2BCI,X1BCI,X2BCI
    5 FØRMAT(1H0,31H       TWØ-PARAMETER SEARCH DATA/
     $       1H ,2(10X,A6,1H(,I4,1H),10X),2(2X,A6,1H(,I4,1H),2X))
   10 X1SAVE = X1ADD
      V1A(2) = X1SAVE
      V1A(3) = X1SAVE
      V1B(1) = X1SAVE
      V1(1) = X1SAVE
      V1(3) = X1SAVE
      X2SAVE = X2ADD
      V2A(2) = X2SAVE
      V2A(3) = X2SAVE
      V2B(1) = X2SAVE
      V2(1) = X2SAVE
      V2(2) = X2SAVE
      V1B(2) = DELTX1 * V1(1)/100.0
      V2B(3) = DELTX2 * V2(1)/100.0
      ESTØP(1) = ES0
      ESTØP(2) = ES1
      ESTØP(3) = ES2
      SCALE(1) = K
      SCALE(2) = L
      SCALE(3) = L
      V1(2) = V1(1) + V1B(2)
      V2(3) = V2(1) + V2B(3)
      DUMMY = 0.0
      FLAG1 = 0
      JFLAG = JACØB
C        ITERATIØN LØØP
  100 I = 0
      IF (JFLAG .LT. 0)  I = 2
      J = 1
  110 X1ADD = V1(J)
      X2ADD = V2(J)
C        CHECK MAX ITERATIØNS
      FLAG1 = FLAG1 + 1
      IF (FLAG1 .LE. MXIT2)  GØ TØ 120
  830 IF (ITAPE .EQ. 0)  GØ TØ 835
      WRITE (ITAPE,2)
      IF (TAPE .GT. 0)  GØ TØ 820
  835 PRINT 2
    2 FØRMAT(44H       MAX ITERATIØNS -- TWØ PARAMETER SEARCH)
C          ERRØR RETURN
  820 X1ADD = X1SAVE
      X2ADD = X2SAVE
      ERRØR = HSRCH2
      RETURN
C          SINGULAR MATRIX
  840 IF (ITAPE .EQ. 0)  GØ TØ 845
```

FIGURE 13. SYMBOLIC LISTING OF SUBROUTINE SRCH2 (cont.)

```
      WRITE (ITAPE,3)
      IF (TAPE .GT. 0)  GØ TØ 820
  845 PRINT 3
    3 FØRMAT(20H      0 PERTURBATIØN)
      GØ TØ 820
C
  120 CALL FUNCG
      IF (ERRØR .NE. 0.0)  GØ TØ 800
      F1(J) = U1ADD
      F2(J) = U2ADD
C           PRINT
      ASSIGN 130 TØ N
      GØ TØ 900
C           TEST CØNVERGENCE
  130 IF (U1STAR .NE. 0.0)  GØ TØ 140
      T = DELTU1 - ABS(F1(J))
      GØ TØ 150
  140 T = ABS(U1STAR * DELTU1) - ABS(U1STAR - F1(J))
  150 IF (T .LT. 0.0)  GØ TØ 200
C           CØNVERGED ØN U1, CHECK U2
      IF (U2STAR .NE. 0.0)  GØ TØ 160
      T = DELTU2 - ABS(F2(J))
      GØ TØ 170
  160 T = ABS(U2STAR * DELTU2) - ABS(U2STAR - F2(J))
  170 IF (T .LT. 0.0)  GØ TØ 200
C     CØNVERGED RETURN
      RETURN
C
  200 J = J + 1
      I = I + 1
      IF (I .LE. 2)  GØ TØ 110
      IF (JFLAG)  220,400,210
  210 JFLAG = JFLAG - 1
      GØ TØ 230
  220 JFLAG = JFLAG + 1
      IF (JFLAG .EQ. 0)  JFLAG = 1
      GØ TØ 250
C           JACØBIAN CØUNTER WAS GREATER THAN ZERØ AND I GREATER THAN 2
C                   CALCULATE PARTIAL MATRIX
  230 IF (V1B(2) .EQ. 0.0)  GØ TØ 840
      MATRIX(1) = (F1(2) - F1(1))/V1B(2)
      IF (V2B(3) .EQ. 0.0)  GØ TØ 840
      MATRIX(2) = (F1(3) - F1(1))/V2B(3)
      MATRIX(3) = (F2(2) - F2(1))/V1B(2)
      MATRIX(4) = (F2(3) - F2(1))/V2B(3)
C           CALCULATE DETERMINANT
  240 DETER = MATRIX(1) * MATRIX(4) - MATRIX(2) * MATRIX(3)
      IF (DETER .EQ. 0.0)  GØ TØ 840
C           SØLVE PARTIAL MATRIX
  250 MATRIX(5) = U1STAR - F1(1)
```

FIGURE 13. SYMBOLIC LISTING OF SUBROUTINE SRCH2 (cont.)

```
      MATRIX(6) = U2STAR - F2(1)
      IF (DETER .EQ. 0.0)  GO TO 840
      DX1 = (MATRIX(4) * MATRIX(5) - MATRIX(2) * MATRIX(6))/DETER
      DX2 = (MATRIX(1) * MATRIX(6) - MATRIX(3) * MATRIX(5))/DETER
C            DAMP DELTAS
      IF (KDAMP2 .EQ. 1.0)  GO TO 260
      IF (NDAMP2 .EQ. 0.0)  GO TO 260
      IF (NDAMP2 .EQ. DUMMY)  GO TO 260
      DAMP = KDAMP2 + (1.0 - KDAMP2) * DUMMY/NDAMP2
      DX1 = DX1 * DAMP
      DX2 = DX2 * DAMP
      DUMMY = DUMMY + 1.0
C         RESET INITIAL CONDITIONS
  260 V1A(1) = V1(1)
      V2A(1) = V2(1)
      FLAGS(1) = 0
      IF (JFLAG .NE. 0)  GO TO 300
C            SHIFT POINTS IF JACOBIAN COUNTER ZERO
      V1(3) = V1(2)
      V1(2) = V1(1)
      V2(3) = V2(2)
      V2(2) = V2(1)
      F1(3) = F1(2)
      F1(2) = F1(1)
      F2(3) = F2(2)
      F2(2) = F2(1)
      V1(1) = V1(1) + DX1
      V2(1) = V2(1) + DX2
      GO TO 100
C         JACOBIAN COUNTER NOT ZERO
  300 V1(1) = V1(1) + DX1
      V1A(2) = V1(1)
      V1A(3) = V1(1)
      V1(3) = V1(1)
      V2(1) = V2(1) + DX2
      V2(2) = V2(1)
      V2A(2) = V2(1)
      V2A(3) = V2(1)
      V1B(2) = V1B(2) * SCALE(2)
      V1(2) = V1(1) + V1B(2)
      V2B(3) = V2B(3) * SCALE(3)
      V2(3) = V2(1) + V2B(3)
      V1B(3) = 0.0
      V2B(2) = 0.0
      FLAGS(2) = 0
      FLAGS(3) = 0
      GO TO 100
C         EVALUATE PARTIALS IF JACOBIAN ZERO
  400 Z(1) = V1(1) - V1(2)
      Z(2) = V1(2) - V1(3)
```

FIGURE 13. SYMBOLIC LISTING OF SUBROUTINE SRCH2 (cont.)

```
      Z(3) = V2(1) - V2(2)
      Z(4) = V2(2) - V2(3)
      ZDETER = Z(1) * Z(4) - Z(2) * Z(3)
      IF (ZDETER .EQ. 0.0)  GO TO 820
      MATRIX(1) = (Z(4) * (F1(1) - F1(2)) - Z(3) * (F1(2) - F1(3)))/
     $            MATRIX(1)
      MATRIX(2) = (Z(1) * (F1(2) - F1(3)) - Z(2) * (F1(1) - F1(2)))/
     $            MATRIX(2)
      MATRIX(3) = (Z(4) * (F2(1) - F2(2)) - Z(3) * (F2(2) - F2(3)))/
     $            MATRIX(3)
      MATRIX(4) = (Z(1) * (F2(2) - F2(3)) - Z(2) * (F2(1) - F2(2)))/
     $            MATRIX(4)
      GO TO 240
C         ERROR CONTROL
  800 ERROR = 0.0
      IF (ITAPE .EQ. 0)  GO TO 805
      WRITE (ITAPE,1)
      IF (TAPE .GT. 0)  GO TO 810
  805 PRINT 1
    1 FORMAT(30H       FUNCTION GENERATOR ERROR )
  810 ASSIGN 815 TO N
      GO TO 900
  815 FLAGS(J) = FLAGS(J) + 1
      IF (FLAGS(J) .GE. ESTOP(J))  GO TO 820
C         MODIFY INCREMENT
  830 V1B(J) = (V1(J) - V1A(J)) * SCALE(J)
      V2B(J) = (V2(J) - V2A(J)) * SCALE(J)
      V1(J) = V1B(J) + V1A(J)
      V2(J) = V2B(J) + V2A(J)
      IF (J .GT. 1)  GO TO 110
C
      V1A(2) = V1(1)
      V1A(3) = V1(1)
      V1(3) = V1(1)
      IF (V1A(1) .NE. 0.0)  GO TO 860
      V1(2) = V1(1) + V1(1) * DELTX1/100.0
      GO TO 870
  860 V1(2) = V1(1) + V1B(2)
  870 V2A(2) = V2(1)
      V2A(3) = V2(1)
      V2(2) = V2(1)
      IF (V2A(1) .NE. 0.0)  GO TO 880
      V2(3) = V2(1) + V2(1) * DELTX2/100.0
      GO TO 890
  880 V2(3) = V2(1) + V2B(3)
  890 GO TO 110
C         PRINT AREA
  900 IF (ITAPE .EQ. 0)  GO TO 905
      WRITE (ITAPE,4)  U1STAR,F1(J),U2STAR,F2(J),V1(J),V2(J)
      IF (TAPE .GT. 0)  GO TO 910
```

FIGURE 13. SYMBOLIC LISTING OF SUBROUTINE SRCH2 (cont.)

```
  905 PRINT 4,  U1STAR,F1(J),U2STAR,F2(J),V1(J),V2(J)
    4 FØRMAT(1H ,6E16.8)
  910 GØ TØ N, (130,815)
C
C
      END
```

FIGURE 13. SYMBOLIC LISTING OF SUBROUTINE SRCH2 (cont.)

IX. SUBROUTINE MNX

This subroutine is used to find the value of an independent variable which minimizes or maximizes the value of a dependent variable.

CALLING SEQUENCE

The subroutine calling sequence is:

```
CALL MNX (FUNCG, ERROR, XBCI, XADD, UBCI, UADD,
          OPTION, DELTAX, DELTAU, MAXIT, MAXER,
          SCALE, TAPE)
```

SOLUTION METHOD

Initialize variables:

1. ERROR = 0
2. V0 = 0
3. V1 = 0
4. VSAVE = XADD
5. FLAG1 = 0

```
6.  ITAPE = |TAPE|
7.  If ITAPE = 0, go to 10
    If ITAPE ≠ 0, go to 8
8.  Write headings on output scratch tape
9.  If TAPE > 0, go to 11
    If TAPE ≤ 0, go to 10
10. Write headings on on-line printer
11. Continue

Determine type of case; maximization or minimization:

12. OPT = OPT1
13. If OPTION > 0, OPT = OPT2
    If OPTION ≤ 0, OPT = OPT1

Test initial value of independent variable. It cannot be zero:

14. DV = VSAVE
15. If DV = 0, go to 45
    If DV ≠ 0, go to 16

Set error indicator to zero:

16. FLAG2 = 0
```

Increment independent variable and evaluate dependent variable:

```
17.  V2 = V1 + DV

18.  XADD = V2

19.  Call FUNCG
```

Test if error flag was set during evaluation of dependent variable:

```
20.  If ERROR ≠ 0, go to 67

     If ERROR = 0, go to 21
```

Write output:

```
21.  F2 = UADD

22.  N = 90

23.  Go to 79
```

Increment iteration control counter:

```
24.  FLAG1 = FLAG1 + 1
```

Determine number of times subroutine has been entered:

```
25.  If FLAG1 < 2, go to 26

     If FLAG1 = 2, go to 33

     If FLAG1 > 2, go to 38
```

First Pass

```
26. If DELTAX = 0, go to 28
    If DELTAX ≠ 0, go to 27
27. DV = (DV)(DELTAX)/(100)
28. V0 = V1
29. V1 = V2
30. F0 = F1
31. F1 = F2
32. Go to 16
```

Second Pass

```
33. If (F2 - F1)(OPTION) ≥ 0, go to 28
    If (F2 - F1)(OPTION) < 0, go to 34
34. DV = (-DV)/(2)
35. V0 = V2
36. F0 = F2
37. Go to 16
```

Third Pass or Greater

```
38. DF = F2 - F1
    Test for convergence:
```

39. If $|(\text{DELTAU})(\text{F2})| \geq |\text{DF}|$, return

If $|(\text{DELTAU})(\text{F2})| < |\text{DF}|$, go to 40

Test for maximum number of iterations:

40. If $\text{MAXIT} \geq \text{FLAG1}$, go to 47

If $\text{MAXIT} < \text{FLAG1}$, go to 41

Error Return

If maximum number of iterations has been exceeded, print error message and restore initial value of indpendent variable.

41. If $\text{ITAPE} = 0$, go to 44

If $\text{ITAPE} \neq 0$, go to 42

42. Write error message on output scratch tape

43. If $\text{TAPE} > 0$, go to 45

If $\text{TAPE} \leq 0$, go to 44

44. Write error message on on-line printer

45. $\text{ERROR} = \text{HMNX}$

46. $\text{XADD} = \text{VSAVE}$

Return

Prediction Loop

Compute coefficient of quadratic square term, A, and predict new value for independent variable.

```
47.  DV0 = V1 - V0

48.  SDV = DV + DV0

49.  DF0 = F1 - F0

50.  SLOPE1 = (DF0)/(DV0)

51.  SLOPE2 = (DF)/(DV)

52.  A = (SLOPE2 - SLOPE1)/(SDV)

53.  If (A)(OPTION) ≥ 0,  go to 60

     If (A)(OPTION) < 0 ,  go to 54

54.  VPK = (V1 + V2 - (SLOPE2)/(A))/(2)

55.  If (DF)(OPTION) < 0, go to 58

     If (DF)(OPTION) ≥ 0, go to 56
```

Value of independent variable which yields a maximum or minimum is greater than current value of independent variable.

```
56.  DV = VPK - V2

57.  Go to 28
```

Value of independent variable which yields a maximum or minimum lies between last two values of the independent variable.

```
58.  DV = VPK - V1

59.  Go to 35

60.  If (F2 - F0)(OPTION) < 0, go to 63

     If (F2 - F0)(OPTION) ≥ 0 , go to 61
```

Value of independent variable which yields a minimum or maximum is greater than current value of independent variable.

61. DV = (2)(DV)

62. Go to 28

Value of independent variable which yields a minimum or maximum is less than next to last value of independent variable.

63. F2 = F0

64. V2 = V0

65. DV = (V2 - V1)/(2)

66. Go to 28

Error Loop

67. If ITAPE = 0, go to 70

If ITAPE ≠ 0, go to 68

68. Write error message on output scratch tape

69. If TAPE > 0, go to 71

If TAPE ≤ 0, go to 70

70. Write error message on the on-line printer

71. ERROR = 0

72. N = 810

73. Go to 79

74. FLAG2 = FLAG2 + 1

75. If FLAG2 $\geq$ MAXER, go to 45

If FLAG2 $<$ MAXER, go to 76

76. If DV $= 0$, go to 45

If DV $\neq 0$, go to 77

Scale change in independent variable and re-evaluate dependent variable:

77. DV = (DV)(SCALE)

78. Go to 17

Output

79. If ITAPE $= 0$, go to 82

If ITAPE $\neq 0$, go to 80

80. Write output on output scratch tape

81. If TAPE > 0, go to 83

If TAPE ≤ 0, go to 82

82. Write output on the on-line printer

83. If N $= 90$, go to 24

If N $= 810$, go to 74

SUBROUTINE MNX NOMENCLATURE

Symbol	Description	Units	Reference
A	Coefficient of quadratic square term	—	52, 53, 54
DELTAU	Convergence tolerance	—	CALL, 39
DELTAX	Per cent increment	—	CALL, 26, 27
DF	F2 - F1	—	38, 39, 51, 55
DF0	F1 - F0	—	49, 50
DV	Change in independent variable	—	14, 15, 17, 27, 34, 48, 51, 56, 58, 61, 65, 76, 77
DV0	V1 - V0	—	47, 48, 50
ERROR	Error communication flag	—	CALL, 1, 20, 45, 71
F0	Next to last value of dependent variable	—	30, 36, 49, 60, 63
F1	Last value of dependent variable	—	30, 31, 33, 38, 49
F2	Current value of dependent variable	—	21, 31, 33, 36, 38, 39, 60, 63
FLAG1	Indicator which denotes number of times subroutine has been entered	—	INTEGER, 5, 24, 25, 40
FLAG2	Error counter	—	INTEGER, 16, 74, 75
FUNCG	Name of subroutine which evaluates dependent variable	—	CALL, 19

SUBROUTINE MNX NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
HMNX	—	—	DATA, 45
ITAPE	—	—	6, 7, 41, 67, 79
MAXER	Maximum number of allowable consecutive errors	—	CALL, 75
MAXIT	Maximum allowable number of iterations	—	CALL, 40
N	—	—	22, 72, 83
OPT	—	—	12, 13
OPT1	—	—	DATA, 12, 13
OPT2	—	—	DATA, 13
OPTION	Option indicator +1 = maximize -1 = minimize	—	CALL, 13, 33, 53, 55, 60
SCALE	Scale factor for independent variable in case of error	—	CALL, 77
SDV	DV + DV0	—	48, 52
SLOPE1	Slope of function between points 1 and 2	—	50, 52
SLOPE2	Slope of function between points 2 and 3	—	51, 52, 54
TAPE	Tape unit to write output data on	—	CALL, INTEGER, 6, 9, 43, 69, 81

SUBROUTINE MNX NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
UADD	Name of dependent variable	—	CALL, 21
UBCI	Hollerith name of dependent variable	—	CALL, DIM
V0	Next to last value of independent variable	—	2, 28, 35, 47, 64
V1	Last value of independent variable	—	3, 17, 28, 29, 47, 54, 58, 65
V2	Current value of independent variable	—	17, 18, 29, 35, 54, 56, 64, 65
VPK	—	—	54, 56, 58
VSAVE	Initial value of independent variable	—	4, 14, 46
XADD	Name of independent variable	—	CALL, 4, 18, 46
XBCI	Hollerith name of independent variable	—	CALL, DIM

```
$IBFTC MNX     FULIST,REF
      SUBRØUTINE MNX(FUNCG,ERRØR,XBCI,XADD,UBCI,UADD,ØPTIØN,DELTAX,
     $             DELTAU,MAXIT,MAXER,SCALE,TAPE)
C              MIN-MAX SEARCH RØUTINE
C         FUNCG  - RØUTINE TØ EVALUATE FUNCTIØN
C         ERRØR  - ERRØR CØMMUNICATIØN FLAG
C         XBCI   - NAME FØR X, 2 WØRDS - HØLLERITH, INTEGER
C         XADD   - LØCATIØN ØF VARIABLE
C         UBCI   - NAME FØR U, 2 WØRDS - HØLLERITH, INTEGER
C         UADD   - LØCATIØN ØF FUNCTIØN
C         ØPTIØN - ØPTIØN INDICATØR, +1.0 - MAXIMIZE  -1.0 - MINIMIZE
C         DELTAX - PER CENT INCREMENT  (0 = 100)
C         DELTAU - CØNVERGENCE TØLERANCE
C         MXIT   - MAX ITERATIØNS
C         MAXER  - MAX NØ ØF CØNSECUTIVE ERRØRS ALLØWABLE
C         SCALE  - SCALE FACTØR FØR DELTA IN CASE ØF ERRØR
C         TAPE   - TAPE UNIT TØ WRITE SEARCH DATA ØN
      INTEGER  TAPE
      INTEGER  FLAG1,FLAG2
      DATA  ØPT1,ØPT2/6H  MIN ,6H  MAX /
      DATA  HMNX/3HMNX/
      DIMENSIØN  XBCI(2),UBCI(2)
C
      ERRØR = 0.0
      V0 = 0.0
      V1 = 0.0
      VSAVE = XADD
      FLAG1 = 0
      ITAPE = IABS(TAPE)
      IF (ITAPE .EQ. 0)  GØ TØ 4
      WRITE (ITAPE,5)  XBCI,UBCI
      IF (TAPE .GT. 0)  GØ TØ 6
    4 PRINT 5,  XBCI,UBCI
    5 FØRMAT(1H0,25H      MIN-MAX SEARCH DATA/
     $      1H ,6X,6HØPTIØN,6X,2(2X,A6,1H(,I4,1H),2X))
    6 CØNTINUE
      ØPT = ØPT1
      IF (ØPTIØN .GT. 0.0)  ØPT = ØPT2
      DV = VSAVE
      IF (DV .EQ. 0.0)  GØ TØ 310
C
   10 FLAG2 = 0
   11 V2 = V1 + DV
      XADD = V2
      CALL FUNCG
      IF (ERRØR .NE. 0.0)  GØ TØ 800
      F2 = UADD
C         PRINT
      ASSIGN 90 TØ N
      GØ TØ 900
```

FIGURE 14. SYMBOLIC LISTING OF SUBROUTINE MNX

```
   90 FLAG1 = FLAG1 + 1
      IF (FLAG1 - 2) 100,200,300
C         FIRST PASS ØNLY
  100 IF (DELTAX .EQ. 0.0)  GØ TØ 110
      DV = DV * DELTAX/100.0
  110 V0 = V1
      V1 = V2
      F0 = F1
      F1 = F2
      GØ TØ 10
C         SECØND PASS
  200 IF (((F2-F1) * ØPTIØN) .GE. 0.0)  GØ TØ 110
      DV = -DV/2.0
  210 V0 = V2
      F0 = F2
      GØ TØ 10
C         THIRD PASS
  300 DF = F2 - F1
      IF ((ABS(DELTAU * F2) - ABS(DF)) .GE. 0.0)  RETURN
C         CHECK FØR MAX ITER
      IF(MAXIT .GE. FLAG1)  GØ TØ 400
C         ERRØR RETURN
      IF (ITAPE .EQ. 0)  GØ TØ 305
      WRITE (ITAPE,3)
      IF (TAPE .GT. 0)  GØ TØ 310
  305 PRINT 3
    3 FØRMAT(38H        MAX ITERATIØNS -- MIN-MAX SEARCH)
  310 ERRØR = HMNX
      XADD = VSAVE
      RETURN
C         PREDICTIØN LØØP
  400 DV0 = V1 - V0
      SDV = DV + DV0
      DF0 = F1 - F0
      SLØPE1 = DF0/DV0
      SLØPE2 = DF/DV
      A = (SLØPE2 - SLØPE1)/SDV
      IF ((A * ØPTIØN) .GE. 0.0)  GØ TØ 500
      VPK = (V1 + V2 - SLØPE2/A)/2.0
      IF ((DF * ØPTIØN) .LT. 0.0)  GØ TØ 410
      DV = VPK - V2
      GØ TØ 110
C
  410 DV = VPK - V1
      GØ TØ 210
C         GET NEW PØINT
  500 IF (((F2 - F0) * ØPTIØN) .LT. 0.0)  GØ TØ 510
      DV=2.0*DV
      GØ TØ 110
  510 F2 = F0
```

FIGURE 14. SYMBOLIC LISTING OF SUBROUTINE MNX (cont.)

```
      V2 = V0
      DV = (V2 - V1)/2.0
      GØ TØ 110
C        ERRØR LØØP
  800 IF (ITAPE .EQ. 0)  GØ TØ 805
      WRITE (ITAPE,1)
      IF (TAPE .GT. 0)  GØ TØ 809
  805 PRINT 1
    1 FØRMAT(30H      FUNCTIØN GENERATØR ERRØR)
  809 ERRØR = 0.0
      ASSIGN 810 TØ N
      GØ TØ 900
  810 FLAG2 = FLAG2 + 1
      IF (FLAG2 .GE. MAXER)  GØ TØ 310
      IF (DV .EQ. 0.0)  GØ TØ 310
      DV = DV * SCALE
      GØ TØ 11
C        PRINT RØUTINE
  900 IF (ITAPE .EQ. 0)  GØ TØ 905
      WRITE (ITAPE,2)  ØPT,XADD,F2
      IF (TAPE .GT. 0)  GØ TØ 910
  905 PRINT 2,  ØPT,XADD,F2
    2 FØRMAT(1H ,6X,A6,6X,2E16.8)
  910 GØ TØ N, (90,810)
      END
```

FIGURE 14. SYMBOLIC LISTING OF SUBROUTINE MNX (cont.)

X. SUBROUTINE DMNX

Subroutine DMNX is identical to Subroutine MNX. It exists in order to allow for more than one minimization/maximization to be accomplished. That is, a separate deck is required for each minimization/maximization desired.

XI. SUBROUTINE PSTDY

This subroutine will perform equal increment perturbations of 1 or 2 parameters. One value is calculated on each entry to the subroutine which saves and restores the nominal values when the desired number of perturbations is completed. Parameter 1 is perturbed N times for each value of Parameter 2.

CALLING SEQUENCE

The calling sequence is:

```
CALL PSTDY (NPAR, PAR1, PAR1D, NPAR1D, P1BCI, PAR2,
                PAR2D, NPAR2D, P2BCI, FINI, ITAPE)
```

SOLUTION METHOD

1. FINI = 0.0

PFLAG is defined to be =1.0 by a DATA statement. It is used to test for the first entry to the subroutine:

2. If PFLAG < 0.0, go to 10

   If PFLAG ≥ 0.0, go to 3

3. PFLAG = -PFLAG

```
4.   P1SAVE = PAR1

5.   P2SAVE = PAR2

6.   NP1D = NPAR1D

7.   NP2D = NPAR2D

8.   If NPAR = 0, go to 35
     If NPAR ≠ 0, go to 9

9.   Go to 23

10.  PAR1 = PAR1 + PAR1D

11.  If NP1D = 0, go to 14
     If NP1D ≠ 0, go to 12

12.  NP1D = NP1D - 1

13.  Go to 23
```

Pass through first parameter has been completed, check for second parameter:

```
14.  NP1D = NPAR1D

15.  PAR1 = P1SAVE

16.  If NPAR < 2, go to 35
     If NPAR ≥ 2, go to 17

17.  PAR2 = PAR2 + PAR2D

18.  If NP2D = 0, go to 21
     If NP2D ≠ 0, go to 19
```

```
19.  NP2D = NP2D - 1
20.  Go to 23
```

Pass through second parameter has been completed:

```
21.  PAR2 = P2SAVE
22.  Go to 35
```

Write names and current values of parameters on ITAPE:

```
23.  IUNIT = $|ITAPE|$
24.  If NPAR $< 2$, go to 30
     If NPAR $\geq 2$, go to 25
25.  If IUNIT $= 0$, go to 28
     If IUNIT $\neq 0$, go to 26
26.  Write P1BCI, PAR1, P2BCI, PAR2 on tape IUNIT
27.  If ITAPE $> 0$, go to 34
     If ITAPE $\leq 0$, go to 28
28.  Print P1BCI, PAR1, P2BCI, PAR2 on the on-line printer
29.  Go to 34
30.  If IUNIT $= 0$, go to 33
     If IUNIT $\neq 0$, go to 31
31.  Write P1BCI, PAR1 on tape IUNIT
```

32. If ITAPE $>$ 0, go to 34

If ITAPE $\leq$ 0, go to 33

33. Print P1BCI, PAR1 on the on-line printer

Incomplete return:

34. Return

Completed return:

35. FINI = 1.0

36. PFLAG = 1.0

37. Return

SUBROUTINE PSTUDY NOMENCLATURE

Symbol	Description	Units	Reference
FINI	Flag for completion	—	CALL, 1, 35
ITAPE	Output unit(s)	—	CALL, 23, 27, 32,
IUNIT	Output tape	—	23, 25, 26, 30, 31
NP1D	Counter for number of perturbations of first parameter	—	6, 11, 12, 14
NP2D	Counter for number of perturbations of second parameter	—	7, 18, 19
NPAR	Number of parameters	—	CALL, 8, 16, 24
NPAR1D	Number of times to perturb first parameter	—	CALL, 6, 14
NPAR2D	Number of times to perturb second parameter	—	CALL, 7
P1BCI	2-part name for first parameter - Hollerith name, integer subscript	—	CALL, DIM, 26, 28, 31, 33
P2BCI	2-part name for second parameter - Hollerith name, integer subscript	—	CALL, DIM, 26, 28
P1SAVE	Saved nominal value of first parameter	—	4, 15
P2SAVE	Saved nominal value of second parameter	—	5, 21

SUBROUTINE PSTUDY NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PAR1	Location of first parameter	—	CALL, 4, 10, 15, 26, 28, 31, 33
PAR2	Location of second parameter	—	5, 17, 21, 26, 28
PAR1D	Perturbation value for first parameter	—	CALL, 10
PAR2D	Perturbation value for second parameter	—	CALL, 17
PFLAG	Flag for nominal or perturbed entry to routine	—	DATA, 2, 3, 36

```
$IBFTC PSTDY   FULIST,REF
C     PARAMETER STUDY RØUTINE
      SUBRØUTINE PSTDY(NPAR,PAR1,PAR1D,NPAR1D,P1BCI,PAR2,PAR2D,NPAR2D,
     $                P2BCI,FINI,ITAPE)
C           NPAR   - NØ ØF PARAMETERS   (1,2)
C           PAR1   - PARAMETER 1  LØCATIØN
C           PAR1D  - PARAMETER 1 DELTA
C           NPAR1D - NØ ØF TIMES TØ INCREMENT PARAMETER 1
C           P1BCI  - NAME ØF PARAMETER 1  HØLLERITH,INTEGER  2 WØRDS
C           PAR2   - PARAMETER 2  LØCATIØN
C           PAR2D  - PARAMETER 2  DELTA
C           NPAR2D - NØ ØF TIMES TØ INCREMENT PARAMETER 2
C           P2BCI  - NAME ØF PARAMETER 2  HØLLERITH,INTEGER  2 WØRDS
C           FINI   - FLAG FØR CØMPLETIØN  0-NØT DØNE  NØN-ZERØ-DØNE
C           ITAPE  - TAPE TØ WRITE  ØN
      DIMENSIØN  P1BCI(2),P2BCI(2)
      DATA  PFLAG/1.0/
      FINI = 0.0
      IF (PFLAG .LT. 0.0)  GØ TØ 200
C           FIRST PASS
      PFLAG = -PFLAG
      P1SAVE = PAR1
      P2SAVE = PAR2
      NP1D = NPAR1D
      NP2D = NPAR2D
      IF (NPAR .EQ. 0)  GØ TØ 900
      GØ TØ 500
C
  200 PAR1 = PAR1 + PAR1D
      IF (NP1D .EQ. 0)  GØ TØ 300
      NP1D = NP1D - 1
      GØ TØ 500
C
  300 NP1D = NPAR1D
      PAR1 = P1SAVE
      IF (NPAR .LT. 2)  GØ TØ 900
C     SECØND PARAMETER
      PAR2 = PAR2 + PAR2D
      IF (NP2D .EQ. 0)  GØ TØ 400
      NP2D = NP2D - 1
      GØ TØ 500
C
  400 PAR2 = P2SAVE
      GØ TØ 900
C
  500 IUNIT = IABS(ITAPE)
      IF (NPAR .LT. 2)  GØ TØ 510
      IF (IUNIT .EQ. 0)  GØ TØ 505
      WRITE (IUNIT,501)  P1BCI,PAR1,P2BCI,PAR2
      IF (ITAPE .GT. 0)  GØ TØ 800
```

FIGURE 15. SYMBOLIC LISTING OF SUBROUTINE PSTDY

```
  505 PRINT 501,  P1BCI,PAR1,P2BCI,PAR2
  501 FØRMAT(1H ,2X,A6,1H(,I4,3H) =,F14.4,3X,A6,1H(,I4,3H) =,F14.4)
      GØ TØ 800
  510 IF (IUNIT .EQ. 0)  GØ TØ 515
      WRITE (IUNIT,511)  P1BCI,PAR1
      IF (ITAPE .GT. 0)  GØ TØ 800
  515 PRINT 511,  P1BCI,PAR1
  511 FØRMAT(1H ,2X,A6,1H(,I4,3H) =,F14.4)
C          INCØMPLETED RETURN
  800 RETURN
C          CØMPLETED RETURN
  900 FINI = 1.0
      PFLAG = 1.0
      RETURN
      END
```

FIGURE 15. SYMBOLIC LISTING OF SUBROUTINE PSTDY (cont.)

XII. SUBROUTINE GAINS

This subroutine calculates the gains of up to 10 dependent variables with respect to up to 10 independent variables which are perturbed external to this routine using the parameter study routine (PSTUDY/PSTDY).

CALLING SEQUENCE

CALL GAINS (KFLAG)

This subroutine uses COMMON blocks TAPES, CONST and GAIND.

SOLUTION METHOD

KFLAG indicates whether this is the set-up entry to the routine.

1. If KFLAG $=0$, go to 27

   If KFLAG $\neq 0$, go to 2

Initialize:

2. IT $= |\text{STAPE}|$
3. Do 5, I $= 1, 10$

```
4.  IVARN(I) = 0.0
5.  DVARN(I) = 0.0
6.  Do 13, I = 1, 10
7.  If IVAR(I) = 0, go to 14
    If IVAR(I) ≠ 0, go to 8
8.  ISIZE = I
9.  CALL UNPACK (IVAR(I), ARRAY)
10. IVARL(I) = ADD
11. ADD = MASKA(IVAR(I))
12. CALL LOOKUP (ADD, IVARH(1, I))
13. IVARH(2, I) = IVARL(I) - ADD + 1
14. Do 25, I = 1, 10
15. If DVAR(I) = 0, go to 26
    If DVAR(I) ≠ 0, go to 16
16. JSIZE = I
17. CALL UNPACK (DVAR(I), ARRAY)
18. DVARL(I) = ADD
19. DVARR(I) = DEC
20. ADD = MASKA (DVAR(I))
21. CALL LOOKUP (ADD, DVARH(1, I))
22. DVARH(2, I) = DVARL(I) - ADD + 1
```

```
23.  DVARC(I) = TAGS(TAG)
24.  If TAG = 0, DVARC(I) = 1.0
     If TAG ≠ 0, DVARC(I) = DVARC(I)
25.  Continue
26.  M = 1
     Return
```

M indicates whether this entry gives nominal or perturbed values:

```
27.  If M = 0, go to 34
     If M ≠ 0, go to 28
28.  M = 0
```

Save nominal values:

```
29.  Do 30, I = 1, ISIZE
30.  IVARN(I) = CLA (IVARL(I))
31.  Do 33, J = 1, JSIZE
32.  CALL XEC(DVARR(J))
33.  DVARN(J) = (CLA(DVARL(J)))(DVARC(J))
     Return
```

Calculate gains:

```
34.  Do 49, I = 1, ISIZE
35.  IVARI = CLA(IVARL(I))
36.  If IT = 0, go to 39
     If IT ≠ 0, go to 37
```

```
37.  Write IVARH(1, I), IVARH(2, I), IVARN(I), IVARI
       on tape IT
38.  If STAPE > 0, go to 40
     If STAPE ≤ 0, go to 39
39.  Print IVARH(1, I), IVARH(2, I), IVARN(I), IVARI
       on the on-line printer
40.  Do 48, J = 1, JSIZE
41.  Call XEC(DVARR(J))
42.  DVARI = (CLA(DVARL(J)))(DVARC(J))
43.  PAR = (DVARI - DVARN(J))/(IVARI - IVARN(J))
44.  If IT = 0, go to 47
     If IT ≠ 0, go to 45
45.  Write DVARH(1,J), DVARH(2, J), DVARN(J), DVARI, PAR,
       on tape IT
46.  If STAPE > 0, go to 48
     If STAPE ≤ 0, go to 47
47.  Print DVARN(1, J), DVARH(2, J), DVARN(J), DVARI, PAR,
       on the on-line printer
48.  Continue
49.  Continue
     Return
```

SUBROUTINE GAINS NOMENCLATURE

Symbol	Description	Units	Reference
ADD	Address portion of DVAR(I), IVAR(I)	—	INTEGER, EQUIV, 10, 11, 12, 13, 18, 20, 21, 22
ARRAY	Array containing address, tag, decrement and prefix of DVAR(I), IVAR(I)	—	DIM, EQUIV, 9
DEC	Decrement portion of IVAR(I)	—	EQUIV, 19
DVARC	Array of dependent variable conversion factors	—	DIM, 23, 24, 33, 42
DVARH	Array of names of dependent variables	—	DIM, INTEGER, 21, 22, 45, 47
DVARI	Current value of some DVAR(I)	—	42, 43, 45, 47
DVARL	Array of locations of dependent variables	—	DIM, INTEGER, 18, 22, 33, 42
DVARN	Array of nominal values of dependent variables	—	DIM, 5, 33, 43, 45, 47
DVARR	Array of locations of routines to be executed to calculate DVARI	—	DIM, 19, 32, 41
DVAR	Input array specifying dependent parameters - "A"-prefix conversion code option	—	/GAIND/, INTEGER, 15, 17, 20
ISIZE	Number of independent parameters	—	8, 29, 34
IT	Output tape	—	2, 36, 37, 44, 45
IVARH	Array of names of independent variables	—	DIM, 12, 13, 37, 39

SUBROUTINE GAINS NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
IVARI	Current value of some IVAR(I)	—	REAL, 35, 37, 39, 43
IVARL	Array of locations of independent variables	—	DIM, 10, 13, 30, 35
IVARN	Array of nominal values of independent variables	—	DIM, REAL, 4, 30, 37, 39, 43
IVAR	Input array specifying independent parameters - "A"-prefix conversion code option	—	/GAIND/, 7, 9, 11
JSIZE	Number of dependent parameters	—	16, 31, 40
KFLAG	Indicator for set-up entry to GAINS	—	CALL, 1
M	Indicator for nominal or perturbation entry to GAINS	—	26, 27, 28
PAR	Partial of a dependent variable with respect to an independent variable	—	43, 45, 47
STAPE	Scratch tape - used for GAINS output	—	/TAPES/, INTEGER, 2, 38, 46
TAG	Indicator for conversion factor	—	INTEGER, EQUIV, 23, 24
TAGS	Array of conversion factors	—	/CONST/, 23

```
$IBFTC GAINS
      SUBROUTINE GAINS(KFLAG)
      COMMON/TAPES/ITAPE,OTAPE,STAPE
      INTEGER  OTAPE,STAPE
      COMMON/CONST/RSUB0,FLAT,GSUB0,GSUB1,OMEGA,PSL,CMAX,HMIN,HMAX,
     $              TAGS(7),INFIN,PI,PIO2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBO,RSUB0),(GSUBO,GSUB0)
      REAL  INFIN
      COMMON /GAIND/ FLPAR,IVAR(10),DVAR(10)
      INTEGER  FLPAR,DVAR
      DIMENSION  IVARN(10),DVARN(10),IVARL(10),DVARL(10),DVARC(10),
     $        DVARR(10),IVARH(2,10),DVARH(2,10)
      INTEGER  DVAR,DVARL,DVARH
      REAL  IVARN,IVARL
      DIMENSION  ARRAY(4)
      EQUIVALENCE  (ARRAY(1),ADD),(ARRAY(2),TAG),(ARRAY(3),DEC),
     $              (ARRAY(4),PRE)
      INTEGER  ADD,TAG
C
      IF (KFLAG .EQ. 0)  GO TO 100
C        INITIALIZE
      IT = IABS(STAPE)
      DO 10  I=1,10
      IVARN(I) = 0.0
   10 DVARN(I) = 0.0
      DO 20  I=1,10
      IF (IVAR(I) .EQ. 0)  GO TO 25
      ISIZE = I
      CALL UNPACK(IVAR(I),ARRAY)
      IVARL(I) = ADD
      ADD = MASKA(IVAR(I))
      CALL LOOKUP(ADD,IVARH(1,I))
   20 IVARH(2,I) = IVARL(I) - ADD + 1
   25 DO 30  I=1,10
      IF (DVAR(I) .EQ. 0)  GO TO 35
      JSIZE = I
      CALL UNPACK(DVAR(I),ARRAY)
      DVARL(I) = ADD
      DVARR(I) = DEC
      ADD = MASKA(DVAR(I))
      CALL LOOKUP(ADD,DVARH(1,I))
      DVARH(2,I) = DVARL(I) - ADD + 1
      DVARC(I) = TAGS(TAG)
      IF (TAG .EQ. 0)  DVARC(I) = 1.0
   30 CONTINUE
   35 M = 1
      RETURN
C
  100 IF (M .EQ. 0)  GO TO 200
      M = 0
```

FIGURE 16. SYMBOLIC LISTING OF SUBROUTINE GAINS

```
C           SAVE NØMINAL VALUES
      DØ 110  I=1,ISIZE
  110 IVARN(I) = CLA(IVARL(I))
      DØ 120  J=1,JSIZE
      CALL XEC(DVARR(J))
  120 DVARN(J) = CLA(DVARL(J)) * DVARC(J)
      RETURN
C           CALCULATE GAINS
  200 DØ 400  I=1,ISIZE
      IVARI = CLA(IVARL(I))
      IF (IT .EQ. 0)  GØ TØ 205
      WRITE (IT,1)  IVARH(1,I),IVARH(2,I),IVARN(I),IVARI
      IF (STAPE .GT. 0)  GØ TØ 210
  205 PRINT 1,   IVARH(1,I),IVARH(2,I),IVARN(I),IVARI
    1 FØRMAT(1H0,25H  INDEPENDENT VARIABLE = ,A6,1H(,I4,1H),
     $    18H  NØMINAL VALUE = ,E15.8,20H  PERTURBED VALUE = ,E15.8)
  210 DØ 300  J=1,JSIZE
      CALL XEC(DVARR(J))
      DVARI = CLA(DVARL(J)) * DVARC(J)
      PAR = (DVARI - DVARN(J))/(IVARI - IVARN(I))
      IF (IT .EQ. 0)  GØ TØ 250
      WRITE (IT,2)  DVARH(1,J),DVARH(2,J),DVARN(J),DVARI,PAR
      IF (STAPE .GT. 0)  GØ TØ 300
  250 PRINT 2,  DVARH(1,J),DVARH(2,J),DVARN(J),DVARI,PAR
    2 FØRMAT(1H ,26H       DEPENDENT VARIABLE = ,A6,1H(,I4,1H),
     $    18H  NØMINAL VALUE = ,E15.8,20H  PERTURBED VALUE = ,E15.8/
     $    1H ,20H          PARTIAL = ,E15.8)
  300 CØNTINUE
  400 CØNTINUE
      RETURN
      END
```

FIGURE 16. SYMBOLIC LISTING OF SUBROUTINE GAINS (cont.)

XIII. SUBROUTINE SDATA

This subroutine "presets" the various search data input to certain nominal values which may be changed on input if desired.

CALLING SEQUENCE

This subroutine is not executable; it has no calling sequence. It is composed only of data values to be loaded with the rest of the program before execution.

USAGE

Parameter Study Data Block /STUDY/

P1ADD1 = 0
P1ADD2 = 0
P1ADD3 = 0
DELP1 = 0
NDELP1 = 0
P2ADD1 = 0
P2ADD2 = 0
P2ADD3 = 0
DELP2 = 0
NDELP2 = 0

One-Parameter Search Input Block /SRCH1A/

XADD1 = BETA0
XADD2 = 0
XADD3 = 0
YADD = BETA,1
YSTAR = 70.0
DELTAX = 10.0
EPSI1 = .001
SLOPE = 1.0
MXIT1 = 30
MXERF = 5
MXERI = 5
SCALE = .5

Two-Parameter Search Input Block /SRCH2A/

X1ADD1 = EDOT4
X1ADD2 = 0
X1ADD3 = 0
X2ADD = BETA0
Y1ADD = H,2
Y2ADD = BETA,1
Y1STAR = 100.0
Y2STAR = 90.0
DELX1 = 10.0
DELX2 = 10.0

EPSI21 = .001

EPSI22 = .001

JACOB = 25

K = .5

L = .5

ES0 = 10

ES1 = 5

ES2 = 5

KDAMP = 1.0

NDAMP = 1.0

MXIT2 = 30

Min-Max Search Input Block /MNX1/

VADD1 = BETA0

VADD2 = 0

VADD3 = 0

ZADD = RANGE

MNXIND = 1.0

DELV = 0.0

MNXEP = .0001

MNXIT = 30

MNXER = 5

MNXSCL = .25

Min-Max Search Input Block /MNX2/ for Double Search Option

VADD12 = 0

VADD22 = 0

VADD32 = 0
ZADD2 = 0
MNXIN2 = 1.0
DELV2 = 0.0
MNXEP2 = .0001
MNXIT2 = 30
MNXER2 = 5
MNXSC2 = .25

One-Parameter Search Input Block/SRCH1B/ for Double Search Option

XADD12 = 0
XADD22 = 0
XADD32 = 0
YADD2 = 0
YSTAR2 = 0
DELTX2 = 10.0
EPSI12 = .001
SLOPE2 = 1.0
MXIT12 = 30
MXERF2 = 5
MXERI2 = 5
SCALE2 = .5

SUBROUTINE SDATA NOMENCLATURE

Symbol	Description	Units	Reference
BETA0	Kick at end of vertical rise	degrees	EQU, XADD1, X2ADD, VADD1
BETA	Angle between local horizontal and velocity vector	radians	EQU, YADD, Y2ADD
/DATA/	Common block containing BETA, H, RANGE	—	EQU
DELP1	Parameter 1 delta	—	/STUDY/
DELP2	Parameter 2 delta	—	/STUDY/
DELTAX	Independent variable increment	%	/SRCH1A/
DELTX2	Independent variable increment	%	/SRCH1B/
DELV2	Independent variable increment	%	/MNX2/
DELV	Independent variable increment	%	/MNX1/
DELX1	Independendnet variable 1 increment	%	/SRCH2A/
DELX2	Independent variable 2 increment	%	/SRCH2A/
EDOT4	Pitch rate, section 4	degrees	EQU, X1ADD1
EPSI1	Convergence tolerance for dependent variable	—	/SRCH1A/

SUBROUTINE SDATA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
EPSI12	Convergence tolerance for dependent variable	—	/SRCH1B/
EPSI21	Convergence tolerance for dependent variable 1	—	/SRCH2A/
EPSI22	Convergence tolerance for dependent variable 2	—	/SRCH2A/
ES0	Maximum consecutive errors for point 0	—	/SRCH2A/
ES1	Maximum consecutive errors for point 1	—	/SRCH2A/
ES2	Maximum consecutive errors for point 2	—	/SRCH2A/
H	Altitude	feet	EQU, Y1ADD
/INPUT/	Common block containing BETA0, EDOT4	—	EQU
JACOB	Jacobian counter	—	/SRCH2A/
K	Scale factor for point 0 if error	—	/SRCH2A/
KDAMP	Scale factor for damping predictions	—	/SRCH2A/
L	Scale factor for points 1, 2, if error	—	/SRCH2A/
MNXEP	Convergence tolerance for dependent variable	—	/MNX1/

SUBROUTINE SDATA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
MNXEP2	Convergence tolerance for dependent variable	—	/MNX2/
MNXER	Maximum allowable consecutive errors	—	/MNX1/
MNXER2	Maximum allowable consecutive errors	—	/MNX2/
MNXIN2	Min/max option indicator	—	/MNX2/
MNXIND	Min/max option indicator	—	/MNX1/
MNXIT	Maximum iterations	—	/MNX1/
MNXIT2	Maximum iterations	—	/MNX2/
MNXSC2	Scale factor for prediction in case of error	—	/MNX2/
MNXSCL	Scale factor for prediction in case of error	—	/MNX1/
MXERF	Maximum allowable consecutive errors	—	/SRCH1A/
MXERF2	Maximum allowable consecutive errors	—	/SRCH1B/
MXERI	Maximum allowable consecutive imaginary solutions	—	/SRCH1A/
MXERI2	Maximum allowable consecutive imaginary solutions	—	/SRCH1B/
MXIT1	Maximum iterations	—	/SRCH1A/

SUBROUTINE SDATA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
MXIT2	Maximum iterations	—	/SRCH2A/
MXIT12	Maximum iterations	—	/SRCH1B/
NDAMP	Number of times to damp predictions	—	/SRCH2A/
NDELP1	Number of times to increment parameter 1	—	/STUDY/
NDELP2	Number of times to increment parameter 2	—	/STUDY/
P1ADD1	Parameter 1 - location 1	—	/STUDY/
P1ADD2	Parameter 1 - location 2	—	/STUDY/
P1ADD3	Parameter 1 - location 3	—	/STUDY/
P2ADD1	Parameter 2 - location 1	—	/STUDY/
P2ADD2	Parameter 2 - location 2	—	/STUDY/
P2ADD3	Parameter 2 - location 3	—	/STUDY/
RANGE	Surface range	n.m.	EQU, ZADD
SCALE	Scale factor for prediction in case of error	—	/SRCH1A/
SCALE2	Scale factor for prediction in case of error	—	/SRCH1A/
SLOPE	Slope of dependent variable at desired solution	—	/SRCH1A/

SUBROUTINE SDATA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
SLOPE2	Slope of dependent variable at desired solution	—	/SRCH1B/
VADD1	Independent variable - location 1	—	/MNX1/
VADD2	Independent variable - location 2	—	/MNX1/
VADD3	Independent variable - location 3	—	/MNX1/
VADD12	Independent variable - location 1	—	/MNX2/
VADD22	Independent variable - location 2	—	/MNX2/
VADD32	Independent variable - location 3	—	/MNX2/
X1ADD1	Independent variable 1 - location 1	—	/SRCH2A/
X1ADD2	Independent variable 1 - location 2	—	/SRCH2A/
X1ADD3	Independent variable 1 - location 3	—	/SRCH2A/
X2ADD	Independent variable 2 - location	—	/SRCH2A/
XADD1	Independent variable - location 1	—	/SRCH1A/

SUBROUTINE SDATA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
XADD2	Independent variable - location 2	—	/SRCH1A/
XADD3	Independent variable - location 3	—	/SRCH1A/
XADD12	Independent variable - location 1	—	/SRCH1B/
XADD22	Independent variable - location 2	—	/SRCH1B/
XADD32	Independent variable - location 3	—	/SRCH1B/
YADD	Dependent variable location, conversion tag and routine to compute	—	/SRCH1A/
YADD2	Dependent variable location, conversion tag and routine to compute	—	/SRCH1B/
Y1ADD	Dependent variable 1 location, conversion tag and routine to compute	—	/SRCH2A/
Y2ADD	Dependent variable 2 location, conversion tag and routine to compute	—	/SRCH2A/
Y1STAR	Desired value of dependent variable 1	—	/SRCH2A/
Y2STAR	Desired value of dependent variable 2	—	/SRCH2A/

SUBROUTINE SDATA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
YSTAR	Desired value of dependent variable	—	/SRCH1A/
YSTAR2	Desired value of dependent variable	—	/SRCH1B/
ZADD	Dependent variable location, conversion tag and routine to compute	—	/MNX1/
ZADD2	Dependent variable location, conversion tag and routine to compute	—	/MNX2/

```
$IBMAP SDATA
BETAO  EQU     INPUT+7
BETA   EQU     DATA+18
EDØT4  EQU     INPUT+98
H      EQU     DATA+11
RANGE  EQU     DATA+45
*
*
*       PARAMETER STUDY INPUT BLØCK
*
STUDY  CØNTRL  STUDY
       USE     STUDY
       EVEN
P1ADD1 PZE                       PARAMETER 1 - LØCATIØN 1
P1ADD2 PZE                       PARAMETER 1 - LØCATIØN 2
P1ADD3 PZE                       PARAMETER 1 - LØCATIØN 3
DELP1  PZE                       PARAMETER 1 DELTA
NDELP1 PZE                       NØ ØF PERTURBATIØNS - PARAMETER 1
P2ADD1 PZE                       PARAMETER 2 - LØCATIØN 1
P2ADD2 PZE                       PARAMETER 2 - LØCATIØN 2
P2ADD3 PZE                       PARAMETER 2 - LØCATIØN 3
DELP2  PZE                       PARAMETER 2 DELTA
NDELP2 PZE                       NØ ØF PERTURBATIØNS - PARAMETER 2
*
*      ØNE-PARAMETER SEARCH INPUT BLØCK
*
SRCH1A CØNTRL  SRCH1A
       USE     SRCH1A
       EVEN
XADD1  PZE     BETAO             INDEPENDENT SEARCH VARIABLE
XADD2  PZE
XADD3  PZE
YADD   PZE     BETA,1            DEPENDENT SEARCH VARIABLE
YSTAR  DEC     70.0              DESIRED VALUE ØF DEPENDENT VARIABLE
DELTAX DEC     10.0              PER-CENT INCREMENT -- INDEPENDENT VAR
EPSI1  DEC     .001              CØNVERGENCE TØLERANCE
SLØPE  DEC     1.0               SLØPE ØF FUNCTIØN AT SØLUTIØN
MXIT1  DEC     30                MAX ITERATIØNS
MXERF  DEC     5                 MAX ERRØRS
MXERI  DEC     5                 MAX IMAGINARY SØLUTIØNS
SCALE  DEC     .5                ERRØR SCALE FACTØR
*
*
*      TWØ-PARAMETER SEARCH INPUT BLØCK
*
SRCH2A CØNTRL  SRCH2A
       USE     SRCH2A
       EVEN
X1ADD1 PZE     EDØT4              INDEPENDENT VARIABLE 1
X1ADD2 PZE
```

FIGURE 17. SYMBOLIC LISTING OF SUBROUTINE SDATA

```
X1ADD3 PZE
X2ADD  PZE    BETAO            INDEPENDENT VARIABLE 2
Y1ADD  PZE    H,2              DEPENDENT VARIABLE 1
Y2ADD  PZE    BETA,1           DEPENDENT VARIABLE 2
Y1STAR DEC    100.0            DESIRED VALUE - DEPENDENT VARIABLE 1
Y2STAR DEC    90.0             DESIRED VALUE - DEPENDENT VARIABLE 2
DELX1  DEC    10.0             PER-CENT INCREMENT - INDEP VARIABLE 1
DELX2  DEC    10.0             PER-CENT INCREMENT - INDEP VARIABLE 2
EPSI21 DEC    .001             CØNVERGENCE TØLERANCES
EPSI22 DEC    .001
JACØB  DEC    25               JACØBIAN CØUNTER
K      DEC    .5               SCALE FACTØRS
L      DEC    .5
ES0    DEC    10               ERRØR STØPS
ES1    DEC    5
ES2    DEC    5
KDAMP  DEC    1.0              DAMPING FACTØRS
NDAMP  DEC    1.0
MXIT2  DEC    30               MAX ITERATIØNS
*
*
*
*      MIN-MAX SEARCH INPUT BLØCK
*
MNX1   CØNTRL MNX1
       USE    MNX1
       EVEN
VADD1  PZE    BETAO            INDEPENDENT VARIABLE
VADD2  PZE
VADD3  PZE
ZADD   PZE    RANGE            DEPENDENT VARIABLE
MNXIND DEC    1.0              MIN-MAX INDICATØR
DELV   DEC    0.0              PER-CENT INCREMENT   0=100
MNXEP  DEC     0001            EPSILØN FØR CØNVERGENCE
MNXIT  DEC    30               MAX ITERATIØNS
MNXER  DEC    5                MAX ERRØRS
MNXSCL DEC    .25              SCALE FACTØR ØN ERRØR
*
*      MIN-MAX SEARCH BLØCK 2 FØR DØUBLE SEARCH
*
MNX2   CØNTRL MNX2
       USE    MNX2
       EVEN
VADD12 PZE
VADD22 PZE
VADD32 PZE
ZADD2  PZE
MNXIN2 DEC    1.0
DELV2  DEC    0.0
MNXEP2 DEC     0001
```

FIGURE 17. SYMBOLIC LISTING OF SUBROUTINE SDATA (cont.)

```
MNXIT2  DEC     30
MNXER2  DEC     5
MNXSC2  DEC     .25
*
*
*       ONE-PARAMETER SEARCH INPUT BLOCK 2 FOR DOUBLE SEARCH
*
SRCH1B  CONTRL  SRCH1B
        USE     SRCH1B
        EVEN
XADD12  PZE
XADD22  PZE
XADD32  PZE
YADD2   PZE
YSTAR2  DEC     0.0
DELTY2  DEC     10.0
EPSI12  DEC     .001
SLOPE2  DEC     1.0
MXIT12  DEC     30
MXERF2  DEC     5
MXERI2  DEC     5
SCALE2  DEC     .5
*
        END
```

FIGURE 17. SYMBOLIC LISTING OF SUBROUTINE SDATA (cont.)

XIV. SUBROUTINE TRAJ

This subroutine provides the main control for the trajectory simulation of Block II.

CALLING SEQUENCE

The subroutine calling sequence is:

```
CALL TRAJ
```

Subroutine TRAJ uses COMMON blocks TAPES, CONST, WFLAG, WORKD, GILL, DATA and DUMPY.

SOLUTION METHOD

Set error flag to zero:

```
1.  CALL SETERR(0)
```

Set discontinuity flag to -1:

```
2.  FLDSC = -1
```

Initialize variables:

```
3.  CALL INIT
```

```
Set up section and stage working data:

 4.  CALL SUBA

Dump COMMON blocks if desired:

 5.  CALL DUMPS (DUMP3, 10, OTAPE)

Check if error flag was set in Subroutine SUBA:

 6.  CALL CHKERR(Z)

 7.  If $Z \neq 0$, go to 37
     If $Z = 0$, go to 8

Check end-of-flight flag:

 8.  If FLEOF = .TRUE., go to 35
     If FLEOF = .FALSE., go to 9

Set end-of-section flag .FALSE.

 9.  FLEOS = .FALSE.

Compute derivatives:

10.  CALL NEWTON

Dump COMMON blocks if desired:

11.  CALL DUMPS (DUMP4, 10, OTAPE)
```

Check if error flag was set in Subroutine NEWTON:

12. CALL CHKERR(Z)

13. If $Z \neq 0$, go to 37

    If $Z = 0$, go to 14

Set value of last integration interval:

14. DTLST = DT

Set integration interval for section:

15. DT = DTSBS

Check convergence criteria and compute integration interval if required:

16. CALL ENDOS

Check if error flag was set in Subroutine ENDOS:

17. CALL CHKERR(Z)

18. If $Z \neq 0$, go to 37

    If $Z = 0$, go to 19

Write output:

19. CALL OUTPUT

Set print flag .FALSE.

```
20.  FLTP = .FALSE.
```

Calculate print time:

```
21.  If DTSBP > 0, go to 24
     If DTSBP ≤ 0, go to 22
22.  TSUBP = INFIN
23.  Go to 26
24.  TSUBP = (FLOAT(IFIX((TIME)/(DTSBP))))(DTSBP) + DTSBP
25.  If TSUBP - TIME ≤ .001, TSUBP = TSUBP + DTSBP
     If TSUBP - TIME > .001, go to 26
26.  Continue
```

Reverse sign of discontinuity flag:

```
27.  FLDSC = -FLDSC
```

Test sign of TIME

```
28.  If TIME = 0, TIME = -TIME
     If TIME ≠ 0, TIME =  TIME
```

Check end-of-section flag:

```
29.  If FLEOS = .TRUE., go to 4
     If FLEOS = .FALSE., go to 30
```

Check discontinuity flag:

30. If FLDSC $<$ 0, go to 10

If FLDSC $\geq$ 0, go to 31

Set discontinuity flag to zero:

31. FLDSC = 0

Integrate:

32. CALL INTEG (NEWTON, 12, TIME, DT, QUE, KPASS)

Dump COMMON blocks if desired:

33. CALL DUMPS (DUMP5, 10, OTAPE)

34. Go to 12

End-of-Flight Output

Write output:

35. CALL OUTPUT

Dump COMMON blocks if desired:

36. CALL DUMPS (DUMP6, 10, OTAPE)

Return

Trajectory Error

37. I = $|$ STAPE $|$

```
Dump COMMON blocks if desired:

38.  CALL DUMPS (DUMP7, 10, OTAPE)

Write error message:

39.  If $I = 0$, go to 42
     If $I \neq 0$, go to 40
40.  Write error message on output tape:
41.  If $STAPE > 0$, Return
     If $STAPE \leq 0$, go to 42
42.  Print error message on the on-line printer
     Return
```

SUBROUTINE TRAJ NOMENCLATURE

Symbol	Description	Units	Reference
DTLST	Last integration interval	sec	/DATA/, 14
DT	Integration interval	sec	/GILL/, 14, 15, 32
DTSBP	Section print interval	sec	/WORKD/, 21, 24, 25
DTSBS	Section integration interval	sec	/WORKD/, 15
DUMP3	Input array of COMMON blocks	—	/DUMPY/, 5
DUMP4	Input array of COMMON blocks	—	/DUMPY/, 11
DUMP5	Input array of COMMON blocks	—	/DUMPY/, 33
DUMP6	Input array of COMMON blocks	—	/DUMPY/, 36
DUMP7	Input array of COMMON blocks	—	/DUMPY/, 38
FLDSC	Discontinuity flag	—	/WFLAG/, INTEGER, 2, 27, 30, 31
FLEOF	End-of-flight flag	—	/WFLAG/, LOGICAL, 8
FLEOS	End-of-section flag	—	/WFLAG/, LOCICAL, 9, 29
FLTP	Print flag	—	/WFLAG/, LOGICAL, 20
INFIN	"Infinity"	—	/CONST/, REAL, 22
I	—	—	37, 39

SUBROUTINE TRAJ NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
KPASS	—	—	/GILL/, 32
OTAPE	Output tape	—	/TAPES/, INTEGER, 5, 11, 33, 36, 38
QUE	—	—	/GILL/, 32
STAPE	Output tape	—	/TAPES/, INTEGER, 37 41
TIME	Range time	sec	/GILL/, 24, 25, 28, 32
TSUBP	Print time	sec	/DATA/, 22, 24, 25
Z	Error flag	—	6, 7, 12, 13, 17, 18

```
$IBFTC TRAJ    FULIST,REF
      SUBRØUTINE TRAJ
      CØMMØN/TAPES/ITAPE,ØTAPE,STAPE
      INTEGER  ØTAPE,STAPE
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $            TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL   INFIN
      CØMMØN/WFLAG/FLEØF,FLEØS,FLTP,FLDSC,TYSC,FLTRW
      LØGICAL FLEØF,FLEØS,FLTP
      INTEGER  FLDSC,TYSC,FLTRW
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $            PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $            PADSK,VESK,PADRT1,PADRT2,
     $          SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $           ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL   ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $          W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $          DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $          WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $            PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $            GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $            QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $            CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $            ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN /DUMPY/ DUMPO(10),DUMP1(10),DUMP2(10),DUMP3(10),DUMP4(10),
     $            DUMP5(10),DUMP6(10),DUMP7(10),DUMP8(10),DUMP9(10)
      EXTERNAL  NEWTØN
C          TRAJECTØRY CØNTRØL
      CALL SETERR(0)
      FLDSC = -1
      CALL INIT
   10 CALL SUBA
      CALL DUMPS(DUMP3,10,ØTAPE)
      CALL CHKERR(Z)
      IF (Z .NE. 0.0)  GØ TØ 100
      IF (FLEØF)  GØ TØ 90
      FLEØS = .FALSE.
   20 CALL NEWTØN
      CALL DUMPS(DUMP4,10,ØTAPE)
   25 CALL CHKERR(Z)
      IF (Z .NE. 0.0)  GØ TØ 100
      DTLST = DT
      DT = DTSBS
      CALL ENDØS
      CALL CHKERR(Z)
```

FIGURE 18. SYMBOLIC LISTING OF SUBROUTINE TRAJ

```
      IF (Z .NE. 0.0)  GØ TØ 100
      CALL ØUTPUT
      FLTP = .FALSE.
C           CALCULATE PRINT TIME
      IF (DTSBP .GT. 0.0)  GØ TØ 40
      TSUBP = INFIN
      GØ TØ 45
   40 TSUBP = FLØAT(IFIX(TIME/DTSBP)) * DTSBP + DTSBP
      IF ((TSUBP - TIME) .LE. .001)  TSUBP = TSUBP + DTSBP
   45 CØNTINUE
      FLDSC = -FLDSC
      IF (TIME .EQ. 0.0)  TIME = -TIME
      IF (FLEØS)  GØ TØ 10
      IF (FLDSC .LT. 0)  GØ TØ 20
      FLDSC = 0
      CALL INTEG(NEWTØN,12,TIME,DT,QUE,KPASS)
      CALL DUMPS(DUMP5,10,ØTAPE)
      GØ TØ 25
C           END ØF FLIGHT
   90 CALL ØUTPUT
      CALL DUMPS(DUMP6,10,ØTAPE)
      RETURN
C           TRAJECTØRY ERRØR
  100 I = IABS(STAPE)
      CALL DUMPS(DUMP7,10,ØTAPE)
      IF (I .EQ. 0)  GØ TØ 101
      WRITE(I,1) Z
      IF (STAPE .GT. 0)  RETURN
  101 PRINT 1, Z
      RETURN
    1 FØRMAT(1H0/1H0,20X,32H******  TRAJECTØRY ERRØR  ******,10XA6)
      END
```

FIGURE 18. SYMBOLIC LISTING OF SUBROUTINE TRAJ (cont.)

XV. SUBROUTINE INIT

This subroutine sets up the initial conditions for Block II.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL INIT

Subroutine INIT uses COMMON blocks CONST, INPUT, WFLAG, WORKD, SAVE, GILL, DATA, COMP, TABLES and ENGINW.

SOLUTION METHOD

Set end-of-flight flag .FALSE.

1. FLEOF = .FALSE.

Initialize variables.

2. LSTAGE = 0

3. ACCEL = 0

4. PIT = 0

5. YAW = 0

6. Do 14, I = 1, 6

7. AVMR(I) = 0

8. WOR(I) = 0

9. WFR(I) = 0

10. KMACH(I) = 0

11. KPIT(I) = 0

12. KYAW(I) = 0

13. KTRUS(I) = 0

14. KWDOT(I) = 0

15. STIND = 0

16. SECTNO = 0

17. TYSC = 0

18. WIOWJ = 0

19. WIOWJS = 0

20. CATM = CMAX

Clear least significant half of integrated variables.

21. Do 23, I = 1, 12

22. GILLE(I) = 0

23. GILLD(2, I) = 0

24. KPASS = 0

Initialize time.

25. TIME = TI

Compute initial geocentric latitude.

26. PHIO = ATAN ((TAN((LATI)/(DPRAD))) $(1 - \mathrm{FLAT})^{2}$))

Convert initial longitude to radians.

27. LONGIR = (LONGI)/(DPRAD)

Compute initial Greenwich sidereal time.

28. SIDETO = (SIDETI)/(DPRAD) + LONGIR

Compute initial Julian date if required.

29. If DATE $\neq$ 0, CALL JULDT (DATE, (OMEGA)(60), JD, SIDETO)

    If DATE = 0, go to 30

Compute initial right ascension.

30. THETAO = SIDETO - LONGIR

31. If THETAO < 0, THETAO = THETAO + (2)(PI)

    If THETAO $\geq$ 0, THETAO = THETAO

Compute sines and cosines of PHIO and THETAO.

32. SPHIO = Sin(PHIO)

33. CPHIO = Cos(PHIO)

34. STHETO = Sin(THETAO)

35. CTHETO = Cos(THETAO)

Convert launch azimuth to radians.

36. AZIR = (AZI)/(DPRAD)

Compute initial Cartesian coordinates.

37. CALL G1OB (RSUB0, FLAT, PHIO, THETAO, HI, VI,
PIO2 - (BETAI)/(DPRAD), AZIR, ROX, ROY, ROZ,
VRX, VRY, VRZ, 1)

Set integrated coordinates to zero. XIN, YIN and ZIN are the differences between the instantaneous and the initial values of the Cartesian coordinates.

38. XIN = 0

39. YIN = 0

40. ZIN = 0

Compute inertial velocity components.

41. XDIN = VRX - (OMEGA)(ROY)

42. YDIN = VRY + (OMEGA)(ROX)

43. ZDIN = VRZ

Compute initial weight.

44. W = WI + WPL

Set velocity gain to zero.

45. VGAIN = 0

Return.

SUBROUTINE INIT NOMENCLATURE

Symbol	Description	Units	Reference
ACCEL	Vehicle acceleration	—	/GILL/, 3
AVMR	Average in-flight mixture ratio	—	/SAVE/, 7
AZI	Initial azimuth	degrees	/INPUT/, 36
AZIR	Initial azimuth	radians	/SAVE/, 36, 37
BETAI	Initial angle between local vertical and velocity vector	degrees	/INPUT/, 37
CATM	Speed of sound	ft/sec	/DATA/, 20
CMAX	Speed of sound above 300,000 feet	ft/sec	/CONST/, 20
CPHIO	Sine of PHIO	—	/SAVE/, 33
CTHETO	Sine of THETAO	—	/SAVE/, 35
DATE	Launch date	—	/INPUT/, 29
DPRAD	Conversion factor	deg/rad	/CONST/, 26, 27, 28, 36, 37
FLAT	Flattening	—	/CONST/, 26, 37
FLEOF	End-of-flight flag	—	LOGICAL, /WFLAG/, 1
GILLD	Storage array for integrated variables	—	DIM, EQUIV, 23
GILLE	Storage array for integrated variables	—	DIM, EQUIV, 22
HI	Initial height	ft	/INPUT/, 37

SUBROUTINE INIT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
I	Loop control indicator	—	6, 7, 8, 9, 10, 11, 12, 13, 14, 21, 22, 23
JD	Launch Julian date	days	/SAVE/, DP, 29
KMACH	Counter used in Subroutine NTERP	—	/TABLES/, 10
KPASS	Subroutine INTEG pass counter	—	/GILL/, 24
KPIT	Counter used in Subroutine NTERP	—	/TABLES/, 11
KTRUS	Counter used in Subroutine NTERP	—	/TABLES/, 13
KWDOT	Counter used in Subroutine NTERP	—	/TABLES/, 14
KYAW	Counter used in Subroutine NTERP	—	/TABLES/, 12
LATI	Initial geodetic latitude	degrees	REAL, /INPUT/, 26
LONGI	Initial west longitude	degrees	REAL, /INPUT/, 27
LONGIR	Initial west longitude	radians	REAL, 27, 28, 30
LSTAGE	Last stage indicator for Subroutine ENGSU	—	/ENGINW/, 2
OMEGA	Earth rotation rate	rad/sec	/CONST/, 29, 41, 42
PHIO	Initial geocentric latitude	radians	/SAVE/, 26, 32, 33, 37
PI	π	—	/CONST/, 31
PIO2	$(\pi)/(2)$	—	/CONST/, 37

SUBROUTINE INIT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PIT	Pitch angle	radians	/DATA/, 4
ROX	X-component of initial position vector	ft	/SAVE/, 37, 42
ROY	Y-component of initial position vector	ft	/SAVE/, 37, 41
ROZ	Z-component of initial position vector	ft	/SAVE/, 37
RSUB0	Earth equatorial radius	ft	EQUIV, /CONST/, 37
SECTNO	Section number	—	INTEGER, /WORKD/, 16
SIDETI	Initial sidereal time	degrees	/INPUT/, 28
SIDETO	Initial Greenwich sidereal time	radians	/SAVE/, 28, 29, 30
SPHIO	Sine of PHIO	—	/SAVE/, 29
STHETO	Sine of THETAO	—	/SAVE/, 34
STIND	Stage indicator	—	INTEGER, /WORKD/, 15
THETAO	Initial right ascension	radians	/SAVE/, 30, 31, 34, 35, 37
TIME	Range time	sec	EQUIV, /GILL/, 25
TI	Initial time	sec	/INPUT/, 25
TYSC	Section tally indicator	—	/WFLAG/, INTEGER, 17
VGAIN	Integrated velocity gain	ft/sec	/GILL/, 45
VI	Initial velocity	ft/sec	/INPUT/, 37

SUBROUTINE INIT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
VRX	X-component of initial relative velocity vector	ft/sec	/COMP/, 37, 41
VRY	Y-component of initial relative velocity vector	ft/sec	/COMP/, 37, 42
VRZ	Z-component of initial relative velocity vector	ft/sec	/COMP/, 37, 43
WFR	Fuel residual weight	lb	/SAVE/, 9
WIOWJ	Initial or jettison weight of section	lb	/WORKD/, 18
WIOWJS	Initial or jettison weight of stage	lb	/WORKD/, 19
WI	Initial weight	lb	/INPUT/, 44
WOR	Oxidizer residual weight	lb	/SAVE/, 8
WPL	Payload weight	lb	/INPUT/, 44
W	Instantaneous weight	lb	/GILL/, 44
XDIN	X-component of initial inertial velocity vector	ft/sec	/GILL/, 41
XIN	Difference between instantaneous and initial value of X-component of position vector	ft	/GILL/, 38
YAW	Yaw angle	radians	/DATA/, 5

SUBROUTINE INIT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
YDIN	Y-component of initial inertial velocity vector	ft/sec	/GILL/, 42
YIN	Difference between instantaneous and initial value of Y-component of position vector	ft	/GILL/, 39
ZDIN	Z-component of initial inertial velocity vector	ft/sec	/GILL/, 43
ZIN	Difference between instantaneous and initial value of Z-component of position vector	ft	/GILL/, 40

```
$IBFTC INIT    FULIST,REF
      SUBRØUTINE INIT
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $             TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL   INFIN
      CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,
     $             DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $             SECTS(20,12),STAGS(20,6),HEAD(11,13)
      REAL  LATI,LØNGI
      INTEGER  TARGET
      INTEGER  FLTR,FLSC,FLPRT
      CØMMØN/WFLAG/FLEØF,FLEØS,FLTP,FLDSC,TYSC,FLTRW
      LØGICAL FLEØF,FLEØS,FLTP
      INTEGER  FLDSC,TYSC,FLTRW
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $             PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $             PADSK,VESK,PADRT1,PADRT2,
     $          SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $            ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL  ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $             AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $             WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $           W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $           DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $           WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $             PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $             GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $             QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $             CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $             ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN/CØMP/THRSTX,THRSTY,THRSTZ,GRAVX,GRAVY,GRAVZ,
     $             DRAGX,DRAGY,DRAGZ,LIFTX,LIFTY,LIFTZ,
     $             VRX,VRY,VRZ
      REAL  LIFTX,LIFTY,LIFTZ
      CØMMØN/TABLES/TMACH(30,6),TCSBN(30,6),TCSBA(30,6),
     $             TTPIT(30,6),TPIT(30,6),TTYAW(30,6),TYAW(30,6),
     $             TTTRS(30,6),TTRUS(30,6),TTWDT(30,6),TWDØT(30,6),
     $             JCSBN(6),JCSBA(6),JPIT(6),JYAW(6),JTRUS(6),JWDØT(6),
     $             LPIT(6),LYAW(6),LTRUS(6),LWDØT(6),
     $             KMACH(6),KPIT(6),KYAW(6),KTRUS(6),KWDØT(6),
     $             IMACH(6),IPIT(6),IYAW(6),ITRUS(6),IWDØT(6)
      CØMMØN/ENGINW/PCBBR,WØCBB,WFCBB,WØGBB,WFGBB,
     $             PCBAR,WØCBR,WFCBR,WØGBR,WFGBR,
```

FIGURE 19. SYMBOLIC LISTING OF SUBROUTINE INIT

```
     $               PØNØM,PFNØM,TØNØM,TFNØM,FØPNØM,ETANØM,
     $               PIND,TREF,DREF,FPTPA,PTØG,PTFG,TSØ1,PTØ1,TSØ2,PTØ2,
     $               C1Ø,C2Ø,TSF1,PTF1,TSF2,PTF2,C1F,C2F,
     $               LØ,RØ,UHØ,LHØ,LF,RF,UHF,LHF(5),
     $               NØFE,FTURB,FØPT,KURF,DTBUP,VØCØFF,VFCØFF,EPVØL,
     $               DTDCY,WMSDT,WMØSD,WMFSD,WMØED,WMFED,RESØ,RESF,
     $               ASUBT,EPSI,ETACF,ETACS,TØI,CTØ,TFI,CTF,
     $               VØBB,VFBB,HØCØL,HFCØL,PARAML,VPARL,
     $               WKPC(6),WKWØC(6),WKWFC(6),WKWØG(6),WKWFG(6),
     $               TBLDUP,TEND,FLGHD,FLGSD,LSTAGE,CSTRN
      LØGICAL  FLGHD,FLGSD
      REAL  LØ,LF,NØFE,KURF
      INTEGER  PIND,FPTPA,UHØ,UHF
      DIMENSIØN  GILLD(2,13),GILLE(13)
      EQUIVALENCE  (GILLD(1,1),TIME),(GILLE(1),DT)
      REAL  LØNGIR
      FLEØF = .FALSE.
      ISTAGE = 0
      ACCEL = 0.0
      PIT = 0.0
      YAW = 0.0
      DØ 10  I=1,6
      AVMR(I) = 0.0
      VØR(I) = 0.0
      VFR(I) = 0.0
      KMACH(I) = 0
      KPIT(I) = 0
      KYAW(I) = 0
      KTRUS(I) = 0
   10 KWDØT(I) = 0
      STIND = 0
      SECTNØ = 0
      TYSC = 0
      WIØWJ = 0.0
      WIØWJS = 0.0
      CATM = CMAX
C        CLEAR LEAST SIGNIFICANT HALF ØF INTEGRATED VARIABLES
      DØ 20  I=1,13
      GILLE(I) = 0.0
   20 GILLD(2,I) = 0.0
      KPASS = 0
C        SET UP INITIAL VALUES ØF INTEGRATED VARIABLES
      TIME = TI
      PHIØ = ATAN(TAN(LATI/DPRAD) * (1.0 - FLAT)**2)
      LØNGIR = LØNGI/DPRAD
      SIDETØ = SIDETI/DPRAD + LØNGIR
      IF (DATE .NE. 0.0)  CALL  JULDT(DATE,ØMEGA*60.0,JD,SIDETØ)
      THETAØ = SIDETØ - LØNGIR
      IF (THETAØ .LT. 0.0)  THETAØ = THETAØ + 2.0*PI
      SPHIØ = SIN(PHIØ)
```

FIGURE 19. SYMBOLIC LISTING OF SUBROUTINE INIT (cont.)

```
      CPHIØ = CØS(PHIØ)
      STHETØ = SIN(THETAØ)
      CTHETØ = CØS(THETAØ)
      AZIR = AZI/DPRAD
      CALL G1ØB(RSUBO,FLAT,PHIØ,THETAØ,HI,VI,PIØ2-BETAI/DPRAD,
     $               AZIR,RØX,RØY,RØZ,VRX,VRY,VRZ,1)
      XIN = 0.0
      YIN = 0.0
      ZIN = 0.0
      XDIN = VRX - ØMEGA * RØY
      YDIN = VRY + ØMEGA * RØX
      ZDIN = VRZ
      V = VI + VPL
      VGAIN = 0.0
      RETURN
      END
```

FIGURE 19. SYMBOLIC LISTING OF SUBROUTINE INIT (cont.)

XVI. SUBROUTINE SUBA

This routine sets up the current section and stage data and moves it to working storage.

CALLING SEQUENCE

The calling sequence is:

CALL SUBA

Subroutine SUBA uses COMMON blocks CONST, INPUT, WFLAG, WORKD, SAVE, GILL, DATA and COMP.

SOLUTION METHOD

Save stage indicator:

1. LSTIND = STIND

Save section print interval:

2. SDTSBP = DTSBP

Save stage initial or jettison weight:

3. T = WIOWJS

```
Increment section tally indicator:

4.  $TYSC = TYSC + 1$

Check for end of flight:

5.  If $TYSC \le FLSC$, go to 7
    If $TYSC > FLSC$, go to 6

6.  FLEOF = .TRUE.
    Return

Check for section jettison weight:

7.  If $WIOWJ < 0.0$, $W = W + WIOWJ$
    If $WIOWJ \ge 0.0$, $W = W$

Search for next section:

8.  $I = SECTNO + 1$

9.  Do 10 J = 1, 12

10. If $I = ISECT(16, J)$, go to 14
    If $I \ne ISECT(16, J)$, continue

11. $I = I + 1$

12. If $I < 50$, go to 9
    If $I \ge 50$, go to 13
```

Cannot find next section. Set error flag:

```
13.  CALL SETERR (4HSUBA)
     Return
```

Move section data:

```
14.  Do 15, I = 1, 20
15.  WORK(I) = SECTS(I, J)
```

Execute optional routine before moving stage data:

```
16.  CALL XEC (PADRT1)
```

Check for new stage:

```
17.  If STIND = LSTIND, go to 21
     If STIND ≠ LSTIND, go to 18
```

Move in new stage data after jettisoning any stage jettison weight:

```
18.  If T < 0.0, W = W + T
     If T ≥ 0.0, W = W
19.  Do 20, I = 1, 20
20.  WORK(I + 20) = STAGS(I, STIND)
```

Execute additional optional routine after moving stage data (if new stage):

```
21.  CALL XEC (PADRT2)

Check for section skip option:

22.  If PADSK = 0, go to 30
     If PADSK ≠ 0, go to 23

23.  CALL UNPACK (PADSK, ARRAY)

24.  CALL XEC (DEC)

25.  CONV = TAGS(TAG)

26.  If TAG = 0, CONV = 1.0
     If TAG ≠ 0, CONV = CONV

27.  V = (CLA(ADD))(CONV)

If PRE = 0,1,2,3  VSKIP = 1.0.   If PRE = 4,5,6,7,  VSKIP = -1.0.

28.  VSKIP = -SIGN(1.0, FLOAT(PRE) - 3.5)

29.  If (V - VESK)(VSKIP) > 0.0, go to 8
     If (V - VESK)(VSKIP) ≤ 0.0, go to 30

If section end parameter is zero, use section time:

30.  If PARAM ≠ 0, go to 34
     If PARAM = 0, go to 31

31.  VE = VE + TIME

32.  PARAM = LOC(TIME)
```

```
33.  Go to 39
```

If section end parameter is W and tag is 1, ending value does not include payload. If value is zero, use stage burn-out weight:

```
34.  CALL UNPACK (PARAM, ARRAY)
35.  If $ADD \neq LOC(W)$, go to 39
     If $ADD = LOC(W)$, go to 36
36.  If $VE = 0.0$, $VE = WBOS$
     If $VE \neq 0.0$, $VE = VE$
37.  If $TAG = 1$, $VE = VE + WPL$
     If $TAG \neq 1$, $VE = VE$
38.  $PARAM = LOC(W)$
```

If print interval is zero, use print interval of previous section:

```
39.  If $DTSBP = 0.0$, $DTSBP = SDTSBP$
     If $DTSBP \neq 0.0$, $DTSBP = DTSBP$
```

Convert pitch and yaw kicks and turning rates to radians:

```
40.  $PKIK = (PKIK)/(DPRAD)$
41.  $PDOT = (PDOT)/(DPRAD)$
42.  $YKIK = (YKIK)/(DPRAD)$
43.  $YDOT = (YDOT)/(DPRAD)$
```

If section 2, kick BETA by BETA0:

```
44.  If TYSC ≠ 2,  go to 52
     If TYSC = 2,  go to 45
45.  BETA = (BETAI + BETA0)/(DPRAD)
```

Determine inertial velocity:

```
46.  GAMMA = PIO2 - BETA
47.  AZ = AZIR
48.  CALL G1OB (RSUBO, FLAT, PHI, THETA, HATM, VSUBR,
                    GAMMA, AZ, U, U, U, VRX, VRY, VRZ, 1)
49.  XDIN = VRX - (OMEGA)(ROY + YIN)
50.  YDIN = VRY + (OMEGA)(ROX + XIN)
51.  ZDIN = VRZ
52.  If STIND = LSTIND,  go to 63
     If STIND ≠ LSTIND,  go to 53
```

Set up new stage values of integrated variables:

```
53.  TSTG = TIME
54.  WO(1) = WOIS
55.  WO(2) = 0.0
56.  WF(1) = WFIS
57.  WF(2) = 0.0
```

```
58.  WPROP(1) = WPRPS
59.  WPROP(2) = 0.0
60.  TBURN(1) = 0.0
61.  TBURN(2) = 0.0
62.  If WIOWJS > 0.0, W = WIOWJS + WPL
     If WIOWJS ≤ 0.0, W = W
Set up section saved data:
63.  If WIOWJ > 0.0, W = WIOWJ + WPL
     If WIOWJ ≤ 0.0, W = W
64.  TSUBS = TIME
65.  PSUBS = PIT
66.  YSUBS = YAW
Store section tally in sense lights:
67.  N = TYSC
68.  CALL SLITE(0)
69.  Do 71, I = 1, 4
70.  If MOD(N, 2) ≠ 0, CALL SLITE(5 - I)
     If MOD(N, 2) = 0, go to 71
71.  N = N/2
     Set up engine working data:
72.  CALL ENGSU
     Return
```

SUBROUTINE SUBA NOMENCLATURE

Symbol	Description	Units	Reference
ADD	Address portion of PADSK or PARAM	—	INTEGER, EQUIV, 27, 35
ARRAY	Array containing address, tag, decrement and prefix of PADSK or PARAM	—	DIM, EQUIV, 23, 34
AZ	Azimuth	radians	/DATA/, 47, 48
AZIR	Initial azimuth	radians	/SAVE/, 47
BETA	Angle between velocity vector and the local vertical	radians	/DATA/, 45, 46
BETA0	Kick angle at end of vertical rise	degrees	/INPUT/, 45
BETAI	Initial BETA	degrees	/INPUT/, 45
CONV	Conversion factor	—	26, 27
DEC	Decrement portion of PADSK or PARAM	—	INTEGER, EQUIV, 24
DPRAD	Degrees per radian conversion factor	deg/rad	/CONST/, 40, 41, 42, 43, 45
DTSBP	Section print interval	sec	/WORKD/, 2, 39
FLAT	Flattening of the Earth	—	/CONST/, 48
FLEOF	End of flight flag	—	LOGICAL, /WFLAG/, 6
FLSC	Flag for number of sections	—	/INPUT/, INTEGER, 5
GAMMA	Angle between velocity vector and local horizontal	radians	/DATA/, 46, 48

SUBROUTINE SUBA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
HATM	Altitude above sea level	ft	/DATA/, 48
I	Counter	—	8, 10, 11, 12
ISECT	SECTS	—	EQUIV, DIM, 10
LSTIND	Stage indicator	—	1, 17, 52
N	Counter	—	67, 70, 71
OMEGA	Earth rotation rate	rad/sec	/CONST/, 49, 50
PADRT1	Location of optional subroutine to be executed	—	/WORKD/, 16
PADRT2	Location of optional subroutine to be executed	—	/WORKD/, 21
PADSK	Parameter for section skip	—	/WORKD/, 22, 23
PARAM	Parameter for end of section	—	INTEGER, EQUIV, /WORKD/, 30, 32, 34, 38
PDOT	Pitch rate	deg/sec or rad/sec	/WORKD/, 41
PHI	Geocentric latitude	radians	/DATA/, 48
PIO2	$(\pi)/(2)$	—	/CONST/, 46
PIT	Pitch angle	radians	/DATA/, 65
PKIK	Pitch kick	degrees or radians	/WORKD/, 40

SUBROUTINE SUBA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PRE	Prefix of PARAM and PADSK	—	INTEGER, EQUIV, 28
PSUBS	Pitch angle at beginning of section	radians	/SAVE/, 65
ROX	X-coordinate of the launch site	ft	/SAVE/, 49
ROY	Y-coordinate of the launch site	ft	/SAVE/, 50
RSUBO	Radius of the Earth	ft	EQUIV, /CONST/, 48
SDTSBP	Print interval of previous section	sec	2, 39
SECTNO	Section sequence number	—	INTEGER, /WORKD/, 8
SECTS	Input section data array	—	EQUIV, /INPUT/, 15
STAGS	Input stage data array	—	/INPUT/, 20
STIND	Stage indicator	—	INTEGER, /WORKD/, 1, 17, 20, 52
T	Temporary storage	—	3, 18
TAG	Tag portion of PADSK and PARAM	—	INTEGER, EQUIV, 25, 26, 37
TAGS	Array of conversion factors	—	/CONST/, 25
TBURN	Stage burning time	sec	/GILL/, 60, 61

SUBROUTINE SUBA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
THETA	Local sidereal time	radians	/DATA/, 48
TIME	Time	sec	/GILL/, 31, 32, 53, 64
TSTG	Time at start of stage	sec	/SAVE/, 53
TSUBS	Time at start of section	sec	/SAVE/, 64
TYSC	Section tally	—	INTEGER, 4, 5, 44, 67
U	Dummy parameter	—	48
V	Value of section skip parameter	—	27, 29
VE	Value to end section	—	/WORKD/, 31, 36, 37
VESK	Value to skip section	—	/WORKD/, 29
VRX	X-component of relative velocity	ft/sec	/COMP/, 48, 49
VRY	Y-component of relative velocity	ft/sec	/COMP/, 48, 50
VRZ	Z-component of relative velocity	ft/sec	/COMP/, 48, 51
VSKIP	Indicator for section skip	—	28, 29
VSUBR	Relative velocity	ft/sec	/DATA/, 48
W	Total weight	lbs	/GILL/, 7, 18, 35, 38, 62, 63
WBOS	Stage burnout weight	lbs	/WORKD/, 36

SUBROUTINE SUBA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WF	Fuel weight	lbs	/GILL/, 56, 57
WFIS	Initial stage fuel weight	lbs	/WORKD/, 56
WIOWJ	Section initial or jettison weight	lbs	/WORKD/, 7, 63
WIOWJS	Stage initial or jettison weight	lbs	/WORKD/, 3, 62
WO	Oxidizer weight	lbs	/GILL/, 54, 55
WOIS	Initial stage oxidizer weight	lbs	/WORKD/, 54
WORK	Dummy array	—	DIM, EQUIV, 15, 20
WPL	Payload weight	lbs	/INPUT/, 37, 62, 63
WPROP	Miscellaneous propellant weight	lbs	/GILL/, 58, 59
WPRPS	Initial stage miscellaneous propellant weight	lbs	/WORKD/, 58
XDIN	X-component of inertial velocity	ft/sec	/GILL/, 49
XIN	Difference between instantaneous and initial value of X-component of position vector	ft	/GILL/, 50
YAW	Yaw angle	radians	/DATA/, 66
YDIN	Y-component of inertial velocity	ft/sec	/GILL/, 50
YDOT	Yaw rate	deg/sec or rad/sec	/WORKD/, 43

SUBROUTINE SUBA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
YIN	Difference between instantaneous and initial value of Y-component of position vector	ft	/GILL/, 49
YKIK	Yaw kick	degrees or radians	/WORKD/, 42
YSUBS	Yaw angle at start of section	radians	/SAVE/, 66
ZDIN	Z-component of inertial position	ft	/GILL/, 51

```
$IBFTC SUBA    FULIST,REF
      SUBRØUTINE SUBA
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $               TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE   (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL   INFIN
      CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,
     $               DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $               SECTS(20,12),STAGS(20,6),HEAD(11,15)
      REAL  LATI,LØNGI
      INTEGER  TARGET
      INTEGER  FLTR,FLSC,FLPRT
      CØMMØN/WFLAG/FLEØF,FLEØS,FLTP,FLDSC,TYSC,FLTRW
      LØGICAL FLEØF,FLEØS,FLTP
      INTEGER  FLDSC,TYSC,FLTRW
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $               PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $               PADSK,VESK,PADRT1,PADRT2,
     $             SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $              ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL  ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $               AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $               WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $             W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $             DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $             WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $               PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $               GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $               QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $               CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $               ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN /CØMP/THRSTX,THRSTY,THRSTZ,GRAVX,GRAVY,GRAVZ,
     $               DRAGX,DRAGY,DRAGZ,LIFTX,LIFTY,LIFTZ,
     $               VRX,VRY,VRZ
      REAL   LIFTX,LIFTY,LIFTZ
      DIMENSIØN  WØRK(20),ISECT(20,12)
      EQUIVALENCE  (WØRK(1),PARAM),(ISECT(1,1),SECTS(1,1))
      DIMENSIØN  ARRAY(4)
      EQUIVALENCE  (ARRAY(1),ADD),(ARRAY(2),TAG),(ARRAY(3),DEC),
     $              (ARRAY(4),PRE)
      INTEGER  PARAM,ADD,TAG,DEC,PRE
C
      LSTIND = STIND
      SDTSBP = DTSBP
```

FIGURE 20. SYMBOLIC LISTING OF SUBROUTINE SUBA

```
      T = WIØWJS
      TYSC = TYSC + 1
      IF (TYSC .LE. FLSC)  GØ TØ 10
      FLEØF = .TRUE.
      RETURN
C
   10 IF (WIØWJ .LT. 0.0)  W = W + WIØWJ
  100 I = SECTNØ + 1
   11 DØ 12  J=1,12
   12 IF (I .EQ. ISECT(16,J))  GØ TØ 13
      I = I + 1
      IF (I .LT. 50)  GØ TØ 11
      CALL SETERR(4HSUBA)
      RETURN
C
   13 DØ 14  I=1,20
   14 WØRK(I) = SECTS(I,J)
      CALL XEC(PADRT1)
      IF (STIND .EQ. LSTIND)  GØ TØ 20
      IF (T .LT. 0.0)  W = W + T
      DØ 15  I=1,20
   15 WØRK(I+20) = STAGS(I,STIND)
C
   20 CALL XEC(PADRT2)
C        CHECK FØR SECTIØN SKIP
      IF (PADSK .EQ. 0.0)  GØ TØ 30
      CALL UNPACK(PADSK,ARRAY)
      CALL XEC(DEC)
      CØNV = TAGS(TAG)
      IF (TAG .EQ. 0)  CØNV = 1.0
      V = CLA(ADD) * CØNV
      VSKIP = -SIGN(1.0,FLØAT(PRE)-3.5)
      IF ((V-VESK)*VSKIP .GT. 0.0)  GØ TØ 100
C
   30 IF (PARAM .NE. 0)  GØ TØ 40
      VE = VE + TIME
      PARAM = LØC(TIME)
      GØ TØ 50
C
   40 CALL UNPACK(PARAM,ARRAY)
      IF (ADD .NE. LØC(W))  GØ TØ 50
      IF (VE .EQ. 0.0)  VE = WBØS
      IF (TAG .EQ. 1)  VE = VE + WPL
      PARAM = LØC(W)
C
   50 IF (DTSBP .EQ. 0.0)  DTSBP = SDTSBP
      PKIK = PKIK/DPRAD
      PDØT = PDØT/DPRAD
      YKIK = YKIK/DPRAD
      YDØT = YDØT/DPRAD
```

FIGURE 20. SYMBOLIC LISTING OF SUBROUTINE SUBA (cont.)

```
      IF (TYSC .NE. 2)  GØ TØ 55
      BETA = (BETAI + BETAO)/DPRAD
      GAMMA = PIØ2 - BETA
      AZ = AZIR
      CALL G1ØB(RSUBØ,FLAT,PHI,THETA,HATM,VSUBR,GAMMA,AZ,U,U,U,VRX,
     $              VRY,VRZ,1)
      XDIN = VRX - ØMEGA * (RØY + YIN)
      YDIN = VRY + ØMEGA * (RØX + XIN)
      ZDIN = VRZ
   55 IF (STIND .EQ. LSTIND)  GØ TØ 60
      TSTG = TIME
      WØ(1) = WØIS
      WØ(2) = 0.0
      WF(1) = WFIS
      WF(2) = 0.0
      WPRØP(1) = WPRPS
      WPRØP(2) = 0.0
      TBURN(1) = 0.0
      TBURN(2) = 0.0
      IF (WIØWJS .GT. 0.0)  W = WIØWJS + WPL
   60 IF (WIØWJ .GT. 0.0)  W = WIØWJ + WPL
      TSUBS = TIME
      PSUBS = PIT
      YSUBS = YAW
      N = TYSC
      CALL SLITE(0)
      DØ 90  I=1,4
      IF (MØD(N,2) .NE. 0)  CALL SLITE(5-I)
   90 N = N/2
      CALL LIGSU
      RETURN
      END
```

FIGURE 20. SYMBOLIC LISTING OF SUBROUTINE SUBA (cont.)

XVII. SUBROUTINE ENDOS

This subroutine calculates the integration interval needed to converge on the end-of-section. In the case of a liquid engine influence coefficient thrust and flow rate simulation, it also converges on thrust build-up and decay times and shutdown volumes.

CALLING SEQUENCE

CALL ENDOS

SOLUTION METHOD

Check for influence coefficient thrust option.

1. If THIND $<$ 4, go to 27

   If THIND $\geq$ 4, go to 2

Check for hold-down before lift-off.

2. If PARAML $=$ 0, go to 9

   If PARAML $\neq$ 0, go to 3

If not initial time, skip setup for hold-down.

```
3.  If TIME $\neq$ TI, go to 9
    If TIME = TI, go to 4
```

Set up for convergence on lift-off criteria.

```
4.  FLGHD = .TRUE.
5.  IPARAM = PARAM
6.  VES = VE
7.  PARAM = PARAML
8.  VE = VPARL
```

Check for thrust build-up simulation.

```
9.  If DTBUP = 0.0, go to 19
    If DTBUP $\neq$ 0.0, go to 10
10. If $|$TIME - TBLDUP$|$ $>$ P001, go to 14
    If $|$TIME - TBLDUP$|$ $\leq$ P001, go to 11
```

Converged on thrust build-up time. Set discontinuity flag and turn off hold-down flag if no other lift-off criteria.

```
11. If FLDSC = 0, FLDSC = 1
    If FLDSC $\neq$ 0, FLDSC = FLDSC
12. If PARAML $\neq$ PARAM, FLGHD = .FALSE.
    If PARAML = PARAM, FLGHD = FLGHD
13. Go to 27
```

Not converged on thrust build-up time. Check if thrust build-up time has been exceeded.

14\. If $TIME > TBLDUP$, go to 19

If $TIME \leq TBLDUP$, go to 15

Calculate integration interval to converge on thrust build-up time. Set hold-down flag if first section.

15\. If $TYSC = 1$, $FLGHD =$ .TRUE.

If $TYSC \neq 1$, $FLGHD = FLGHD$

16\. $DELT = TBLDUP - TIME$

17\. If $DELT < DT$, $DT = DELT$

If $DELT \geq DT$, $DT = DT$

18\. Go to 27

Calculate integration interval to converge on shutdown volumes.

19\. If $FLGSD =$ .TRUE., go to 27

If $FLGSD \neq$ .TRUE., go to 20

20\. If $|VO - VOCOFF| \leq EPVOL$, $FLCNV =$ .TRUE.

If $|VO - VOCOFF| > EPVOL$, $FLCNV = FLCNV$

21\. If $|VF - VFCOFF| \leq EPVOL$, $FLCNV =$ .TRUE.

If $|VF - VFCOFF| > EPVOL$, $FLCNV = FLCNV$

```
22.  If FLCNV = .TRUE., go to 27
     If FLCNV ≠ .TRUE., go to 23
```

Not converged on shutdown volumes.

```
23.  VODOT = (WODOT)/(RHOO)
24.  VFDOT = (WFDOT)/(RHOF)
25.  DELT = AMIN1((VOCOFF - VO)/(VODOT),
               (VFCOFF - VF)/(VFDOT))
```

AMIN1 is a FORTRAN function which selects the minimum value.

```
26.  If DELT < DT, DT = DELT
     If DELT ≥ DT, DT = DT
```

See if propellant depletion is desired section ending criterion.

```
27.  If PARAM ≠ IND, go to 30
     If PARAM = IND, go to 28
28.  If FLCNV = .TRUE., go to 54
     If FLCNV ≠ .TRUE., go to 29
29.  Go to 73
```

Propellant depletion not desired section ending criterion.

```
30.  CALL UNPACK (PARAM, ARRAY)
```

Execute routine to calculate ending parameter.

```
31.  CALL XEC(DEC)
```

Check for error:

```
32.  CALL CHKERR(Z)

33.  If Z ≠ 0.0,  go to 69

     If Z = 0.0,  go to 34
```

Calculate current value of ending parameter.

```
34.  Z = TAGS(TAG)

35.  If TAG = 0,  Z = 1.0

     If TAG ≠ 0,  Z = Z

36.  PE = (Z)(CLA(ADD))

37.  If FLDSC ≥ 0,  go to 40

     If FLDSC < 0,  go to 38
```

Exit without executing if second side of discontinuity.

```
38.  ITER = 0

39.  Go to 94
```

Test for convergence on end-of-section parameter.

```
40.  DELPE = VE - PE

41.  DELPN = PE - PELST

42.  If DELPN = 0.0,  go to 98

     If DELPN ≠ 0.0,  go to 43
```

43. SLOPE = (DELPN)/(DTLST)

44. SLPIND = FLOAT(ISIGN(1, 4 - PRE))

FLOAT is a FORTRAN function to convert an integer to a floating point number.

ISIGN is a FORTRAN function. The use of ISIGN(A, B) assigns the algebraic sign of B to the magnitude of A.

45. If PRE = 0 or PRE = 4, SLPIND = 0.0

If PRE $\neq$ 0 and PRE $\neq$ 4, SLPIND = SLPIND

46. If SLPIND $<$ 0, go to 47

If SLPIND = 0, go to 50

If SLPIND $>$ 0, go to 49

Slope indicator is less than 0.

47. If SLOPE $>$ 0.0, go to 73

If SLOPE $\leq$ 0.0, go to 48

48. Go to 50

Slope indicator is greater than 0.

49. If SLOPE $<$ 0.0, go to 73

If SLOPE $\geq$ 0.0, go to 50

Check for convergence.

50. If $VE \neq 0.0$, go to 53

If $VE = 0.0$, go to 51

51. If $|PE| \leq EPSILN$, go to 54

If $|PE| > EPSILN$, go to 52

52. Go to 73

53. If $|(DELPE)/(VE)| > EPSILN$, go to 73

If $|(DELPE)/(VE)| \leq EPSILN$, go to 54

Converged on end-of-section.

54. If $THIND < 4$, go to 69

If $THIND \geq 4$, go to 55

Liquid engine simulation does not end section until after thrust decay simulation. Check for end of hold-down simulation.

55. If FLGHD = .FALSE., go to 60

If FLGHD ≠ .FALSE., go to 56

Converged on lift-off rather than end-of-section. Turn off flag for hold-down and restore section ending data.

56. If $TIME > TBLDUP$, FLGHD = .FALSE.

If $TIME \leq TBLDUP$, FLGHD = FLGHD

57. PARAM = IPARAM

58. VE = VES

```
59.  Go to 71

Is this the start or end of shutdown ?

60.  If FLGSD = .TRUE.,  go to 69
     If FLGSD ≠ .TRUE.,  go to 61
61.  If FLCNV = .FALSE.,  go to 69
     If FLCNV ≠ .FALSE.,  go to 62

This is start of shutdown.

62.  FLCNV = .FALSE.
63.  FLGSD = .TRUE.
64.  TEND = TIME + DTDCY
65.  PARAM = LOC(TIME)
66.  VE = TEND
67.  If DTDCY = 0.0,  go to 69
     If DTDCY ≠ 0.0,  go to 68
68.  Go to 71
69.  FLEOS = .TRUE.
70.  FLGSD = .FALSE.
71.  FLDSC = 1
72.  Go to 94

Check for print time.
```

```
73.  If $FLCNV = .TRUE.$, go to 94
     If $FLCNV \neq .TRUE.$, go to 74
74.  $DELTP = TSUBP - TIME$
75.  If $|DELTP| > P001$, go to 78
     If $|DELTP| \leq P001$, go to 76
76.  $FLTP = .TRUE.$
77.  Go to 79
78.  If $DELTP < DT$, $DT = DELTP$
     If $DELTP \geq DT$, $DT = DT$
79.  If $PARAM = IND$, go to 94
     If $PARAM \neq IND$, go to 80
```

Calculate integration interval to converge on end-of-section.

```
80.  If $SLPIND = 0.0$, go to 82
     If $SPLIND \neq 0.0$, go to 81
```

See if ending parameter has desired slope.

```
81.  If $(SLOPE)(SLPIND) < 0.0$, go to 93
     If $(SLOPE)(SLPIND) \geq 0.0$, go to 82
82.  $DELT = (DELPE)/(SLOPE)$
83.  If $DELT = 0.0$, go to 54
     If $DELT \neq 0.0$, go to 84
```

Compare convergence interval with integration interval.

84. If DELT $\geq$ DT, go to 93

If DELT $<$ DT, go to 85

Check for negative convergence interval.

85. If DELT $>$ 0.0, go to 89

If DELT $\leq$ 0.0, go to 86

Do not integrate back beyond start of section.

86. If TIME + DELT − TSUBS $\geq$ 0.0, go to 89

If TIME + DELT − TSUBS $<$ 0.0, go to 87

If prefix slope indicator is 4, continue until section end parameter function slope gives convergence within the section.

87. If PRE = 4, go to 93

If PRE $\neq$ 4, go to 88

Integrate back to start of section.

88. DELT = TSUBS − TIME

Do not integrate back more than 2 integration steps.

89. If DELT + DTSBS + DTSBS $<$ 0.0, go to 93

If DELT + DTSBS + DTSBS $\geq$ 0.0, go to 90

90. DT = DELT

Increment convergence iteration counter.

91. ITER = ITER + 1

Check for max iterations on convergence

92. If ITER $>$ 30, CALL SETERR(5H ENDOS)

If ITER $\leq$ 30, go to 93

Turn off flag to print if integration interval is negative.

93. If DT $<$ 0.0, FLTP = .FALSE.

If DT $\geq$ 0.0, FLTP = FLTP

Save values of parameter and exit.

94. PELST = PE

If volume convergence flag is on, premature shutdown is indicated.

95. If FLCNV = .TRUE., CALL SETERR(IND)

If FLCNV $\neq$ .TRUE., go to 96

96. FLCNV = .FALSE.

97. If FLGSD = .TRUE., DT = AMIN1(DT, TEND - TIME)

If FLGSD $\neq$ .TRUE., continue

Return

No change in parameter with time.

98. If DTLST = DTSBS, go to 73

If DTLST $\neq$ DTSBS, go to 99

99. Go to 54

SUBROUTINE ENDOS NOMENCLATURE

Symbol	Description	Units	Reference
ADD	Address portion of PARAM giving location of section ending parameter	—	INTEGER, EQUIV, 36
ARRAY	Array containing address, tag, decrement and prefix of PARAM	—	DIM, EQUIV, 30
DEC	Decrement portion of PARAM giving location of auxiliary routine	—	INTEGER, EQUIV, 31
DELPE	Difference between desired value and current value of ending parameter	—	40, 53, 82
DELPN	Difference between current value and last value of ending parameter	—	41, 42, 43
DELT	Miscellaneous calculated time intervals	sec	16, 17, 25, 26, 83, 84, 85, 86, 88, 89, 90
DELTP	Interval to time-to-print	sec	75, 78
DT	Integration interval	sec	/GILL/, 17, 26, 78, 84, 90, 93
DTBUP	Time interval for thrust build up	sec	/ENGINW/, 9
DTDCY	Time interval for thrust decay	sec	/ENGINW/, 64, 67
DTLST	Last integration interval	sec	/DATA/, 43, 98

SUBROUTINE ENDOS NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DTSBS	Nominal section integration interval	sec	/WORKD/, 89, 98
EPSILN	Convergence tolerance on ending parameter	—	DATA, 51, 53
EPVOL	Convergence tolerance on shutdown volumes	—	/ENGINW/, 20, 21
FLCNV	Volume convergence flag	—	LOGICAL, DATA, 20, 21, 22, 28, 61, 62, 73, 95, 96
FLDSC	Discontinuity flag	—	INTEGER, /WFLAG/, 11, 37, 71
FLEOS	End of section flag	—	LOGICAL, /WFLAG/, 69
FLGHD	Hold-down flag	—	LOGICAL, /ENGINW/, 12, 15, 55, 56
FLGSD	Shutdown flag	—	LOGICAL, /ENGINW/, 19, 60, 63, 70, 97
FLTP	Time-to-print flag	—	LOGICAL, /WFLAG/, 76, 93
IND	Indicator to end section on propellant depletion	—	DATA, 27, 79, 95
IPARAM	Saved value of PARAM during hold-down	—	5, 57
ITER	Iteration counter to converge on end of section	—	38, 91, 92
P001	.001	—	DATA, 10, 75

SUBROUTINE ENDOS NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PARAM	Parameter to end section	—	INTEGER, /WORKD/, 5, 7, 27, 30, 57, 65, 79
PARAML	Parameter for lift-off	—	INTEGER, /ENGINW/, 2, 7, 12
PE	Current value of section ending parameter	—	36, 40, 41, 51, 94
PELST	Last value of section ending parameter	—	41, 94
PRE	Prefix portion of PARAM giving slope indicator	—	INTEGER, EQUIV, 44, 45, 87
RHOF	Fuel density	lb/ft^3	/ENGINO/, 24
RHOO	Oxidizer density	lb/ft^3	/ENGINO/, 23
SLOPE	Slope sign of section ending parameter	—	43, 47, 49, 81, 82
SLPIND	Desired slope sign of section ending parameter	—	44, 45, 46, 80, 81
TAG	Tag portion of PARAM giving conversion tag	—	INTEGER, EQUIV, 34, 35
TAGS	Array of conversion factors	—	/CONST/, 34
TBLDUP	Time when thrust build-up is complete	sec	/ENGINW/, 10, 14, 16
TEND	Time when thrust decay is complete	sec	/ENGINW/, 64, 66, 97
THIND	Thrust indicator	—	INTEGER, /WORKD/, 1, 54

SUBROUTINE ENDOS NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TI	Initial time	sec	/INPUT/, 3
TIME	integrated time	sec	/GILL/, 3, 10, 14, 16, 56, 64, 65, 74, 86, 88
TSUBP	Time to print	sec	/DATA/, 74
TSUBS	Time at start of section	sec	/SAVE/, 86, 88
TYSC	Section tally	—	INTEGER, /WFLAG/, 15
VE	Value to end section	—	/WORKD/, 6, 8, 40, 50, 53, 58, 66
VES	Saved value of VE during hold-down simulation	—	6, 58
VF	Fuel volume	ft^3	/ENGINO/, 21, 25
VFCOFF	Fuel shutdown volume	ft^3	/ENGINW/, 21, 25
VFDOT	Fuel flow rate	ft^3/sec	24, 25
VO	Oxidizer volume	ft^3	/ENGINO/, 20, 25
VOCOFF	Oxidizer shutdown volume	ft^3	/ENGINW/, 20, 25
VODOT	Oxidizer flow rate	ft^3/sec	23, 25
VPARL	Value of lift-off parameter	—	/ENGINW/, 8
WFDOT	Fuel flow rate	lb/sec	/GILL/, 24
WODOT	Oxidizer flow rate	lb/sec	/GILL/, 23
Z	Temporary storage	—	32, 33, 34, 35, 36

```
$IBFTC ENDØS   FULIST,REF
      SUBRØUTINE ENDØS
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $              TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL  INFIN
      CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,
     $              DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $              SECTS(20,12),STAGS(20,6),HEAD(11,15)
      REAL  LATI,LØNGI
      INTEGER  TARGET
      INTEGER  FLTR,FLSC,FLPRT
      CØMMØN/WFLAG/FLEØF,FLEØS,FLTP,FLDSC,TYSC,FLTRW
      LØGICAL FLEØF,FLEØS,FLTP
      INTEGER  FLDSC,TYSC,FLTRW
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $              PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $              PADSK,VESK,PADRT1,PADRT2,
     $             SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $              ASUBTS,ISP,KSUBF,KSUBV,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL  ISP,KSUBF,KSUBV,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $              AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $              WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $             W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $             DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $             WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $              PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $              GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $              QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $              CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $              ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN /ENGINØ/ PØ,PF,TØ,TF,CSTR,RHØØ,RHØF,PTØ,PTF,VØ,VF,HØ,HF,
     $             WØCDT,WFCDT,MRC,PC,WØGDT,WFGDT,CVTH,CFTH,CSUBF,FCHMBR,
     $             FENGIN,WØEDT,WFEDT,MRE,WEDØT,ISPE,WØPDT,WFPDT,
     $             FS,WØSDT,WFSDT,MRS,WSDØT,ISPS
      REAL  MRC,MRE,MRS,ISPE,ISPS
      CØMMØN/ENGINW/PCBBR,WØCBB,WFCBB,WØGBB,WFGBB,
     $              PCBAR,WØCBR,WFCBR,WØGBR,WFGBR,
     $              PØNØM,PFNØM,TØNØM,TFNØM,FØPNØM,ETANØM,
     $              PIND,TREF,DREF,FPTPA,PTØG,PTFG,TSØ1,PTØ1,TSØ2,PTØ2,
     $              C1Ø,C2Ø,TSF1,PTF1,TSF2,PTF2,C1F,C2F,
     $              LØ,RØ,UHØ,LHØ,LF,RF,UHF,LHF(5),
     $              NØFE,FTURB,FØPT,KURF,DTBUP,VØCØFF,VFCØFF,EPVØL,
     $              DTDCY,WMSDT,WMØSD,WMFSD,WMØED,WMFED,RESØ,RESF,
     $              ASUBT,EPSI,ETACF,ETACS,TØI,CTØ,TFI,CTF,
     $              VØBB,VFBB,HØCØL,HFCØL,PARAML,VPARL,
```

FIGURE 21. SYMBOLIC LISTING OF SUBROUTINE ENDOS

```
     $             WKPC(6),WKWØC(6),WKWFC(6),WKWØG(6),WKWFG(6),
     $             TBLDUP,TEND,FLGHD,FLGSD,LSTAGE,CSTRN
      LØGICAL  FLGHD,FLGSD
      REAL  LØ,LF,NØFE,KURF
      INTEGER  PIND,FPTPA,UHØ,UHF
      DIMENSIØN  ARRAY(4)
      EQUIVALENCE  (ARRAY(1),ADD),(ARRAY(2),TAG),(ARRAY(3),DEC),
     $             (ARRAY(4),PRE)
      INTEGER  ADD,TAG,DEC,PRE
      INTEGER  PARAM,PARAML
      LØGICAL  FLCNV
      DATA  EPSILN,P001/.00001,.001/,IND/4HPØØP/,FLCNV/.FALSE./
C
      IF (THIND .LT. 4)  GØ TØ 15
      IF (PARAML .EQ. 0)  GØ TØ 5
      IF (TIME .NE. TI)  GØ TØ 5
C        SET UP FØR CØNVERGENCE ØN LIFT-ØFF PARAMETER
      FLGHD = .TRUE.
      IPARAM = PARAM
      VES = VE
      PARAM = PARAML
      VE = VPARL
C        CØNVERGE ØN BUILD-UP TIME
    5 IF (DTBUP .EQ. 0.0)  GØ TØ 10
      IF (ABS(TIME - TBLDUP) .GT. P001)  GØ TØ 6
      IF (FLDSC .EQ. 0)  FLDSC = 1
      IF (PARAML .NE. PARAM)  FLGHD = .FALSE.
      GØ TØ 15
    6 IF (TIME .GT. TBLDUP)  GØ TØ 10
      IF (TYSC .EQ. 1)  FLGHD = .TRUE.
      DELT = TBLDUP - TIME
      IF (DELT .LT. DT)  DT = DELT
      GØ TØ 15
C        CØNVERGE ØN SHUT-DØWN VØLUMES
   10 IF (FLGSD)  GØ TØ 15
      IF (ABS(VØ - VØCØFF) .LE. EPVØL)  FLCNV = .TRUE.
      IF (ABS(VF - VFCØFF) .LE. EPVØL)  FLCNV = .TRUE.
      IF (FLCNV)  GØ TØ 15
      VØDØT = WØDØT/RHØØ
      VFDØT = WFDØT/RHØF
      DELT = AMIN1((VØCØFF-VØ)/VØDØT,(VFCØFF-VF)/VFDØT)
      IF (DELT .LT. DT)  DT = DELT
C
   15 IF (PARAM .NE. IND)  GØ TØ 16
      IF (FLCNV)  GØ TØ 61
      GØ TØ 70
   16 CALL UNPACK(PARAM,ARRAY)
      CALL XEC(DEC)
      CALL CHKERR(Z)
      IF (Z .NE. 0.0)  GØ TØ 62
```

FIGURE 21. SYMBOLIC LISTING OF SUBROUTINE ENDOS (cont.)

```
      Z = TAGS(TAG)
      IF (TAG .EQ. 0)  Z = 1.0
      PE = Z * CLA(ADD)
      IF (FLDSC .GE. 0)  GO TO 20
      ITER = 0
      GO TO 900
C
   20 DELPE = VE - PE
      DELPN = PE - PELST
      IF (DELPN .EQ. 0.0)  GO TO 800
      SLOPE = DELPN/DTLST
      SLPIND = FLOAT(ISIGN(1,4-PRE))
      IF (PRE .EQ. 0 .OR. PRE .EQ. 4)  SLPIND = 0.0
      IF (SLPIND)  30,50,40
C         SLPIND .LT. 0.0
   30 IF (SLOPE .GT. 0.0)  GO TO 70
      GO TO 50
C         SLPIND .GT. 0.0
   40 IF (SLOPE .LT. 0.0)  GO TO 70
C         CHECK FOR CONVERGENCE
   50 IF (VE .NE. 0.0)  GO TO 60
      IF (ABS(PE) .LE. EPSILN)  GO TO 61
      GO TO 70
   60 IF (ABS(DELPE/VE) .GT. EPSILN)  GO TO 70
C         CONVERGED ON END OF SECTION
   61 IF (THIND .LT. 4)  GO TO 62
      IF (.NOT. FLGHD)  GO TO 64
      IF (TIME .GT. TBLDUP)  FLGHD = .FALSE.
      PARAM = IPARAM
      VE = VES
      GO TO 63
   64 IF (FLGSD)  GO TO 62
      IF (.NOT. FLCNV)  GO TO 62
      FLCNV = .FALSE.
      FLGSD = .TRUE.
      TEND = TIME + DTDCY
      PARAM = LOC(TIME)
      VE = TEND
      IF (DTDCY .EQ. 0.0)  GO TO 62
      GO TO 63
   62 FLEOS = .TRUE.
      FLGSD = .FALSE.
   63 FLDSC = 1
      GO TO 900
C         CHECK FOR PRINT TIME
   70 IF (FLCNV)  GO TO 900
      DELTP = TSUBP - TIME
      IF (ABS(DELTP) .GT. P001)  GO TO 80
   71 FLTP = .TRUE.
      GO TO 85
```

FIGURE **21**. SYMBOLIC LISTING OF SUBROUTINE ENDOS (cont.)

```
C
C   80 IF (DELTP .LT. DT)  DT = DELTP
    85 IF (PARAM .EQ. IND)  GØ TØ 900
C
    90 IF (SLPIND .EQ. 0.0)  GØ TØ 100
       IF (SLØPE * SLPIND .LT. 0.0)  GØ TØ 120
   100 DELT = DELPE/SLØPE
       IF (DELT .EQ. 0.0)  GØ TØ 61
       IF (DELT .GE. DT)  GØ TØ 120
       IF (DELT .GT. 0.0)  GØ TØ 110
       IF ((TIME + DELT - TSUBS) .GE. 0.0)  GØ TØ 110
       IF (PRE .EQ. 4)  GØ TØ 120
       DELT = TSUBS - TIME
   110 IF ((DELT + DTSBS + DTSBS)  .LT. 0.0) GØ TØ 120
       DT = DELT
       ITER = ITER + 1
       IF (ITER .GT. 30)  CALL SETERR(5HENDØS)
   120 IF (DT .LT. 0.0)  FLTP = .FALSE.
C
   900 PELST = PE
       IF (FLCNV)  CALL SETERR(IND)
       FLCNV = .FALSE.
       IF (FLGSD)  DT = AMIN1(DT,TEND-TIME)
       RETURN
C            NØ CHANGE
   800 IF (DTLST .EQ. DTSBS)  GØ TØ 70
       GØ TØ 61
C
       END
```

FIGURE 21. SYMBOLIC LISTING OF SUBROUTINE ENDOS (cont.)

XVIII. SUBROUTINE NEWTON

This subroutine computes the derivatives used in the integration of the trajectory, plus various auxiliary parameters. Atmospheric properties, propulsion system performance, missile orientation, and aerodynamic forces are computed internal to Subroutine NEWTON and/or through CALLS to other subroutines.

CALLING SEQUENCE

The calling sequence is:

CALL NEWTON

Subroutine NEWTON uses COMMON blocks TAPES, CONST, INPUT, WFLAG, WORKD, SAVE, GILL, DATA, COMP, TABLES, ENGINW, and DUMPY.

SOLUTION METHOD

Calculate relative flight, stage, and section times.

1. TFLIT = TIME - TI
2. TSTAG = TIME - TSTG
3. TSECT = TIME - TSUBS

Convert vehicle weight to mass.

4. MASS = (W)/(GSUB1)

Calculate components of inertial position vector.

5. RXIN = ROX + XIN

6. RYIN = ROY + YIN

7. RZIN = ROZ + ZIN

Calculate magnitude of inertial position vector.

8. RVCTR = $(RXIN^2 + RYIN^2 + RZIN^2)^{.5}$

Convert inertial position and velocity to polar coordinates.

9. CALL G2OB (RXIN, RYIN, RZIN, XDIN, YDIN, ZDIN,
RSUBO, FLAT, PHI, THETA, HATM,
VSUBI, GAMIN, AZIN, 1)

Calculate potential and kenetic energy.

10. ESUBP = $(RSUB0^2)$ (GSUB1)/(RVCTR)

11. ESUBK = $(VSUBI^2)$/(2.0)

Calculate energy ratio.

12. ERATIO = (ESUBK)/(ESUBP)

Calculate relative velocity components, GAMMA and azimuth.

13. VRX = XDIN + (OMEGA)(RYIN)

14. VRY = YDIN - (OMEGA)(RXIN)

15. VRZ = ZDIN

Correct relative velocity components roundoff error during first section vertical rise.

```
16.  If TYSC = 1 and ATIND = 1, CALL G1OB (0, 0, PHI, THETA, 0, $(VRX^2 + VRY^2 + VRZ^2)^{.5}$, PIO2, AZIR, U, U, U, VRX, VRY, VRZ, 1)
     Otherwise, go to 17
17.  CALL G2OB (RXIN, RYIN, RZIN, VRX, VRY, VRZ, 0, 0, U, U, U, VSUBR, GAMMA, AZ, 1)
18.  If $||GAMMA| - PIO2| < .0000002$, AZ = AZIR
     If $||GAMMA| - PIO2| \geq .0000002$, AZ = AZ
Check for hold-down.
19.  If $XDIND + YDIND + ZDIND \neq 0$, go to 25
     If $XDIND + YDIND + ZDIND = 0$, go to 20
If hold-down, use input initial conditions
20.  HATM   = HI
21.  VSUBR  = VI
22.  GAMMA  = PIO2 - (BETAI)/(DPRAD)
23.  AZ     = AZIR
24.  Go to 28
25.  If $VSUBR \neq 0$, go to 28
     If VSUBR = 0, go to 26
Define GAMMA and azimuth for zero velocity.
26.  GAMMA  = 0
27.  AZ     = AZIR
```

```
28.   Continue

Compute angle between velocity vector and local vertical.

29.   BETA = PIO2 - GAMMA

Check for trajectory error.

30.   If HATM > HMIN, go to 33

      If HATM ≤ HMIN, go to 31

31.   CALL SETERR (2H - H)

32.   Go to 39

33.   If W > 0, go to 36

      If W ≤ 0, go to 34

34.   CALL SETERR (2H - W)

35.   Go to 39

36.   If W > WPL, go to 42

      If W ≤ WPL, go to 37

37.   CALL SETERR (5HW(PL))

38.   Go to 39

Dump desired data.

39.   CALL DUMPS (DUMP9, 10, OTAPE)

Abort integration.

40.   KPASS = 4

41.   Return
```

```
42.  Continue
```

Calculate latitude, longitude and surface range.

```
43.  LAT     = ATAN ((Tan (PHI))/(1.0 - FLAT)$^2$) (DPRAD)

44.  THETAP  = THETAO + (OMEGA) (TIME - TI)

45.  LONG    = AMOD (LONGI - (THETA - THETAP) (DPRAD), 360)

46.  If LONG < 0, LONG = LONG + 360
     If LONG ≥ 0, LONG = LONG

47.  DTHETA = THETA - THETAP

48.  If DTHETA < 0, DTHETA = DTHETA + (2) (PI)
     If DTHETA ≥ 0, DTHETA = DTHETA

49.  If DTHETA > PI and TIME < 50, DTHETA = 0
     Otherwise, DTHETA = DTHETA

50.  If AZI > 180, DTHETA = (2) (PI) - DTHETA
     If AZI ≤ 180, DTHETA = DTHETA

51.  CPSI    = (SPHIO) (Sin (PHI))
               + (CPHIO) (Cos (PHI)) (Cos (DTHETA))

52.  If CPSI < .99998, go to 58
     If CPSI ≥ .99998, go to 53

53.  ANGLE1 = PHI - PHIO

54.  ANGLE2 = DTHETA
```

55. If $|\text{ANGLE2}| > \text{PI}$, $\text{ANGLE2} = \text{SIGN}((2)(\text{PI}) - |\text{ANGLE2}|, \text{ANGLE2})$

Otherwise, go to 56

56. $\text{PSI} = (\text{ANGLE1}^2 + (\text{ANGLE2}^2)(\text{Cos}(\text{PHIO}))^2)^{.5}$

57. Go to 59

58. PSI = ACOS (CPSI)

59. RCIRC = PSI

60. If $\text{DTHETA} > \text{PI}$, RCIRC = (2)(PI) - PSI

If $\text{DTHETA} \leq \text{PI}$, RCIRC = RCIRC

61. RANGE = (RCIRC)(RSUBO)/(6076.1033)

Calculate atmospheric parameters.

62. If $\text{HATM} > \text{HMAX}$, go to 65

If $\text{HATM} \leq \text{HMAX}$, go to 63

63. CALL STDATM (HATM, RATM, CATM, TATM, PATM)

64. Go to 69

Above atmosphere, set density, temperature, pressure, and speed of sound.

65. RATM = 0

66. TATM = 0

67. PATM = 0

68. CATM = CMAX

69. Continue

Compute Mach number and dynamic pressure.

```
70.  MACH    = (VSUBR)/(CATM)

71.  QATM    = (RATM)(VSUBR$^{2}$)/(2)
```

Calculate GAMMA and azimuth relative to launch site.

```
72.  CALL G2OB (ROX, ROY, ROZ, VRX, VRY, VRZ, 0, 0, U, U,
                 U, U, GAMRF, AZREF, 1)

73.  If |GAMRF| - PIO2 < .0000002, AZREF = AZIR

     If |GAMRF| - PIO2 ≥ .0000002, AZREF = AZREF
```

Check for hold-down.

```
74.  If XDIND + YDIND + ZDIND ≠ 0,  go to 78

     If XDIND + YDIND + ZDIND = 0,  go to 75

75.  GAMRF   = GAMMA

76.  AZREF   = AZ

77.  Go to 81

78.  If VSUBR ≠ 0,  go to 81

     If VSUBR = 0,  go to 79
```

Define GAMRF and AZREF if velocity is zero.

```
79.  GAMRF   = GAMMA

80.  AZREF   = AZ

81.  Continue
```

Calculate missile attitude and angle of attack.

82. CALL GUIDE

83. ALPHAP = PIO2 - GAMRF - PIT

84. ALPHAY = AZIR - AZREF + YAW

85. ALPHA = $(\mathrm{ALPHAP}^2 + \mathrm{ALPHAY}^2)^{.5}$

86. ALPHAD = (ALPHA)(DPRAD)

Calculate aerodynamic forces.

87. CALL NTERP (TMACH(1, STIND), TCSBN(1, STIND), 30, MACH, CSUBN, U, 2, KMACH(STIND), L)

88. CALL NTERP (TMACH(1, STIND), TCSBA(1, STIND), 30, MACH, CSUBA, U, 2, KMACH(STIND), L)

89. CSUBN = (KSUBN)(CSUBN)(ALPHAD)

90. CSUBA = (KSUBA)(CSUBA)

91. SALPHA = Sin(ALPHA)

92. CALPHA = Cos(ALPHA)

93. CSUBL = (CSUBN)(CALPHA) - (CSUBA)(SALPHA)

94. CSUBD = (CSUBN)(SALPHA) + (CSUBA)(CALPHA)

95. LIFT = (CSUBL)(QATM)(SREF)

96. DRAG = (CSUBD)(QATM)(SREF)

Calculate lift and drag components.

97. CALL G1OB (0, 0, PHI, THETA, 0, LIFT, GAMMA + PIO2, AZ, U, U, U, LIFTX, LIFTY, LIFTZ, 1)

98. DRAGX = - (DRAG)(VRX)/(VSUBR)

99. DRAGY = - (DRAG)(VRY)/(VSUBR)

100. DRAGZ = - (DRAG)(VRZ)/(VSUBR)

Calculate gravity components.

101. GRAV = (GSUB0)((RSUBO)/(RVCTR))^2

102. CALL G1OB (0, 0, PHI, THETA, 0, GRAV - PIO2, 0, U, U, U, GRAVX, GRAVY, GRAVZ, 1)

Calculate thrust and flow rate.

103. CALL ENGINE

104. If FLDSC < 0, CALL ENGINE

If FLDSC ≥ 0, go to 105

105. If TIME = TI, CALL ENGINE

If TIME ≠ TI, go to 106

Calculate thrust components.

106. CALL G1OB (0, 0, PHIO, THETAO, 0, THRST, PIO2 - PIT, AZIR + YAW, U, U, U, THRSTX, THRSTY, THRSTZ, 1)

Sum forces for inertial acceleration.

107. FXIN = THRSTX + DRAGX + LIFTX

108. FYIN = THRSTY + DRAGY + LIFTY

109. FZIN = THRSTZ + DRAGZ + LIFTZ

110. XDIND = (FXIN)/(MASS) + GRAVX

111. YDIND = (FYIN)/(MASS) + GRAVY

112. ZDIND = (FZIN)/(MASS) + GRAVZ

```
113.  XIND   = XDIN

114.  YIND   = YDIN

115.  ZIND   = ZDIN

116.  ACCEL  = (THRST - DRAG)/(W)

117.  CALL DUMPS (DUMP8, 10, OTAPE)

118.  If FLGHD .FALSE. , return

      If FLGHD .TRUE.  , go to 119
```

If hold-down, inertial acceleration is zero. Inertial velocity is rotational velocity.

```
119.  XDIND  = 0

120.  YDIND  = 0

121.  ZDIND  = 0

122.  XDIN   = (-OMEGA)(RYIN)

123.  YDIN   = (OMEGA)(RXIN)

124.  ZDIN   = 0

125.  XIND   = XDIN

126.  YIND   = YDIN

127.  ZIND   = ZDIN

128.  ACCEL  = 0

129.  Return
```

SUBROUTINE NEWTON NOMENCLATURE

Symbol	Description	Units	Reference
ACCEL	Thrust-to-weight ratio	g's	/GILL/, 116, 128
ALPHAD	Angle of attack	deg	/DATA/, 86, 89
ALPHAP	Pitch plane component of angle of attack	rad	/DATA/, 83, 85
ALPHA	Angle of attack	rad	/DATA/, 85, 86, 91, 92
ALPHAY	Yaw plane component of angle of attack	rad	/DATA/, 84, 85
ANGLE1	Difference between PHI and PHIO	rad	53, 56
ANGLE2	Difference between THETA and THETAP	rad	54, 55, 56
ATIND	Attitude control indicator	—	/WORKD/, INTEGER, 16
AZIN	Azimuth of the inertial velocity vector	rad	/DATA/, 9
AZIR	Launch azimuth	rad	/SAVE/, 16, 18, 23, 27, 73, 84, 106
AZI	Launch azimuth	deg	/INPUT/, 50
AZREF	Azimuth of relative velocity vector measured at inertial launch point	rad	/DATA/, 72, 73, 76, 80, 84
AZ	Local azimuth of inertial velocity vector	rad	/DATA/, 17, 18, 23, 27, 76, 80, 97
BETAI	Initial BETA	deg	/INPUT/, 22

SUBROUTINE NEWTON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
BETA	Angle between velocity vector and local vertical	deg	/DATA/, 29
CALPHA	Cosine of ALPHA	—	92, 93, 94
CATM	Local speed of sound	ft/sec	/DATA/, 68
CMAX	Speed of sound above 300,000 feet	ft/sec	/CONST/ , 68, 70
CPHIO	Cosine PHIO	—	/SAVE/, 51
CPSI	Cosine PSI	—	51, 52, 58
CSUBA	Axial force coefficient	—	/DATA/, 88, 90, 93, 94
CSUBD	Drag coefficient	—	/DATA/, 94, 96
CSUBL	Lift coefficient	—	/DATA/, 93, 95
CSUBN	Normal force coefficient	—	/DATA/, 87, 89, 93, 94
DPRAD	Conversion factor	deg/rad	/CONST/, 22, 43, 45, 86
DRAG	Aerodynamic drag force	lb	/DATA/, 96, 98, 99, 100, 116
DRAGX	X-component of the drag force	lb	/COMP/, 98
DRAGY	Y-component of the drag force	lb	/COMP/, 99
DRAGZ	Z-component of the drag force	lb	/COMP/, 100

SUBROUTINE NEWTON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DTHETA	Clockwise difference between THETA and THETAP	rad	47, 48, 49, 50, 51, 54, 60
DUMP8	Dump list	—	/DUMPY/, 117
DUMP9	Dump list	—	/DUMPY/, 39
ERATIO	Ratio of kinetic to potential energy	—	/DATA/, 12
ESUBK	Kinetic energy	ft^2/sec^2	/DATA/, 11, 12
ESUBP	Potential energy	ft^2/sec^2	/DATA/, 10, 12
FLAT	Flattening of the spheroid	—	/CONST/, 9, 43
FLDSC	Discontinuity flag	—	/WFLAG/, INTEGER, 104
FLGHD	Hold-down flag	—	/ENGINW/, LOGICAL, 118
FXIN	X-component of the sum of the forces	lb	107, 110
FYIN	Y-component of the sum of the forces	lb	108, 111
FZIN	Z-component of the sum of the forces	lb	109, 112
GAMIN	GAMMA of inertial velocity vector	rad	/DATA/, 9
GAMMA	Angle of velocity vector with local horizontal	rad	/DATA/, 17, 18, 22, 26, 29, 75, 79, 97

SUBROUTINE NEWTON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
GAMRF	Angle of velocity vector with local horizontal relative to launch site horizontal	rad	/DATA/, 72, 73, 75, 79, 83
GRAV	Gravitational force	lb	/DATA/, 101, 102
GRAVX	X-component of gravitational force	lb	/COMP/, 102, 110
GRAVY	Y-component of gravitational force	lb	/COMP/, 102, 111
GRAVZ	Z-component of gravitational force	lb	/COMP/, 102, 112
GSUB0	Surface gravitational force	ft/sec^2	/CONST/, EQUIV, 101
GSUB1	Weight to mass conversion	ft/sec^2	/CONST/, 4, 10
HATM	Height above reference spheroid	ft	/DATA/, 9, 20, 30, 62
H	Altitude above spheroid surface	ft	31
HI	Initial height	ft	/INPUT/, 20
HMAX	Upper limit of the atmosphere	ft	/CONST/, 62
HMIN	Crash altitude	ft	/CONST/, 30
KMACH	Counter for interpolation routine	—	87, 88
KPASS	Integration counter for the number of passes into the derivative routine	—	/GILL/, 40
KSUBA	Axial force multiplier	—	/WORKD/, 90

SUBROUTINE NEWTON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
KSUBN	Normal force multiplier	—	/WORKD/, 89
LAT	Geodetic latitude	deg	/DATA/, REAL, 43
LIFT	Aerodynamic lift force	lb	/DATA/, REAL, 95, 97
LIFTX	X-component of lift	lb	/COMP/, REAL, 97, 107
LIFTY	Y-component of lift	lb	/COMP/, REAL, 97, 108
LIFTZ	Z-component of lift	lb	/COMP/, REAL, 97, 109
LONGI	Initial longitude	deg	/INPUT/, REAL, 45
LONG	Longitude west of Greenwich	deg	/DATA/, REAL, 45, 46
L	Error indicator set by interpolation routine	—	87, 88
MACH	Mach number	—	/DATA/, REAL, 70, 87, 88
MASS	Vehicle mass	slugs	REAL, 4, 110, 111, 112
OMEGA	Spin rate of spheroid	rad/sec	/CONST/, 13, 14, 44, 122, 123
OTAPE	Output tape designator	—	/TAPES/, INTEGER, 39, 117
PATM	Atmospheric pressure	lb/in^2	/DATA/, 67
PHIO	Geocentric latitude at initial time	rad	/SAVE/, 53, 56, 106
PHI	Geocentric latitude	rad	/DATA/, 9, 16, 43, 51, 53, 97, 102
PIO2	$(\pi)/(2)$	—	/CONST/, 16, 18, 22, 29, 73, 83, 97, 102, 106

SUBROUTINE NEWTON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PI	3.14159269	—	/CONST/, 48, 49, 50, 55, 60
PIT	Pitch angle	rad	/DATA/, 83, 106
PSI	Range angle	rad	56, 58, 59, 60
QATM	Dynamic pressure	lb/ft^2	/DATA/, 71, 95, 96
RANGE	Surface range from launch to vehicle position	n.m.	/DATA/, 61
RATM	Atmospheric density	$slugs/ft^3$	/DATA/, 65, 71
RCIRC	Total surface range angle	rad	59, 60, 61
ROX	X-component of initial position	ft	/SAVE/, 5, 72
ROY	Y-component of initial position	ft	/SAVE/, 6, 72
ROZ	Z-component of initial position	ft	/SAVE/, 7, 72
RSUB0	Equatorial radius of spheroid	ft	/CONST/, EQUIV, 10
RSUBO	Radius of the earth	ft	EQUIV, 9, 61, 101
RVCTR	Magnitude of the position vector	ft	/DATA/, 8, 10, 101
RXIN	X-component of position vector	ft	/DATA/, 5, 8, 9, 14, 17 123
RYIN	Y-component of position vector	ft	/DATA/, 6, 8, 9, 13, 17 122

SUBROUTINE NEWTON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RZIN	Z-component of position vector	ft	/DATA/, 7, 8, 9, 17
SALPHA	Sine of ALPHA	—	91, 93, 94
SPHIO	Sine of PHIO	—	/SAVE/, 51
SREF	Stage reference area	ft^2	/WORKD/, 95, 96
STIND	Stage indicator	—	/WORKD/, INTEGER, 87, 88
TATM	Atmospheric temperature	°R	/DATA/, 66
TCSBA	Array of Mach number - axial force coefficient tables	—	/TABLES/, 88
TCSBN	Array of Mach number - normal force coefficient tables	—	/TABLES/, 87
TFLIT	Flight time	sec	/DATA/, 1
THETAO	Initial sidereal time at launch	rad	/SAVE/, 44, 106
THETAP	Instantaneous sidereal time of the rotating launch point	rad	44, 45, 47
THETA	Local sidereal time	rad	/DATA/, 9, 16, 45, 47, 97, 102
THRST	Engine thrust	lb	/DATA/, 106, 116
THRSTX	X-component of thrust	lb	/COMP/, 106, 107
THRSTY	Y-component of thrust	lb	/COMP/, 106, 108

SUBROUTINE NEWTON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
THRSTZ	Z-component of thrust	lb	/COMP/, 106, 109
TIME	Instantaneous time	sec	/GILL/, 1, 2, 3, 44, 49 105
TI	Initial time	sec	/INPUT/, 1, 44, 105
TMACH	Array of Mach number - force coefficient tables	—	87, 88
TSECT	Section time	sec	/DATA/, 3
TSTAG	Stage time	sec	/DATA/, 2
TSTG	Initial stage time	sec	/SAVE/, 2
TSUBS	Initial section time	sec	/SAVE/, 3
TYSC	Section tally	—	/WFLAG/, 16
U	Dummy variable	—	16, 17, 87, 88, 97, 102, 106
VI	Initial velocity	ft/sec	/INPUT/, 21
VRX	X-component of relative velocity	ft/sec	/COMP/, 13, 16, 17, 72, 98
VRY	Y-component of relative velocity	ft/sec	/COMP/, 14, 16, 17, 72, 99
VRZ	Z-component of relative velocity	ft/sec	/COMP/, 15, 16, 17, 72, 100
VSUBI	Initial velocity	ft/sec	/DATA/, 9, 11
VSUBR	Relative velocity	ft/sec	/DATA/, 17, 21, 25, 70, 71, 78, 98, 99, 100

SUBROUTINE NEWTON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WPL	Payload weight	lb	/INPUT/, 36, 37
W	Stage or vehicle weight	lb	/GILL/, 4, 33, 34, 36, 116
XDIND	X-component of inertial acceleration	ft/sec^2	/GILL/, 19, 74, 110, 119
XDIN	X-component of inertial velocity (derivative of position)	ft/sec	/GILL/, 9, 13, 113, 122, 125
XIND	X-component of inertial velocity (integral of acceleration)	ft/sec	/GILL/, 113, 125
XIN	Inertial X-component of the difference between the inertial position and launch point	ft	/GILL/, 5
YAW	Yaw angle	rad	/DATA/, 84, 106
YDIND	Y-component of inertial acceleration	ft/sec^2	/GILL/, 19, 74, 111, 120
YDIN	Y-component of inertial velocity (derivative of position)	ft/sec	/GILL/, 9, 14, 114, 123, 126
YIND	Y-component of inertial velocity (integral of acceleration)	ft/sec	/GILL/, 114, 126
YIN	Inertial Y-component of the difference between inertial position and launch point	ft	/GILL/, 6

SUBROUTINE NEWTON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
ZDIND	Z-component of inertial acceleration	ft/sec^2	/GILL/, 19, 74, 112, 121
ZDIN	Z-component of inertial velocity (derivative of position)	ft/sec	/GILL/, 9, 15, 115, 124, 127
ZIND	Z-component of inertial velocity (integral of acceleration)	ft/sec	/GILL/, 115, 127
ZIN	Inertial Z-component of the difference between inertial position and launch point	ft	/GILL/, 7

```
$IBFTC NEWTØN  FULIST,REF
      SUBRØUTINE NEWTØN
C             CALCULATES DERIVATIVES FØR INTEGRATIØN
C
      CØMMØN/TAPES/ITAPE,ØTAPE,STAPE
      INTEGER  ØTAPE,STAPE
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $              TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL  INFIN
      CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,
     $              DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $              SECTS(20,12),STAGS(20,6),HEAD(11,15)
      REAL  LATI,LØNGI
      INTEGER  TARGET
      INTEGER  FLTR,FLSC,FLPRT
      CØMMØN/WFLAG/FLEØF,FLEØS,FLTP,FLDSC,TYSC,FLTRW
      LØGICAL FLEØF,FLEØS,FLTP
      INTEGER  FLDSC,TYSC,FLTRW
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $              PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $              PADSK,VESK,PADRT1,PADRT2,
     $             SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $              ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL  ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $              AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $              WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $             W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $             DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $             WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $              PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $              GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $              QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $              CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $              ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN/CØMP/THRSTX,THRSTY,THRSTZ,GRAVX,GRAVY,GRAVZ,
     $              DRAGX,DRAGY,DRAGZ,LIFTX,LIFTY,LIFTZ,
     $              VRX,VRY,VRZ
      REAL  LIFTX,LIFTY,LIFTZ
      CØMMØN/TABLES/TMACH(30,6),TCSBN(30,6),TCSBA(30,6),
     $              TTPIT(30,6),TPIT(30,6),TTYAW(30,6),TYAW(30,6),
     $              TTTRS(30,6),TTRUS(30,6),TTWDT(30,6),TWDØT(30,6),
     $              JCSBN(6),JCSBA(6),JPIT(6),JYAW(6),JTRUS(6),JWDØT(6),
     $              LPIT(6),LYAW(6),LTRUS(6),LWDØT(6),
     $              KMACH(6),KPIT(6),KYAW(6),KTRUS(6),KWDØT(6),
     $              IMACH(6),IPIT(6),IYAW(6),ITRUS(6),IWDØT(6)
```

FIGURE 22. SYMBOLIC LISTING OF SUBROUTINE NEWTON

```
      CØMMØN/ENGINW/PCBBR,WØCBB,WFCBB,WØGBB,WFGBB,
     $             PCBAR,WØCBR,WFCBR,WØGBR,WFGBR,
     $             PØNØM,PFNØM,TØNØM,TFNØM,FØPNØM,ETANØM,
     $             PIND,TREF,DREF,FPTPA,PTØG,PTFG,TSØ1,PTØ1,TSØ2,PTØ2,
     $             C1Ø,C2Ø,TSF1,PTF1,TSF2,PTF2,C1F,C2F,
     $             LØ,RØ,UHØ,LHØ,LF,RF,UHF,LHF(5),
     $             NØFE,FTURB,FØPT,KURF,DTBUP,VØCØFF,VFCØFF,EPVØL,
     $             DTDCY,WMSDT,WMØSD,WMFSD,WMØED,WMFED,RESØ,RESF,
     $             ASUBT,EPSI,ETACF,ETACS,TØI,CTØ,TFI,CTF,
     $             VØBB,VFBB,HØCØL,HFCØL,PARAML,VPARL,
     $             WKPC(6),WKWØC(6),WKWFC(6),WKWØG(6),WKWFG(6),
     $             TBLDUP,TEND,FLGHD,FLGSD,LSTAGE,CSTRN
      LØGICAL  FLGHD,FLGSD
      REAL  LØ,LF,NØFE,KURF
      INTEGER  PIND,FPTPA,UHØ,UHF
      CØMMØN /DUMPY/ DUMPO(10),DUMP1(10),DUMP2(10),DUMP3(10),DUMP4(10),
     $             DUMP5(10),DUMP6(10),DUMP7(10),DUMP8(10),DUMP9(10)
C
      REAL  MASS
C
C          CALCULATE RELATIVE SECTIØN AND STAGE TIMES
      TFLIT = TIME - TI
      TSTAG = TIME - TSTG
      TSECT = TIME - TSUBS
      MASS = W/GSUB1
C
C        CALCULATE INERTIAL PØSITIØN
      RXIN = RØX + XIN
      RYIN = RØY + YIN
      RZIN = RØZ + ZIN
      RVCTR = SQRT(RXIN**2 + RYIN**2 + RZIN**2)
C
C        CØNVERT INERTIAL PØSITIØN AND VELØCITY TØ PØLAR CØØRDINATES
      CALL G2ØB(RXIN,RYIN,RZIN,XDIN,YDIN,ZDIN,RSUBØ,FLAT,PHI,THETA,
     $        HATM,VSUBI,GAMIN,AZIN,1)
      ESUBP = RSUBO**2 * GSUB1/RVCTR
      ESUBK = VSUBI**2/2.0
      ERATIØ = ESUBK/ESUBP
C
C        CALCULATE RELATIVE VELØCITY, GAMMA, AZIMUTH
      VRX = XDIN + ØMEGA * RYIN
      VRY = YDIN - ØMEGA * RXIN
      VRZ = ZDIN
C        DURING VERTICAL RISE ØF FIRST SECTIØN, RØUND-ØFF MAY GIVE
C             INACCURATE CØMPØNENTS ØF RELATIVE VELØCITY
      IF (TYSC .EQ. 1 .AND. ATIND .EQ. 1)  CALL G1ØB(0.0,0.0,PHI,THETA,
     $             0.0,SQRT(VRX**2+VRY**2+VRZ**2),PIØ2,AZIR,U,U,U,
     $             VRX,VRY,VRZ,1)
      CALL G2ØB(RXIN,RYIN,RZIN,VRX,VRY,VRZ,0.0,0.0,U,U,U,VSUBR,GAMMA,AZ,
     $        1)
```

FIGURE 22. SYMBOLIC LISTING OF SUBROUTINE NEWTON (cont.)

```
      IF(ABS(ABS(GAMMA) - PIØ2) .LT. .0000002) AZ=AZIR
C        IF INITIAL TIME ØR HØLD-DØWN, USE INPUT V,H,GAMMA,AZ
      IF (XDIND+YDIND+ZDIND .NE. 0.0)  GØ TØ 50
      HATM = HI
      VSUBR = VI
      GAMMA = PIØ2 - BETAI/DPRAD
      AZ = AZIR
      GØ TØ 55
   50 IF (VSUBR .NE. 0.0)  GØ TØ 55
      GAMMA = 0.0
      AZ = AZIR
   55 CØNTINUE
      BETA = PIØ2 - GAMMA
C
C        CHECK FØR TRAJECTØRY ERRØR
      IF (HATM .GT. HMIN)  GØ TØ 10
      CALL SETERR(2H-H)
      GØ TØ 900
   10 IF (W .GT. 0.0)  GØ TØ 20
      CALL SETERR(2H-W)
      GØ TØ 900
   20 IF (W .GT. WPL)  GØ TØ 30
      CALL SETERR(5HW(PL))
      GØ TØ 900
C        TRAJECTØRY ERRØR -- ABØRT INTEGRATIØN
  900 CALL DUMPS(DUMP9,10,ØTAPE)
      KPASS = 4
      RETURN
C
   30 CØNTINUE
C
C        CALCULATE LATITUDE,LØNGITUDE, AND SURFACE RANGE
      LAT = ATAN(TAN(PHI)/(1.0 - FLAT)**2) * DPRAD
      THETAP = THETAØ + ØMEGA * (TIME - TI)
      LØNG = AMØD(LØNGI - (THETA - THETAP) * DPRAD,360.0)
      IF (LØNG .LT. 0.0)  LØNG = LØNG + 360.0
C
      DTHETA = THETA - THETAP
      IF (DTHETA .LT. 0.0)  DTHETA = DTHETA + 2.0*PI
      IF (DTHETA .GT. PI .AND. TIME .LT. 50.0)  DTHETA = 0.0
      IF (AZI .GT. 180.0)  DTHETA = 2.0*PI - DTHETA
      CPSI = SPHIØ * SIN(PHI) + CPHIØ * CØS(PHI) * CØS(DTHETA)
      IF (CPSI .LT. .99998)  GØ TØ 65
      ANGLE1 = PHI - PHIØ
      ANGLE2 = DTHETA
      IF (ABS(ANGLE2) .GT. PI)  ANGLE2 = SIGN(2.0*PI-ABS(ANGLE2),ANGLE2)
      PSI = SQRT(ANGLE1**2 + ANGLE2**2 * CØS(PHIØ)**2)
      GØ TØ 66
   65 PSI = ACØS(CPSI)
   66 RCIRC = PSI
```

FIGURE 22. SYMBOLIC LISTING OF SUBROUTINE NEWTON (cont.)

```
      IF (DTHETA .GT. PI)  RCIRC = 2.0*PI - PSI
      RANGE = RCIRC * RSUBØ/6076.1033
C
C          CALCULATE ATMØSPHERIC PARAMETERS
      IF (HATM .GT. HMAX)  GØ TØ 70
      CALL STDATM(HATM,RATM,CATM,TATM,PATM)
      GØ TØ 75
   70 RATM = 0.0
      TATM = 0.0
      PATM = 0.0
      CATM = CMAX
   75 CØNTINUE
      MACH = VSUBR/CATM
      QATM = RATM * VSUBR**2/2.0
C
C
C          CALCULATE GAMMA RELATIVE TØ LAUNCH SITE (GAMMA-REF)
      CALL G2ØB(RØX,RØY,RØZ,VRX,VRY,VRZ,0.0,0.0,U,U,U,U,GAMRF,AZREF,1)
      IF (ABS(GAMRF) - PIØ2 .LT. .0000002)  AZREF = AZIR
      IF (XDIND+YDIND+ZDIND .NE. 0.0)  GØ TØ 80
      GAMRF = GAMMA
      AZREF = AZ
      GØ TØ 85
   80 IF (VSUBR .NE. 0.0)  GØ TØ 85
      GAMRF = GAMMA
      AZREF = AZ
   85 CØNTINUE
C
C          CALCULATE MISSILE ATTITUDE AND ANGLE ØF ATTACK
      CALL GUIDE
      ALPHAP = PIØ2 - GAMRF - PIT
      ALPHAY = AZIR - AZREF + YAW
      ALPHA = SQRT(ALPHAP**2 + ALPHAY**2)
      ALPHAD = ALPHA * DPRAD
C          CALCULATE AERØDYNAMIC FØRCES
      CALL NTERP(TMACH(1,STIND),TCSBN(1,STIND),30,MACH,CSUBN,U,2,
     $           KMACH(STIND),L)
      CALL NTERP(TMACH(1,STIND),TCSBA(1,STIND),30,MACH,CSUBA,U,2,
     $           KMACH(STIND),L)
      CSUBN = KSUBN * CSUBN * ALPHAD
      CSUBA = KSUBA * CSUBA
      SALPHA = SIN(ALPHA)
      CALPHA = CØS(ALPHA)
      CSUBL = CSUBN * CALPHA - CSUBA * SALPHA
      CSUBD = CSUBN * SALPHA + CSUBA * CALPHA
      LIFT = CSUBL * QATM * SREF
      DRAG = CSUBD * QATM * SREF
C
C          CALCULATE LIFT AND DRAG CØMPØNENTS
      CALL G1ØB(0.0,0.0,PHI,THETA,0.0,LIFT,GAMMA+PIØ2,AZ,U,U,U,LIFTX,
     $              LIFTY,LIFTZ,1)
      DRAGX = -DRAG * VRX/VSUBR
```

FIGURE 22. SYMBOLIC LISTING OF SUBROUTINE NEWTON (cont.)

```
      DRAGY = -DRAG * VRY/VSUBR
      DRAGZ = -DRAG * VRZ/VSUBR
C
C          CALCULATE GRAVITY AND CØMPØNENTS
      GRAV = GSUBO * (RSUBØ/RVCTR)**2
      CALL G1ØB(0.0,0.0,PHI,THETA,0.0,GRAV,-PIØ2,0.0,U,U,U,GRAVX,
     $             GRAVY,GRAVZ,1)
C
C          CALCULATE THRUST AND FLØW RATES
      CALL ENGINE
      IF (FLDSC .LT. 0)  CALL ENGINE
      IF (TIME .EQ. TI)  CALL ENGINE
C
C          CALCULATE THRUST CØMPØNENTS
      CALL G1ØB(0.0,0.0,PHIØ,THETAØ,0.0,THRST,PIØ2-PIT,AZIR+YAW,U,U,U,
     $             THRSTX,THRSTY,THRSTZ,1)
C
C          SUM FØRCES FØR INERTIAL ACCELERATIØN
      FXIN = THRSTX + DRAGX + LIFTX
      FYIN = THRSTY + DRAGY + LIFTY
      FZIN = THRSTZ + DRAGZ + LIFTZ
      XDIND = FXIN/MASS + GRAVX
      YDIND = FYIN/MASS + GRAVY
      ZDIND = FZIN/MASS + GRAVZ
      XIND = XDIN
      YIND = YDIN
      ZIND = ZDIN
      ACCEL = (THRST - DRAG)/W
      CALL DUMPS(DUMP8,10,ØTAPE)
      IF (.NØT. FLGHD)  RETURN
C
C        IF HØLD-DØWN, INERTIAL ACCELERATIØN IS ZERØ
C             INERTIAL VELØCITY IS RØTATIØNAL VELØCITY
      XDIND = 0.0
      YDIND = 0.0
      ZDIND = 0.0
      XDIN = -ØMEGA * RYIN
      YDIN = ØMEGA * RXIN
      ZDIN = 0.0
      XIND = XDIN
      YIND = YDIN
      ZIND = ZDIN
      ACCEL = 0.0
      RETURN
      END
```

FIGURE 21. SYMBOLIC LISTING OF SUBROUTINE NEWTON (cont.)

XIX. SUBROUTINE GUIDE

This subroutine computes the vehicle attitude for six attitude control options. The six options are:

<u>Option 1</u>:	Gravity turn
<u>Option 2</u>:	Vertical rise
<u>Option 3</u>:	Constant turning rates
<u>Option 4</u>:	Tabular pitch and yaw
<u>Option 5</u>:	Tabular angle of attack
<u>Option 6</u>:	Guidance system simulation which provides steering for a velocity-to-go requirement

CALLING SEQUENCE

The subroutine calling sequence is:

```
CALL GUIDE
```

Subroutine GUIDE uses COMMON blocks CONST, INPUT, WORKD, SAVE, GILL, DATA, COMP and TABLES.

SOLUTION METHOD

Set the guidance control indicator.

1. I = ATIND + 1

Determine the guidance shceme.

2. If I = 1, go to 3
   If I = 2, go to 8
   If I = 3, go to 21
   If I = 4, go to 25
   If I = 5, go to 31
   If I = 6, go to 49

Gravity Turn

Compute pitch.

3. PIT = PIO2 - GAMRF

Compute yaw.

4. YAW = AZREF - AZIR

Set pitch and yaw turning rates to zero.

5. PRATE = 0
6. YRATE = 0
7. Go to 35

Vertical Rise

Compute orientation of position vector.

```
 8.  CALL G2OB (ROX, ROY, ROZ, RXIN, RYIN, RZIN,
                    0, 0, U, U, U, R, PIT, YAW, 1)

 9.  If | PIT | - PIO2 < .0000002, YAW = AZIR
     If | PIT | - PIO2 ≥ .0000002, YAW = YAW

10.  PIT = PIO2 - PIT

11.  YAW = YAW - AZIR
```

Compute rate of change of position vector.

```
12.  RDOT = ((RXIN)(XDIN) + (RYIN)(YDIN)
                 + (RZIN)(ZDIN))/(R)
```

Compute North, East and Down components and their rates of change, of the position vector transposed to the launch point.

```
13.  RSUBN = - (RXIN)(CTHETO)(SPHIO)
               - (RYIN)(STHETO)(SPHIO) + (RZIN)(CPHIO)

14.  RNDOT = - (XDIN)(CTHETO)SPHIO)
               - (YDIN)(STHETO)(SPHIO) + (ZDIN)(CPHIO)

15.  RSUBE = - (RXIN)(STHETO) + (RYIN)(CTHETO)

16.  REDOT = - (XDIN)(STHETO) + (YDIN)(CTHETO)
```

17. RSUBD = - (RXIN)(CTHETO)(CPHIO)
- (RYIN)(STHETO) (CPHIO) - (RZIN)(SPHIO)

18. RDDOT = - (XDIN)(CTHETO)(CPHIO)
- (YDIN)(STHETO)(CPHIO) - (ZDIN)(SPHIO)

Compute pitch and yaw rates.

19. PRATE = - ((RSUBD)(RDOT) - (R)(RDDOT))/
$((R^2)(1 - ((RSUBD)/(R))^2)^{.5})$

20. YRATE = ((RSUBN)(REDOT) - (RSUBE)(RNDOT))/
$(RSUBN^2 + RSUBE^2)$

Constant Turning Rates

21. PIT = PSUBS + PKIK + (PDOT)(TSECT)

22. YAW = YSUBS + YKIK + (YDOT)(TSECT)

23. PRATE = PDOT

24. YRATE = YDOT

Return

Tabular Pitch and Yaw

Determine the pitch angle and pitch rate.

25. CALL NTERP(TTPIT(1, STIND), TPIT(1, STIND),
IPIT(STIND), CLA(LPIT(STIND)), PIT,
PRATE, JPIT(STIND), KPIT(STIND), K)

Determine the yaw angle and yaw rate.

26. CALL NTERP (TTYAW(1, STIND), TYAW(1, STIND),
IYAW(STIND), CLA(LYAW(STIND)), YAW,
YRATE, JYAW(STIND), KYAW(STIND), K)

Scale and convert the angles and turning rates.

27. PIT = (KSUBP)(PIT)/(DPRAD)

28. YAW = (KSUBY)(YAW)/(DPRAD)

29. PRATE = (KSUBP)(PRATE)/(DPRAD)

30. YRATE = (KSUBY)(YRATE)/(DPRAD)

Return

Tabular Angle of Attack

Determine the angle of attack and its rate of change in the pitch plane.

31. CALL NTERP (TTPIT(1, STIND), TPIT(1, STIND),
IPIT(STIND), CLA(LPIT(STIND)), PIT,
PRATE, JPIT(STIND), KPIT(STIND), K)

Determine the angle of attack and its rate of change in the yaw plane.

32. CALL NTERP (TTYAW(1, STIND), TYAW(1, STIND),
IYAW(STIND), CLA(LYAW(STIND)), YAW,
YRATE, JYAW(STIND), KYAW(STIND), K)

Convert and scale the pitch plane component of the angle of attack and compute pitch.

33. PIT = PIO2 - GAMRF - (PIT)(KSUBP)/(DPRAD)

Convert and scale the yaw plane component of the angle of attack and compute yaw.

34. YAW = (AZI + YAW)(KSUBY)/(DPRAD)

Compute components of relative acceleration vector.

35. VRXDT = XDIND + (OMEGA)(YIND)

36. VRYDT = YDIND - (OMEGA)(XIND)

37. VRZDT = ZDIND

Compute magnitude of relative acceleration vector.

38. VRDOT = ((VRX)(VRXDT) + (VRY)(VRYDT)
+ (VRZ)(VRZDT))/(VSUBR)

Compute North, East and Down components of velocity and acceleration vectors transposed to the launch position.

39. VSUBN = - (VRX)(CTHETO)(SPHIO)
- (VRY)(STHETO)(SPHIO) + (VRZ)(CPHIO)

40. VNDOT = - (VRXDT)(CTHETO)(SPHIO)
- (VRYDT)(STHETO)(SPHIO) + (VRZDT)(CPHIO)

41. VSUBE = - (VRX)(STHETO) + (VRY)(CTHETO)

42. VEDOT = - (VRXDT)(STHETO) + (VRYDT)(CTHETO)

43. VSUBD = - (VRX)(CTHETO)(CPHIO)
- (VRY)(STHETO)(CPHIO) - (VRZ)(SPHIO)

44. VDDOT = - (VRXDT)(CTHETO)(CPHIO)
- (VRYDT)(STHETO)(CPHIO) - (VRZDT)(SPHIO)

Compute the rate of change of the angle between the velocity vector transposed to the launch site, and the launch horizontal.

45. GRFDT = ((VSUBD)(VRDOT) - (VSUBR)(VDDOT))/
((VSUBR^2)(1 - ((VSUBD)/(VSUBR))^2)$^{.5}$)

Compute the rate of change of the azimuth of the velocity vector transposed to the launch position.

46. AZRFDT = ((VSUBN)(VEDOT) - (VSUBE)(VNDOT))/
(VSUBN^2 + VSUBE^2)

Compute pitch and yaw rates.

47. PRATE = - GRFDT - (PRATE)(KSUBP)/(DPRAD)

48. YRATE = AZRFDT - (YRATE)(KSUBY)/(DPRAD)

Return

Guidance System Simulation

The guidance system simulation is executed only once per integration step.

49. If KPASS $<$ 4, return

If KPASS $\geq$ 4, go to 51

Compute required components of velocity vector to match velocity-to-go.

50. VX = VGX - XDIN

51. VY = VGY - YDIN

52. VZ = VGZ - ZDIN

Compute magnitude and orientation of required velocity vector.

53. CALL G2OB (ROX, ROY, ROZ, VX, VY, VZ, 0, 0, U, U, U, VTOGO, PIT, YAW, 1)

Compute pitch and yaw angles.

54. PIT = PIO2 - PIT

55. YAW = YAW - AZIR

Compute rate of change of required velocity vector.

56. VDOT = (- (VX)(XDIND) - (VY)(YDIND) - (VZ)(ZDIND))/(VTOGO)

Compute North, East and Down components and their rates of change, of the required velocity vector transposed to the launch position.

57. VSUBN = - (VX)(CTHETO)(SPHIO)

- (VY)(STHETO)(SPHIO) + (VZ)(CPHIO)

58. VNDOT = - (XDIND)(CTHETO)(SPHIO)

- (YDIND)(STHETO)(SPHIO) + (ZDIND)(CPHIO)

59. VSUBE = - (VX)(STHETO) + (VY)(CTHETO)

60. VEDOT = - (XDIND)(STHETO) + (YDIND)(CTHETO)

61. VSUBD = - (VX)(CTHETO)(CPHIO)

- (VY)(STHETO)(CPHIO) - (VZ)(SPHIO)

62. VDDOT = - (XDIND)(CTHETO)(CPHIO)

- (YDIND)(STHETO)(CPHIO)

- (ZDIND)(SPHIO)

Compute pitch and yaw rates.

63. PRATE = - ((VSUBD)(VDOT) - (VTOGO)(VDDOT))/

((VTOGO^2)(1 - ((VSUBD)/(VTOGO))^2)$^{.5}$)

64. YRATE = ((VSUBN)(VEDOT) - (VSUBE)(VNDOT))/

(VSUBN^2 + VSUBE^2)

Return

SUBROUTINE GUIDE NOMENCLATURE

Symbol	Description	Units	Reference
ATIND	Attitude control indicator	—	/WORKD/, INTEGER, 1
AZI	Initial azimuth	deg	/INPUT/, 34
AZIR	Initial azimuth	rad	/SAVE/, 4, 9, 11, 55
AZREF	Reference azimuth	rad	/DATA/, 4
AZRFDT	Rate of change of reference azimuth	rad/sec	46, 48
CPHIO	Cosine of initial geocentric latitude	—	/SAVE/, 13, 14, 17, 18, 39, 40, 43, 44, 57, 58, 61, 62
CTHETO	Cosine of initial sidereal time	—	/SAVE/, 13, 14, 15, 16, 17, 18, 39, 40, 41, 42, 43, 44, 57, 58, 59, 60, 61, 62
DPRAD	Conversion factor	deg/rad	/CONST/, 27, 28, 29, 30, 33, 34, 47, 48
GAMRF	Reference flight path angle	rad	/DATA/, 3, 33
GRFDT	Rate of change of reference flight path angle	—	45, 47
I	Guidance control indicator	—	1, 2
IPIT	Size of tabular pitch or tabular pitch angle of attack tables	—	/TABLES/, 25, 31
IYAW	Size of tabular yaw or tabular yaw angle of attack tables	—	/TABLES/, 26, 32

SUBROUTINE GUIDE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
JPIT	Degree of interpolation for tabular pitch or tabular pitch angle of attack tables	—	/TABLES/, 25, 31
JYAW	Degree of interpolation for tabular yaw or tabular yaw angle of attack tables	—	/TABLES/, 26, 32
K	Interpolation error indicator	—	25, 26, 31, 32
KPASS	Pass counter used in Subroutine INTEG	—	/GILL/, 49
KPIT	Pitch or pitch angle of attack location indicator	—	/TABLES/, 25, 31
KSUBP	Pitch angle scale factor	—	/WORKD/, REAL, 27, 29, 33, 47
KSUBY	Yaw angle scale factor	—	/WORKD/, REAL, 28, 30, 34, 48
KYAW	Yaw or yaw angle of attack location indicator	—	/TABLES/, 26, 32
LPIT	Pitch table or pitch angle of attack table independent argument	—	/TABLES/, 25, 31
LYAW	Yaw table or yaw angle of attack table independent argument	—	/TABLES/, 26, 32
OMEGA	Earth rotation rate	rad/sec	/CONST/, 35, 36
PDOT	Constant pitch turning rate	rad/sec	/WORKD/, 21, 23

SUBROUTINE GUIDE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PIO2	$(\pi)/(2)$	—	/CONST/, 3, 9, 10, 33, 54
PIT	Pitch angle or pitch angle of attack	deg or rad	/DATA/, 3, 8, 9, 10, 21, 25, 27, 31, 33, 53, 54
PKIK	Instantaneous pitch kick at beginning of section	rad	/WORKD/, 21
PRATE	Rate of change of pitch angle or pitch angle of attack	deg/sec or rad/sec	/DATA/, 5, 19, 23, 25, 29, 31, 47, 63
PSUBS	Pitch angle at beginning of section	rad	/SAVE/, 21
R	Magnitude of position vector	ft	8, 12, 19
RDOT	Rate of change of position vector	ft/sec	12, 19
RDDOT	Rate of change of Down component of position vector transposed to launch site	ft/sec	18, 19
REDOT	Rate of change of East component of position vector transposed to launch site	ft/sec	16, 20
RNDOT	Rate of change of North component of position vector transposed to launch site	ft/sec	14, 20
ROX	X-component of initial position vector	ft	/SAVE/, 8, 53

SUBROUTINE GUIDE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
ROY	Y-component of initial position vector	ft	/SAVE/, 8, 53
ROZ	Z-component of initial position vector	ft	/SAVE/, 8, 53
RSUBD	Down component of position vector transposed to launch site	ft	17, 19
RSUBE	East component of position vector transposed to launch site	ft	15, 20
RSUBN	North component of position vector transposed to launch site	ft	13, 20
RXIN	X-component of position vector	ft	/DATA/, 8, 12, 13, 15, 17
RYIN	Y-component of position vector	ft	/DATA/, 8, 12, 13, 15, 17
RZIN	Z-component of position vector	ft	/DATA/, 8, 12, 13, 17
SPHIO	Sine of initial geocentric latitude	—	/SAVE/, 13, 14, 17, 18, 39, 40, 43, 44, 57, 58, 61, 62
STHETO	Sine of initial sidereal time	—	/SAVE/, 13, 14, 15, 16, 17, 18, 39, 40, 41, 42, 43, 44, 57, 58, 59, 60, 61, 62

SUBROUTINE GUIDE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
STIND	Stage indicator	—	/WORKD/, INTEGER, 25, 26, 31, 32
TPIT	Pitch angle or pitch angle of attack input array	deg	/TABLES/, 25, 31
TSECT	Relative section time	sec	/DATA/, 21, 22
TTPIT	Pitch angle or pitch angle of attack independent argument input array	—	/TABLES/, 25, 31
TTYAW	Yaw angle or yaw angle of attack independent argument input array	—	/TABLES/, 26, 32
TYAW	Yaw angle or yaw angle of attack input array	deg	/TABLES/, 26, 32
U	Dummy variable	—	8, 53
VDDOT	Rate of change of Down component of velocity vector transposed to launch site	ft/sec^2	44, 45, 62, 63
VDOT	Rate of change of inertial velocity vector	ft/sec^2	56, 63
VEDOT	Rate of change of East component of velocity vector transposed to launch site	ft/sec^2	42, 46, 60, 64
VGX	X-component of velocity-to-go	ft/sec	/WORKD/, 50
VGY	Y-component of velocity-to-go	ft/sec	/WORKD/, 51
VGZ	Z-component of velocity-to-go	ft/sec	/WORKD/, 52

SUBROUTINE GUIDE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
VNDOT	Rate of change of North component of velocity vector transposed to launch site	ft/sec^2	40, 46, 58, 64
VRDOT	Rate of change of relative velocity vector	ft/sec^2	38, 45
VRX	X-component of relative velocity vector	ft/sec	/COMP/, 38, 39, 41, 43
VRXDT	Rate of change of X-component of relative velocity vector	ft/sec^2	35, 38, 40, 42, 44
VRY	Y-component of relative velocity vector	ft/sec	/COMP/, 38, 39, 41, 43
VRYDT	Rate of change of Y-component of relative velocity vector	ft/sec^2	36, 38, 40, 42, 44
VRZ	Z-component of relative velocity vector	ft/sec	/COMP/, 38, 39, 43
VRZDT	Rate of change of Z-component of relative velocity vector	ft/sec^2	37, 38, 40, 44
VSUBD	Down component of velocity vector transposed to launch site	ft/sec	43, 45, 61, 63
VSUBE	East component of velocity vector transposed to launch site	ft/sec	41, 46, 59, 64

SUBROUTINE GUIDE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
VSUBN	North component of velocity vector transposed to launch site	ft/sec	39, 46, 57, 64
VSUBR	Magnitude of relative velocity vector	ft/sec	/DATA/, 38, 45
VTOGO	Velocity-to-go	ft/sec	53, 56, 63
VX	X-component of required velocity vector to match velocity-to-go	ft/sec	50, 53, 56, 57, 59, 61
VY	Y-component of required velocity vector to match velocity-to-go	ft/sec	51, 53, 56, 57, 59, 61
VZ	Z-component of required velocity vector to match velocity-to-go	ft/sec	52, 53, 56, 57, 61
XDIN	X-component of inertial velocity vector	ft/sec	/GILL/, 12, 14, 16, 18, 50
XDIND	X-component of inertial acceleration	ft/sec^2	/GILL/, 35, 56, 58, 60, 62
XIND	X-component of inertial velocity vector	ft/sec	/GILL/, 36
YAW	Yaw angle or yaw angle of attack	deg or rad	/DATA/, 4, 8, 9, 11, 22, 26, 28, 32, 34, 53, 55
YDIN	Y-component of inertial velocity vector	ft/sec	/GILL/, 12, 14, 16, 18, 51

SUBROUTINE GUIDE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
YDIND	Y-component of inertial acceleration	ft/sec^2	/GILL/, 36, 56, 58, 60, 62
YDOT	Constant pitch turning rate	rad/sec	/WORKD/, 22, 24
YIND	Y-component of inertial velocity vector	ft/sec	/GILL/, 35
YKIK	Instantaneous yaw kick at beginning of section	rad	/WORKD/, 23
YRATE	Rate of change of yaw angle or yaw angle of attack	deg/sec or rad/sec	/DATA/, 6, 20, 24, 26, 30, 32, 48, 64
YSUBS	Yaw angle at beginning of section	rad	/SAVE/, 22
ZDIN	Z-component of inertial velocity vector	ft/sec	/GILL/, 12, 14, 18, 52
ZDIND	Z-component of inertial acceleration	ft/sec^2	/GILL/, 37, 56, 58, 62

```
$IBFTC GUIDE   FULIST,REF
      SUBRØUTINE GUIDE
C            CALCULATES MISSILE ATTITUDE FØR VARIØUS ATTITUDE CØNTRØL
C                  ØPTIØNS
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $             TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL   INFIN
      CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,
     $             DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $             SECTS(20,12),STAGS(20,6),HEAD(11,15)
      REAL  LATI,LØNGI
      INTEGER  TARGET
      INTEGER  FLTR,FLSC,FLPRT
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $             PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $             PADSK,VESK,PADRT1,PADRT2,
     $            SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $             ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL  ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $             AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $             WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $            W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $            DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $            WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $             PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $             GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $             QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $             CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $             ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN/CØMP/THRSTX,THRSTY,THRSTZ,GRAVX,GRAVY,GRAVZ,
     $             DRAGX,DRAGY,DRAGZ,LIFTX,LIFTY,LIFTZ,
     $             VRX,VRY,VRZ
      REAL  LIFTX,LIFTY,LIFTZ
      CØMMØN/TABLES/TMACH(30,6),TCSBN(30,6),TCSBA(30,6),
     $             TTPIT(30,6),TPIT(30,6),TTYAW(30,6),TYAW(30,6),
     $             TTTRS(30,6),TTRUS(30,6),TTWDT(30,6),TWDØT(30,6),
     $             JCSBN(6),JCSBA(6),JPIT(6),JYAW(6),JTRUS(6),JWDØT(6),
     $             LPIT(6),LYAW(6),LTRUS(6),LWDØT(6),
     $             KMACH(6),KPIT(6),KYAW(6),KTRUS(6),KWDØT(6),
     $             IMACH(6),IPIT(6),IYAW(6),ITRUS(6),IWDØT(6)
      I = ATIND + 1
      GØ TØ (10,20,30,40,50,60), I
C            GRAVITY TURN
   10 PIT = PIØ2 - GAMRF
      YAW = AZREF - AZIR
```

FIGURE 23. SYMBOLIC LISTING OF SUBROUTINE GUIDE

```
      PRATE = 0.0
      YRATE = 0.0
      GØ TØ 55
C           VERTICAL RISE
   20 CALL G2ØB(RØX,RØY,RØZ,RXIN,RYIN,RZIN,0.0,0.0,U,U,U,R,PIT,YAW,1)
      IF (ABS(PIT) - PIØ2 .LT. .0000002)  YAW = AZIR
      PIT = PIØ2 - PIT
      YAW = YAW - AZIR
      RDØT = (RXIN * XDIN + RYIN * YDIN + RZIN * ZDIN)/R
      RSUBN = -RXIN*CTHETØ*SPHIØ - RYIN*STHETØ*SPHIØ + RZIN*CPHIØ
      RNDØT = -XDIN*CTHETØ*SPHIØ - YDIN*STHETØ*SPHIØ + ZDIN*CPHIØ
      RSUBE = -RXIN*STHETØ + RYIN*CTHETØ
      REDØT = -XDIN*STHETØ + YDIN*CTHETØ
      RSUBD = -RXIN*CTHETØ*CPHIØ - RYIN*STHETØ*CPHIØ - RZIN*SPHIØ
      RDDØT = -XDIN*CTHETØ*CPHIØ - YDIN*STHETØ*CPHIØ - ZDIN*SPHIØ
      PRATE = -(RSUBD*RDØT - R*RDDØT)/(R**2 * SQRT(1.0 - (RSUBD/R)**2))
      YRATE = (RSUBN*REDØT - RSUBE*RNDØT)/(RSUBN**2 + RSUBE**2)
      RETURN
C           CØNSTANT TURNING RATES
   30 PIT = PSUBS + PKIK + PDØT * TSECT
      YAW = YSUBS + YKIK + YDØT * TSECT
      PRATE = PDØT
      YRATE = YDØT
      RETURN
C           TABULAR PITCH AND YAW
   40 CALL NTERP(TTPIT(1,STIND),TPIT(1,STIND),IPIT(STIND),
     $             CLA(LPIT(STIND)),PIT,PRATE,JPIT(STIND),KPIT(STIND),K)
      CALL NTERP(TTYAW(1,STIND),TYAW(1,STIND),IYAW(STIND),
     $             CLA(LYAW(STIND)),YAW,YRATE,JYAW(STIND),KYAW(STIND),K)
      PIT = KSUBP * PIT/DPRAD
      YAW = KSUBY * YAW/DPRAD
      PRATE = KSUBP * PRATE/DPRAD
      YRATE = KSUBY * YRATE/DPRAD
      RETURN
C           TABULAR ANGLE ØF ATTACK
   50 CALL NTERP(TTPIT(1,STIND),TPIT(1,STIND),IPIT(STIND),
     $             CLA(LPIT(STIND)),PIT,PRATE,JPIT(STIND),KPIT(STIND),K)
      CALL NTERP(TTYAW(1,STIND),TYAW(1,STIND),IYAW(STIND),
     $             CLA(LYAW(STIND)),YAW,YRATE,JYAW(STIND),KYAW(STIND),K)
      PIT = PIØ2 - GAMRF - PIT/DPRAD * KSUBP
      YAW = (AZI + YAW)/DPRAD * KSUBY
   55 VRXDT = XDIND + ØMEGA * YIND
      VRYDT = YDIND - ØMEGA * XIND
      VRZDT = ZDIND
      VRDØT = (VRX * VRXDT + VRY * VRYDT + VRZ * VRZDT)/VSUBR
      VSUBN = -VRX*CTHETØ*SPHIØ - VRY*STHETØ*SPHIØ + VRZ*CPHIØ
      VNDØT = - VRXDT*CTHETØ*SPHIØ - VRYDT*STHETØ*SPHIØ + VRZDT*CPHIØ
      VSUBE = - VRX*STHETØ + VRY*CTHETØ
      VEDØT = - VRXDT*STHETØ + VRYDT*CTHETØ
      VSUBD = - VRX*CTHETØ*CPHIØ - VRY*STHETØ*CPHIØ - VRZ*SPHIØ
```

FIGURE 23. SYMBOLIC LISTING OF SUBROUTINE GUIDE (cont.)

```
      VDDØT = - VRXDT*CTHETØ*CPHIØ - VRYDT*STHETØ*CPHIØ - VRZDT*SPHIØ
      GRFDT = (VSUBD*VRDØT - VSUBR*VDDØT)/(VSUBR**2 *
     $                                 SQRT(1.0 - (VSUBD/VSUBR)**2))
      AZRFDT = (VSUBN * VEDØT - VSUBE * VNDØT)/(VSUBN**2 + VSUBE**2)
      PRATE = -GRFDT - PRATE/DPRAD * KSUBP
      YRATE = AZRFDT - YRATE/DPRAD * KSUBY
      RETURN
C          GUIDANCE SYSTEM SIMULATIØN
   60 IF (KPASS .LT. 4)  RETURN
C          EXECUTE ØNLY ØNCE PER INTEGRATIØN STEP
      VX = VGX - XDIN
      VY = VGY - YDIN
      VZ = VGZ - ZDIN
      CALL G2ØB(RØX,RØY,RØZ,VX,VY,VZ,0.0,0.0,U,U,U,VTØGØ,PIT,YAW,1)
      PIT = PIØ2 - PIT
      YAW = YAW - AZIR
      VDØT = (-VX * XDIND - VY * YDIND - VZ * ZDIND)/VTØGØ
      VSUBN = - VX*CTHETØ*SPHIØ - VY*STHETØ*SPHIØ + VZ*CPHIØ
      VNDØT = - XDIND*CTHETØ*SPHIØ - YDIND*STHETØ*SPHIØ + ZDIND*CPHIØ
      VSUBE = - VX*STHETØ + VY*CTHETØ
      VEDØT = - XDIND*STHETØ + YDIND*CTHETØ
      VSUBD = - VX*CTHETØ*CPHIØ - VY*STHETØ*CPHIØ - VZ*SPHIØ
      VDDØT = - XDIND*CTHETØ*CPHIØ - YDIND*STHETØ*CPHIØ - ZDIND*SPHIØ
      PRATE = -(VSUBD*VDØT - VTØGØ*VDDØT)/(VTØGØ**2 *
     $                                 SQRT(1.0 - (VSUBD/VTØGØ)**2))
      YRATE = (VSUBN * VEDØT - VSUBE * VNDØT)/(VSUBN**2 + VSUBE**2)
      RETURN
      END
```

FIGURE 23. SYMBOLIC LISTING OF SUBROUTINE GUIDE (cont.)

XX. SUBROUTINE ENGINE

This subroutine computes the instantaneous total engine thrust and propellant flow rates during flight.

The program includes the following six options:

Option 1: The no thrust, or vehicle coast condition option.

Option 2: The analytic thrust option, where the vacuum thrust value is corrected for the influence of atmospheric pressure.

Option 3: The tabular option, where thrust and flow rate values are read from tables.

Option 4: The calculation of solid rocket engine performance from tabular vacuum thrust and constant specific impulse.

Option 5: The calculation of liquid rocket engine performance by influence coefficients.

Option 6: This option simulates a propulsion system composed of both liquid and solid rocket engines. It is a combination of Options 4 and 5.

CALLING SEQUENCE

The calling sequence is:

CALL ENGINE

Subroutine ENGINE uses COMMON blocks CONST, WORKD, SAVE, GILL, DATA, TABLES, ENGINI, ENGINO, and ENGINW.

SOLUTION METHOD

The thrust option indicator is set:

1. I = THIND + 1

The thrust option is selected:

2. If I = 1, go to 3

   If I = 2, go to 9

   If I = 3, go to 13

   If I = 4, go to 19

   If I = 5, go to 25

   If I = 6, go to 25

First Option: Coast

3. THRST = 0

4. WPDOT = 0

5. TBDOT = 0

6. WODOT = 0

7. WFDOT = 0

8. WDOT = WPDOT

   Return

Second Option: Analytic

```
9.   THRST  = FVAC - (EPSIS) (ASUBTS) (PATM)

10.  WPDOT = - WDOTV

11.  TBDOT  = 1

12.  Go to 6
```

Third Option: Tabular

```
Read thrust from table.

13.  CALL NTERP (TTTRS(1, STIND), TTRUS(1, STIND),
                ITRUS(STIND), CLA(LTRUS(STIND)), THRST,
                Z, JTRUS(STIND), KTRUS(STIND), L)

Scale thrust.

14.  THRST  = (THRST) (KSUBF)

Read total flow rate from table.

15.  CALL NTERP (TTWDT(1, STIND), TWDOT(1, STIND),
                IWDOT(STIND), CLA(LWDOT(STIND)), WPDOT,
                Z, JWDOT(STIND), KWDOT(STIND), L)

Scale total flow rate.

16.  WPDOT =  -(WPDOT)(KSUBW)

17.  TBDOT  = 1

18.  Go to 6
```

Fourth Option: Tabular Vacuum Thrust, Constant Specific Impulse

Read vacuum thrust from tables.

19. CALL NTERP (TTTRS(1, STIND), TTRUS(1, STIND),
ITRUS(STIND), CLA(LTRUS(STIND)), THRST,
Z, JTRUS(STIND), KTRUS(STIND), L)

Scale and correct thrust for ambient pressure.

20. THRST = (THRST) (KSUBF) - (EPSIS) (ASUBTS) (PATM)

Compute total flow rate.

21. WPDOT = - (THRST)/(ISP)

22. TBDOT = 1

Test if a combination solid/liquid propulsion system simulation is desired.

23. If THIND = 5, go to 108
If THIND ≠ 5, go to 24

24. Go to 6

Fifth Option: Influence Coefficients

Compute acceleration and test hold down flag.

25. ACCEL = (THRST - DRAG)/(W)

26. If FLGHD = .TRUE. , ACCEL = 0
If FLGHD = .FALSE., ACCEL = ACCEL

The pump inlet temperatures are computed:

27. CALL VPOLY(6, CTOX(1, STIND), (WO)/(WOIS), DELTO)

28. TO = TOI + (CTO)(DELTO)

29. CALL VPOLY (6, CTFL(1, STIND), (WF)/(WFIS), DELTF)

30. TF = TFI + (CTF)(DELTF)

The pump inlet densities are computed:

```
31.   CALL PRPDEN (1, PO, TO - 459.688, Z, Z, PVO,
                     RHOO)

32.   CALL PRPDEN (PIND + 1, PF, TF - 459.688, TREF,
                     DREF, PVF, RHOF)
```

The tank pressures are computed:

```
33.   If FPTPA $\neq$ 0, go to 37

      If FPTPA = 0, go to 34
```

Constant gage pressures.

```
34.   PTO = PTOG + PATM

35.   PTF = PTFG + PATM

36.   Go to 53
```

"Constant" stepped absolute pressures.

```
37.   If TSTAG $\geq$ TSO2, go to 44

      If TSTAG $<$ TSO2, go to 38

38.   If TSTAG $>$ TSO1, go to 41

      If TSTAG $\leq$ TSO1, go to 39

39.   PTO = PTO1

40.   Go to 45

41.   TSO = (TSTAG - TSO1) (PTO2 - PTO1)/(TSO2 - TSO1)

42.   PTO = PTO1 + (C1O) (TSO) + (C2O) (TSO$^2$)

43.   Go to 45
```

```
44.  PTO = PTO2
45.  If TSTAG ≥ TSF2, go to 52
     If TSTAG < TSF2, go to 46
46.  If TSTAG > TSF1, go to 49
     If TSTAG ≤ TSF1, go to 47
47.  PTF = PTF1
48.  Go to 53
49.  TSF = (TSTAG - TSF1)(PTF2 - PTF1)/(TSF2 - TSF1)
50.  PTF = PTF1 + (C1F)(TSF) + (C2F)($TSF^2$)
51.  Go to 53
52.  PTF = PTF2
53.  Continue
```

The tank volumes and propellant heights are calculated:

```
54.  HO = 0
55.  HF = 0
56.  VO = (WO)/(RHOO) - VOBB
57.  VF = (WF)/(RHOF) - VFBB
58.  If VO < 0, go to 60
     If VO ≥ 0, go to 59
59.  CALL HITE (VO, LO, HO)
```

```
60.  Continue
61.  If VF $< 0$, go to 63
     If VF $\geq 0$, go to 62
62.  CALL HITE (VF, LF, HF)
63.  Continue
64.  $HO = HO + HOCOL$
65.  $HF = HF + HFCOL$
```

The pump inlet pressures are determined:

```
66.  $PO = PTO + (RHOO)(ACCEL)(HO)/(144)$
          $- (RESO)(WOPDT^2)/((144)(RHOO))$
67.  $PF = PTF + (RHOF)(ACCEL)(HF)/(144)$
          $- (RESF)(WFPDT^2)/((144)(RHOF))$
```

The variances of the first four independent variables (pressures and temperatures) are computed:

```
68.  CALL VARYD (PO, XBAR(1, STIND), PAR, 4)
```

For use in a thrust level optimization study, the percentage change in the nominal thrust level requirement is determined:

```
69.  PAR(5) = ( KURF)(FOPT - XBAR(5, STIND))/(XBAR(5, STIND))
```

The thrust chamber flow rates with no C* correction are calculated:

70. CALL GICEQ(WOCDT, WOCBB, WOCBR, WKWOC, PAR, 5)

71. CALL GICEQ(WFCDT, WFCBB, WFCBR, WKWFC, PAR, 5)

The C* correction factor is computed:

72. CALL GICEQ (PC, PCBBR, PCBAR, WKPC, PAR, 5)

73. MRC = (WOCDT)/(WFCDT)

74. CALL TCPROP (PIND, MRC, PC, EPSI, CVTH, CFTH)

75. CSTR = (PC)(ASUBT)(GSUB1)/(WOCDT + WFCDT)

The per cent change in the C* correction factor is determined:

76. PAR(6) = ((CVTH)(ETACS) - CSTR)/(CSTRN)

The corrected chamber flow rates and mixture ratio are computed:

77. WOCDT = WOCDT + (WOCBR)(WKWOC(6))(PAR(6))

78. WFCDT = WFCDT + (WFCBR)(WKWFC(6))(PAR(6))

79. MRC = (WOCDT)/(WFCDT)

The chamber pressure and gas generator flow rates are calculated:

80. PC = PC + (PCBAR)(WKPC(6))(PAR(6))

81. CALL GICEQ(WOGDT, WOGBB, WOGBR, WKWOG, PAR, 6)

82. CALL GICEQ (WFGDT, WFGBB, WFGBR, WKWFG, PAR, 6)

The theoretical and actual thrust coefficients are computed:

83. CALL TCPROP (PIND, MRC, PC, EPSI, CVTH, CFTH)

84. CSUBF = (ETACF)(CFTH) - (EPSI)(PATM)/(PC)

The chamber and engine thrusts are calculated:

85. FCHMBR = (CSUBF)(PC)(ASUBT)

86. FENGIN = FCHMBR + FTURB

The engine flow rates and mixture ratio are computed:

87. WOEDT = WOCDT + WOGDT

88. WFEDT = WFCDT + WFGDT

89. MRE = (WOEDT)/(WFEDT)

90. WEDOT = WOEDT + WFEDT

91. ISPE = (FENGIN)/(WEDOT)

92. WOPDT = WOEDT + WMOED

93. WFPDT = WFEDT + WMFED

The total stage thrust, flow rates, mixture ratio, and specific impulse are computed:

94. WOSDT = (NOFE)(WOPDT) + WMOSD

95. WFSDT = (NOFE)(WFPDT) + WMFSD

96. MRS = (WOSDT)/(WFSDT)

97. WSDOT = WOSDT + WFSDT + WMSDT

98. FS = (NOFE)(FENGIN)

```
99.  ISPS = (FS)/(WSDOT)

Set performance level multiplier to 1:

100. KLVL = 1

Check for start-up:

101. If TIME $\geq$ TBLDUP, go to 104
     If TIME $<$ TBLDUP, go to 102
102. KLVL = (TIME - TSUBS)/(DTBUP)
103. Go to 105

Check for shutdown:

104. If FLGSD = .TRUE., KLVL = (TEND-TIME)/(DTDCY)
     If FLGSD = .FALSE., KLVL = KLVL

Check for combination solid/liquid propulsion system:

105. If THIND = 5, go to 19
     If THIND $\neq$ 5, go to 106

Compute final values of stage performance:

106. THRST = 0
107. WPDOT = 0
108. THRST = (FS)(KLVL) + THRST
```

```
109.  WDOT = - (WSDOT)(KLVL) + WPDOT

110.  WODOT = - (WOSDT)(KLVL)

111.  WFDOT = - (WFSDT)(KLVL)

112.  TBDOT = 1

113.  If FLGSD = .TRUE., return

      If FLGSD = .FALSE., go to 114

114.  WOR(STIND) = (VO - VOCOFF)(RHOO)

115.  WFR(STIND = (VF - VFCOFF)(RHOF)

116.  AVMR(STIND) = (WOIS - WO)/(WFIS - WF)

      Return
```

SUBROUTINE ENGINE NOMENCLATURE

Symbol	Description	Units	Reference
ACCEL	Axial acceleration	g's	/GILL/, 25, 26, 66, 67
ASUBT	Engine throat area	in^2	/ENGINW/, 75, 85
ASUBTS	Engine throat area	in^2	/WORKD/, 9, 20
AVMR	Average in-flight mixture ratio	—	/SAVE/, 116
C1F	First degree polynomial coefficient (fuel tank pressure)	—	/ENGINW/, 50
C2F	Second degree polynomial coefficient (fuel tank pressure)	—	/ENGINW/, 50
C1O	First degree polynomial coefficient (oxidizer tank pressure)	—	/ENGINW/, 42
C2O	Second degree polynomial coefficient (oxidizer tank pressure)	—	/ENGINW/, 42
CFTH	Theoretical thrust chamber thrust coefficient	—	/ENGINO/, 74, 83, 84
CLA	Entry to Subroutine UNPACK	—	13, 15, 19
CSTRN	Nominal C^* value	ft/sec	/ENGINW/, 76
CSTR	Instantaneous C^* value	ft/sec	/ENGINO/, 75, 76
CSUBF	Actual thrust chamber thrust coefficient	—	/ENGINO/, 84, 85

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
CTFL	Array of fuel temperature polynomial coefficients	—	/ENGINI/, 29
CTF	Fuel temperature rise scale factor	—	/ENGINW/, 30
CTO	Oxidizer temperature rise scale factor	—	/ENGINW/, 28
CTOX	Array of oxidizer temperature polynomial coefficients	—	/ENGINI/, 27
CVTH	Theoretical characteristic velocity	ft/sec	/ENGINO/, 74, 76, 83
DELTF	Fuel temperature rise at pump inlet	^{o}R	29, 30
DELTO	Oxidizer temperature rise at pump inlet	^{o}R	27, 28
DRAG	Drag force	lb	/DATA/, 25
DREF	Fuel specific gravity at reference temperature	—	/ENGINW/, 32
DTBUP	Time for thrust build-up	sec	/ENGINW/, 102
DTDCY	Thrust decay time	sec	/ENGINW/, 104
EPSI	Thrust chamber nozzle expansion area ratio	—	/ENGINW/, 74, 83, 84
EPSIS	Thrust chamber nozzle expansion area ratio	—	/WORKD/, 9, 20

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
ETACF	Nozzle efficiency	—	/ENGINW/, 84
ETACS	Combustion efficiency	—	/ENGINW/, 76
FCHMBR	Thrust chamber thrust	lb	/ENGINO/, 85, 86
FENGIN	Engine thrust	lb	/ENGINO/, 86, 91, 98
FLGHD	Hold down flag	—	/ENGINW/, LOGICAL, 26
FLGSD	Shutdown flag	—	/ENGINW/, LOGICAL, 104, 113
FPTPA	Tank pressure option flag	—	/ENGINW/, INTEGER, 33
FS	Total stage thrust	lb	/ENGINO/, 98, 99, 108
FTURB	Turbine exhaust thrust	lb	/ENGINW/, 86
FVAC	Vacuum thrust	lb	/WORKD/, 9
FOPT	Optimum thrust value	lb	/ENGINW/, 69
GSUB1	Dimensional constant	$lb_m ft/lb_f sec^2$	/CONST/, 75
HFCOL	Height from fuel tank bottom bulkhead to fuel pump inlet	ft	/ENGINW/, 65
HF	Fuel height above fuel pump inlet	ft	/ENGINO/, 55, 62, 65, 67
HOCOL	Height from oxidizer tank bottom bulkhead to oxidizer pump inlet	ft	/ENGINW/, 64

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
HO	Oxidizer height above oxidizer pump inlet	ft	/ENGINO/, 54, 59, 64, 66
I	Option indicator	—	1, 2
ISP	Input constant specific impulse	sec	/WORKD/, REAL, 21
ISPE	Engine specific impulse	sec	/ENGINO/, REAL, 91
ISPS	Stage specific impulse	sec	/ENGINO/, REAL, 99
ITRUS	Size of TTRUS and TTTRS arrays	—	/TABLES/, 13, 19
IWDOT	Size of TWDOT and TTWDT arrays	—	/TABLES/, 15
JTRUS	Indicator for degree of interpolation for thrust tables	—	/TABLES/, 13, 19
JWDOT	Indicator for degree of interpolation for flow rate tables	—	/TABLES/, 15
KLVL	Performance level multiplier	—	REAL, 100, 102, 104, 108, 109, 110, 111
KSUBF	Scale factor for tabular thrust	—	/WORKD/, REAL, 14, 20
KSUBW	Scale factor for tabular flow rate	—	/WORKD/, REAL, 16
KTRUS	Interpolation routine counter for tabular thrust	—	/TABLES/, 13, 19
KURF	Flag for optimized thrust study	—	/ENGINW/, REAL, 69

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
KWDOT	Interpolation routine counter for tabular flow rate	—	/TABLES/, 15
LF	Array of fuel tank data for Subroutine HITE	—	/ENGINW/, 62
LO	Array of oxidizer tank data for Subroutine HITE	—	/ENGINW/, 59
L	Interpolation routine error flag	—	13, 15
LTRUS	Interpolation routine argument indicator for tabular thrust	—	/TABLES/, 13, 19
LWDOT	Interpolation routine argument indicator for tabular flow rate	—	/TABLES/, 15
MRC	Thrust chamber mixture ratio	—	/ENGINO/, REAL, 73, 74, 79, 83
MRE	Engine mixture ratio	—	/ENGINO/, REAL, 89
MRS	Stage mixture ratio	—	/ENGINO/, REAL, 96
NOFE	Number of engines	—	/ENGINW/, REAL, 94, 95, 98
PAR	Array of ratio changes of engine independent variables from nominal values	—	DIM, 68, 69, 70, 71, 72, 76, 77, 78, 80, 81, 82
PATM	Atmospheric pressure	lb/in^2	/DATA/, 9, 20, 34, 35, 84
PCBAR	Nominal engine thrust chamber nozzle stagnation pressure	lb/in^2	/ENGINW/, 72, 80

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PCBBR	Specific engine thrust chamber nozzle stagnation pressure	lb/in^2	/ENGINW/, 72
PC	Thrust chamber nozzle stagnation pressure	lb/in^2	/ENGINO/, 72, 74, 75, 80, 83, 84, 85
PF	Fuel pump inlet pressure	lb/in^2	/ENGINO/, 32, 67
PIND	Propellant indicator	—	/ENGINW/, INTEGER, 32, 74, 83
PO	Oxidizer pump inlet pressure	lb/in^2	/ENGINO/, 31, 66, 68
PTF1	Initial constant fuel tank top pressure	lb/in^2	/ENGINW/, 47, 49, 50
PTF2	Final constant fuel tank top pressure	lb/in^2	/ENGINW/, 49, 52
PTFG	Constant fuel tank top pressure	psig	/ENGINW/, 35
PTF	Fuel tank top pressure	lb/in^2	/ENGINO/, 35, 47, 50, 52, 67
PTO1	Initial constant oxidizer tank top pressure	lb/in^2	/ENGINW/, 39, 41, 42
PTO2	Final constant oxidizer tank top pressure	lb/in^2	/ENGINW/, 41, 44
PTOG	Constant oxidizer tank top pressure	psig	/ENGINW/, 34
PTO	Oxidizer tank top pressure	lb/in^2	/ENGINO/, 34, 39, 42, 44, 66
PVF	Fuel vapor pressure	lb/in^2	32

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PVO	Oxidizer vapor pressure	lb/in^2	31
RESF	Hydraulic resistance of feed line from fuel tank bottom bulkhead to fuel pump inlet	sec^2/ft^5	/ENGINW/, 67
RESO	Hydraulic resistance of feed line from oxidizer tank bottom bulkhead to oxidizer pump inlet	sec^2/ft^5	/ENGINW/, 66
RHOF	Fuel density	lb/ft^3	/ENGINO/, 32, 57, 67, 115
RHOO	Oxidizer density	lb/ft^3	/ENGINO/, 31, 56, 66, 114
STIND	Stage indicator	—	/WORKD/, INTEGER, 13, 15, 19, 27, 29, 68, 69, 114, 115, 116
TBDOT	Derivative of burning time	—	/GILL/, 5, 11, 17, 22, 112
TBLDUP	Time to reach steady-state thrust	sec	/ENGINW/, 101
TEND	Time to shut down	sec	/ENGINW/, 104
TFI	Initial fuel temperature	^oR	/ENGINW/, 30
TF	Fuel temperature	^oR	/ENGINO/, 30, 32
THIND	Thrust indicator	—	/WORKD/, INTEGER, 1, 23, 105
THRST	Total thrust	lb	/DATA/, 3, 9, 13, 14, 19, 20, 21, 25, 106, 108
TIME	Range time	sec	/GILL/, 101, 102, 104

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TOI	Initial oxidizer temperature	^{o}R	/ENGINW/, 28
TO	Oxidizer temperature	^{o}R	/ENGINO/, 28, 31
TREF	Fuel reference temperature	^{o}F	/ENGINW/, 32
TSF1	Stage time at which constant fuel tank top pressure is stepped	sec	/ENGINW/, 46, 49
TSF2	Stage time at which final fuel tank top pressure is reached	sec	/ENGINW/, 45, 49
TSF	Ratio of fuel pressure stepping time	—	49, 50
TSO1	Stage time at which constant oxidizer tank top pressure is stepped	sec	/ENGINW/, 38, 41
TSO2	Stage time at which final oxidizer tank top pressure is reached	sec	/ENGINW/, 37, 41
TSO	Ratio of oxidizer pressure stepping time	—	40, 41
TSTAG	Stage time	sec	/DATA/, 37, 38, 41, 45, 46, 49
TSUBS	Section start time	sec	/SAVE/, 102
TTRUS	Array of functions, time vs. thrust table	—	/TABLES/, 13, 19
TTTRS	Array of arguments, time vs. thrust table	—	/TABLES/, 13, 19

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TTWDT	Array of arguments, time vs. flow rate table	—	/TABLES/, 15
TWDOT	Array of functions, time vs. flow rate table	—	/TABLES/, 15
VFBB	Fuel volume below bulkhead	ft^3	/ENGINW/, 57
VFCOFF	Fuel cut-off volume	ft^3	/ENGINW/, 115
VF	Fuel volume above bulkhead	ft^3	/ENGINO/, 57, 61, 62, 115
VOBB	Oxidizer volume below bulkhead	ft^3	/ENGINW/, 56
VOCOFF	Oxidizer cut-off volume	ft^3	/ENGINW/, 114
VO	Oxidizer volume above bulkhead	ft^3	/ENGINO/, 56, 58, 59, 114
W	Vehicle weight	lb	/GILL/, 25
WDOT	Total vehicle flow rate	lb/sec	/GILL/, 8
WDOTV	Vacuum flow rate	lb/sec	/WORKD/, 10
WEDOT	Total engine flow rate	lb/sec	/ENGINO/, 90, 91
WFCBB	Specific engine thrust chamber fuel flow rate	lb/sec	/ENGINW/, 71
WFCBR	Nominal engine thrust chamber fuel flow rate	lb/sec	/ENGINW/, 71, 78
WFCDT	Thrust chamber fuel flow rate	lb/sec	/ENGINO/, 71, 73, 75, 78, 79, 88

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WFDOT	Total fuel flow rate	lb/sec	/GILL/, 7, 111
WFEDT	Engine fuel flow rate	lb/sec	/ENGINO/, 88, 89, 90, 93
WFGBB	Specific engine gas generator fuel flow rate	lb/sec	/ENGINW/, 82
WFGBR	Nominal engine gas generator fuel flow rate	lb/sec	/ENGINW/, 82
WFGDT	Gas generator fuel flow rate	lb/sec	/ENGINO/, 82, 88
WFIS	Initial stage fuel weight	lb	/WORKD/, 29, 116
WFPDT	Pump fuel flow rate	lb/sec	/ENGINO/, 67, 93, 95
WF	Fuel weight	lb	/GILL/, 29, 57, 116
WFR	Fuel remaining	lb	/SAVE/, 115
WFSDT	Stage fuel flow rate	lb/sec	/ENGINO/, 95, 96, 97, 111
WKPC	Array of thrust chamber nozzle stagnation pressure influence coefficients	—	/ENGINW/, 72, 80
WKWFC	Array of thrust chamber fuel flow rate influence coefficients	—	/ENGINW/, 71, 78
WKWFG	Array of gas generator fuel flow rate influence coefficients	—	/ENGINW/, 82
WKWOC	Array of thrust chamber oxidizer flow rate influence coefficients	—	/ENGINW/, 70, 77

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WKWOG	Array of gas generator oxidizer flow rate influence coefficients	—	/ENGINW/, 81
WMFED	Miscellaneous engine fuel flow rate	lb/sec	/ENGINW/, 93
WMFSD	Miscellaneous stage fuel flow rate	lb/sec	/ENGINW/, 95
WMOED	Miscellaneous engine oxidizer flow rate	lb/sec	/ENGINW/, 92
WMOSD	Miscellaneous stage oxidizer flow rate	lb/sec	/ENGINW/, 94
WMSDT	Stage miscellaneous flow rate	lb/sec	/ENGINW/, 97
WOCBB	Specific engine thrust chamber oxidizer flow rate	lb/sec	/ENGINW/, 70
WOCBR	Nominal engine thrust chamber oxidizer flow rate	lb/sec	/ENGINW/, 70, 77
WOCDT	Thrust chamber oxidizer flow rate	lb/sec	/ENGINO/, 70, 73, 75, 77, 79, 87
WODOT	Total vehicle oxidizer flow rate	lb/sec	/GILL/, 6, 110
WOEDT	Engine oxidizer flow rate	lb/sec	/ENGINO/, 87, 89, 90, 92
WOGBB	Specific engine gas generator oxidizer flow rate	lb/sec	/ENGINW/, 81

SUBROUTINE ENGINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WOGBR	Nominal engine gas generator oxidizer flow rate	lb/sec	/ENGINW/, 81
WOGDT	Gas generator oxidizer flow rate	lb/sec	/ENGINO/, 81, 87
WOIS	Initial stage oxidizer weight	lb	/WORKD/, 27, 116
WOPDT	Pump oxidizer flow rate	lb/sec	/ENGINO/, 66, 92, 94
WO	Oxidizer weight	lb	/GILL/, 27, 56, 116
WOR	Oxidizer remaining	lb	/SAVE/, 114
WOSDT	Stage oxidizer flow rate	lb/sec	/ENGINO/, 94, 96, 97 110
WPDOT	Total non influence coefficient flow rate	lb/sec	/GILL/, 4, 8, 10, 15, 16, 21, 107, 109
WSDOT	Total stage flow rate	lb/sec	/ENGINO/, 97, 99, 109
XBAR	Input nominal independent variables	—	/ENGINI/, 68, 69
Z	Dummy argument in Subroutine PRPDEN	—	13, 15, 19, 31

```
$IBFTC ENGINE  FULIST,REF
      SUBRØUTINE ENGINE
C           ENGINE CØMPUTES MISSILE THRUSTS AND FLØW RATES
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $              TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL  INFIN
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $              PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $              PADSK,VESK,PADRT1,PADRT2,
     $             SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FRAC,[illegible],EPSIS,
     $              ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(6)
      REAL  ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $              AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $              WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $           W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $           DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $           WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $              PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $              GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $              QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $              CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $              ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN/TABLES/TMACH(30,6),TCSBN(30,6),TCSBA(30,6),
     $              TTPIT(30,6),TPIT(30,6),TTYAW(30,6),TYAW(30,6),
     $              TTTRS(30,6),TTRUS(30,6),TTWDT(30,6),TWDØT(30,6),
     $              JCSBN(6),JCSBA(6),JPIT(6),JYAW(6),JTRUS(6),JWDØT(6),
     $              LPIT(6),LYAW(6),LTRUS(6),LWDØT(6),
     $              KMACH(6),KPIT(6),KYAW(6),KTRUS(6),KWDØT(6),
     $              IMACH(6),IPIT(6),IYAW(6),ITRUS(6),IWDØT(6)
      CØMMØN/ENGIN1/ YBBAR(5,6),YBAR(5,6),XBAR(6,6),KPC(6,6),KWØC(6,6),
     $          KWFC(6,6),KWØG(6,6),KWFG(6,6),CTØX(6,6),CTFL(6,6),
     $          MISC(30,6),TANKS(30,6)
      REAL  KPC,KWØC,KWFC,KWØG,KWFG,MISC
      CØMMØN /ENGINØ/ PØ,PF,TØ,TF,CSTR,RHØØ,RHØF,PTØ,PTF,VØ,VF,HØ,HF,
     $           WØCDT,WFCDT,MRC,PC,WØGDT,WFGDT,CVTH,CFTH,CSUBF,FCHMBR,
     $           FENGIN,WØEDT,WFEDT,MRE,WEDØT,ISPE,WØPDT,WFPDT,
     $           FS,WØSDT,WFSDT,MRS,WSDØT,ISPS
      REAL  MRC,MRE,MRS,ISPE,ISPS
      CØMMØN/ENGINW/PCBBR,WØCBB,WFCBB,WØGBB,WFGBB,
     $              PCBAR,WØCBR,WFCBR,WØGBR,WFGBR,
     $              PØNØM,PFNØM,TØNØM,TFNØM,FØPNØM,ETANØM,
     $              PIND,TREF,DREF,FPTPA,PTØG,PTFG,TSØ1,PTØ1,TSØ2,PTØ2,
     $              C1Ø,C2Ø,TSF1,PTF1,TSF2,PTF2,C1F,C2F,
     $              LØ,RØ,UHØ,LHØ,LF,RF,UHF,LHF(5),
```

FIGURE 24. SYMBOLIC LISTING OF SUBROUTINE ENGINE

```
     $                     NØFE,FTURB,FØPT,KURF,DTBUP,VØCØFF,VFCØFF,EPVØL,
     $                     DTDCY,WMSDT,WMØSD,WMFSD,WMØED,WMFED,RESØ,RESF,
     $                     ASUBT,EPSI,ETACF,ETACS,TØI,CTØ,TFI,CTF,
     $                     VØBB,VFBB,HØCØL,HFCØI,PARANL,VPARL,
     $                     WKPC(6),WKWØC(6),WKWFC(6),WKWØG(6),WKWFG(6),
     $                     TBLDUP,TEND,FLGHD,FLGSD,LSTAGE,CSTRN
      LØGICAL  FLGHD,FLGSD
      REAL   LØ,LF,NØFE,KURF
      INTEGER  PIND,FPTPA,UHØ,UHF
      REAL   KLVL
      DIMENSIØN  PAR(6)
      INTEGER  ADD,TAG,DEC,PRE
      I = THIND + 1
      GØ TØ (10,20,30,35,40,40), I
C           CØAST
   10 THRST = 0.0
      WPDØT = 0.0
      TBDØT = 0.0
   11 WØDØT = 0.0
      WFDØT = 0.0
      WDØT = WPDØT
      RETURN
C           ANALYTIC
   20 THRST = FVAC - FPSIS * ASUBTS * PATM
      WPDØT = -WDØTV
      TBDØT = 1.0
      GØ TØ 11
C           TABULAR
   30 CALL NTERP(TTTRS(1,STIND),TTRUS(1,STIND),ITRUS(STIND),
     $              CLA(ITRUS(STIND)),THRST,Z,JTRUS(STIND),KTRUS(STIND),L)
      THRST = THRST * KSUBF
      CALL NTERP(TTWDT(1,STIND),TWDØT(1,STIND),IWDØT(STIND),
     $              CLA(LWDØT(STIND)),WPDØT,Z,JWDØT(STIND),KWDØT(STIND),L)
      WPDØT = -WPDØT * KSUBW
      TBDØT = 1.0
      GØ TØ 11
C           TABULAR VACUUM THRUST,  CØNSTANT ISP
   35 CALL NTERP(TTTRS(1,STIND),TTRUS(1,STIND),ITRUS(STIND),
     $         CLA(LTRUS(STIND)),THRST,Z,JTRUS(STIND),KTRUS(STIND),L)
      THRST = THRST * KSUBF - EPSIS * ASUBTS * PATM
      WPDØT = -THRST/ISP
      TBDØT = 1.0
      IF (THIND .EQ. 5)  GØ TØ 50
      GØ TØ 11
C           LIQUID ENGINE INFLUENCE CØEFFICIENTS
   40 ACCEL = (THRST - DRAG)/W
      IF (FLGHD)  ACCEL = 0.0
C           CØMPUTE PUMP INLET TEMPERATURES
      CALL VPØLY(6,CTØX(1,STIND),WØ/WØIS,DELTØ)
      TØ = TØI + CTØ * DELTØ
      CALL VPØLY(6,CTFL(1,STIND),WF/WFIS,DELTF)
```

FIGURE 24. SYMBOLIC LISTING OF SUBROUTINE ENGINE (cont.)

```
      TF = TFI + CTF * DELTF
C           CØMPUTE PUMP INLET DENSITIES
      CALL PRPDEN(1,PØ,TØ-459.688,Z,Z,PVØ,RHØØ)
      CALL PRPDEN(PIND+1,PF,TF-459.688,TREF,DREF,PVF,RHØF)
C           CØMPUTE TANK PRESSURES
      IF (FPTPA .NE. 0)  GØ TØ 41
      PTØ = PTØG + PATM
      PTF = PTFG + PATM
      GØ TØ 47
   41 IF (TSTAG .GE. TSØ2)  GØ TØ 43
      IF (TSTAG .GT. TSØ1)  GØ TØ 42
      PTØ = PTØ1
      GØ TØ 44
   42 TSØ = (TSTAG - TSØ1) * (PTØ2 - PTØ1)/(TSØ2 - TSØ1)
      PTØ = PTØ1 + C1Ø * TSØ + C2Ø * TSØ**2
      GØ TØ 44
   43 PTØ = PTØ2
   44 IF (TSTAG .GE. TSF2)  GØ TØ 46
      IF (TSTAG .GT. TSF1)  GØ TØ 45
      PTF = PTF1
      GØ TØ 47
   45 TSF = (TSTAG - TSF1) * (PTF2 - PTF1)/(TSF2 - TSF1)
      PTF = PTF1 + C1F * TSF + C2F * TSF**2
      GØ TØ 47
   46 PTF = PTF2
   47 CØNTINUE
C           CØMPUTE TANK VØLUMES AND HEIGHTS
      HØ = 0.0
      HF = 0.0
      VØ = WØ/RHØØ - VØBB
      VF = WF/RHØF - VFBB
      IF (VØ .LT. 0.0)  GØ TØ 471
      CALL HITE(VØ,LØ,HØ)
  471 CØNTINUE
      IF (VF .LT. 0.0)  GØ TØ 472
      CALL HITE(VF,LF,HF)
  472 CØNTINUE
      HØ = HØ + HØCØL
      HF = HF + HFCØL
C           CØMPUTE PUMP INLET PRESSURES
      PØ = PTØ + RHØØ * ACCEL * HØ/144.0 - RESØ * WØPDT**2/
     $                                                  (144.0 * RHØØ)
      PF = PTF + RHØF * ACCEL * HF/144.0 - RESF * WFPDT**2/
     $                                                  (144.0 * RHØF)
C           CØMPUTE VARIENCE ØF FIRST FØUR INDEPENDENT VARIABLES
      CALL VARYD(PØ,PØNØM,PAR,4)
      PAR(5) = KURF * (FØPT - FØPNØM)/FØPNØM
C           CØMPUTE THRUST CHAMBER FLØW RATES - NØ C* CØRRECTIØN
      CALL GICEQ(WØCDT,WØCBB,WØCBR,WKWØC,PAR,5)
      CALL GICEQ(WFCDT,WFCBB,WFCBR,WKWFC,PAR,5)
```

FIGURE 24. SYMBOLIC LISTING OF SUBROUTINE ENGINE (cont.)

```
C           CØMPUTE C* CØRRECTIØN FACTØR
      CALL GICEQ(PC,PCBBR,PCBAR,WKPC,PAR,5)
      MRC = WØCDT/WFCDT
      CALL TCPRØP(PIND,MRC,PC,EPSI,CVTH,CFTH)
      CSTR = PC * ASUBT * GSUB1/(WØCDT + WFCDT)
      PAR(6) = (CVTH * ETACS - CSTR)/CSTRN
C           CØRRECT CHAMBER FLØW RATES AND MIXTURE RATIØ
      WØCDT = WØCDT + WØCBR * WKWØC(6) * PAR(6)
      WFCDT = WFCDT + WFCBR * WKWFC(6) * PAR(6)
      MRC = WØCDT/WFCDT
C           CØMPUTE CHAMBER PRESSURE AND GAS GENERATØR FLØW RATES
      PC = PC + PCBAR * WKPC(6) * PAR(6)
      CALL GICEQ(WØGDT,WØGBB,WØGBR,WKWØG,PAR,6)
      CALL GICEQ(WFGDT,WFGBB,WFGBR,WKWFG,PAR,6)
C           CØMPUTE THEØRETICAL AND ACTUAL THRUST CØEFFICIENTS
      CALL TCPRØP(PIND,MRC,PC,EPSI,CVTH,CFTH)
      CSUBF = ETACF * CFTH - EPSI * PATM/PC
C           CØMPUTE CHAMBER AND ENGINE THRUSTS
      FCHMBR = CSUBF * PC * ASUBT
      FENGIN = FCHMBR + FTURB
C           CØMPUTE ENGINE FLØW RATES AND MIXTURE RATIØ
      WØEDT = WØCDT + WØGDT
      WFEDT = WFCDT + WFGDT
      MRE = WØEDT/WFEDT
      WEDØT = WØEDT + WFEDT
C
      ISPE = FENGIN/WEDØT
C
      WØPDT = WØEDT + WMØED
      WFPDT = WFEDT + WMFED
C           CØMPUTE TØTAL STAGE THRUSTS,FLØW RATES, ETC.
      WØSDT = NØFE * WØPDT + WMØSD
      WFSDT = NØFE * WFPDT + WMFSD
      MRS = WØSDT/WFSDT
      WSDØT = WØSDT + WFSDT + WMSDT
      FS = NØFE * FENGIN
      ISPS = FS/WSDØT
C
      KLVL = 1.0
      IF (TIME .GE. TBIDUP)  GØ TØ 48
      KLVL = (TIME - TSUBS)/DTBUP
      GØ TØ 49
C           CHECK FØR SHUT-DØWN
   48 IF (FLGSD)  KLVL = (TEND - TIME)/DTDCY
C
   49 IF (THIND .EQ. 5)  GØ TØ 35
      THRST = 0.0
      WPDØT = 0.0
   50 THRST = FS * KLVL + THRST
      WDØT = -WSDØT * KLVL + WPDØT
```

FIGURE 24. SYMBOLIC LISTING OF SUBROUTINE ENGINE (cont.)

```
WØDØT = -WØSDT * KLVL

WFDØT = -WFSDT * KLVL
TBDØT = 1.0
IF (FLGSD)  RETURN
WØR(STIND) = (VØ - VØCØFF) * RHØØ
WFR(STIND) = (VF - VFCØFF) * RHØF
AVMR(STIND) = (WØIS - WØ)/(WFIS - WF)
RETURN
END
```

FIGURE 24. SYMBOLIC LISTING OF SUBROUTINE ENGINE (cont.)

XXI. SUBROUTINE ENGSU

This subroutine sets up the working data - /ENGINW/ -, for Subroutine ENGINE at the beginning of each stage.

CALLING SEQUENCE

CALL ENGSU

This subroutine uses COMMON blocks CONST, INPUT, WORKD, SAVE, ENGINI, ENGINW, STAGES, INFLTB, ENGINO and DATA.

METHOD

```
1.  If STIND = LSTAGE, return

    If STIND ≠ LSTAGE, go to 2

2.  LSTAGE = STIND

Set up nominal and specific engine dependent variables:

3.  Do 5,  I = 1, 5

4.  YBBARW(I) = YBBAR(I, STIND)

5.  YBARW(I) = YBAR(I, STIND)
```

Set up tank and miscellaneous data:

```
6.  Do 8, I = 1, 30
7.  TANKD(I) = TANKS(I, STIND)
8.  MISCD(I) = MISC(I, STIND)
```

Test if influence coefficients were generated by Block I:

```
9.  If ISTG(STIND) > 0 and ISTG(STIND) < 5, go to 19
    Otherwise, go to 10
10. Do 15, I = 1, 6
11. WKPC(I) = KPC(I, STIND)
12. WKWOC(I) = KWOC(I, STIND)
13. WKWFC(I) = KWFC(I, STIND)
14. WKWOG(I) = KWOG(I, STIND)
15. WKWFG(I) = KWFG(I, STIND)
```

Set up nominal independent variables:

```
16. Do 17, I = 1, 6
17. XBARW(I) = XBAR(I, STIND)
18. Go to 49
```

Set up influence coefficients and nominal values generated by Block I:

```
19. Do 24, I = 1, 4
20. WKPC(I) = YY(I, 5, STIND)
```

```
21.  WKWOC(I) = YY(I, 1, STIND)
22.  WKWFC(I) = YY(I, 2, STIND)
23.  WKWOG(I) = YY(I, 3, STIND)
24.  WKWFG(I) = YY(I, 4, STIND)
25.  WKPC(5) = YY(6, 5, STIND)
26.  WKPC(6) = YY(5, 5, STIND)
27.  WKWOC(5) = YY(6, 1, STIND)
28.  WKWOC(6) = YY(5, 1, STIND)
29.  WKWFC(5) = YY(6, 2, STIND)
30.  WKWFC(6) = YY(5, 2, STIND)
31.  WKWOG(5) = YY(6, 3, STIND)
32.  WKWOG(6) = YY(5, 3, STIND)
33.  WKWFG(5) = YY(6, 4, STIND)
34.  WKWFG(6) = YY(5, 4, STIND)
35.  YBBARW(1) = VDEPA(5, STIND)
36.  YBBARW(2) = VDEPA(1, STIND)
37.  YBBARW(3) = VDEPA(2, STIND)
38.  YBBARW(4) = VDEPA(3, STIND)
39.  YBBARW(5) = VDEPA(4, STIND)
40.  Do 41, I = 1, 5
41.  YBARW(I) = YBBARW(I)
```

```
42.  Do 43, I = 1, 4
43.  XBARW(I) = VINDA(I, STIND)
44.  XBARW(5) = VINDA(6, STIND)
45.  XBARW(6) = VINDA(5, STIND)
46.  EPSI = EAE(1, STIND)
47.  ASUBT = EAE(2, STIND)
48.  ETACF = EAE(3, STIND)
49.  Continue
```

Initialize variables:

```
50.  TBLDUP = TSUBS + DTBUP
51.  TEND = INFIN
52.  FLGHD = .FALSE.
53.  FLGSD = .FALSE.
54.  PO = PONOM
55.  PF = PFNOM
56.  WOPDT = WOCBR
57.  WFPDT = WFCBR
58.  THRST = 0
```

Test for thrust buildup:

59. If DTBUP = 0, THRST = FOPNOM

If DTBUP $\neq$ 0, go to 60

Compute nominal characteristic velocity:

60. CSTRN = ((PCBAR)(ASUBT)(GSUB1))/(WOCBR + WFCBR)

Return

SUBROUTINE ENGSU NOMENCLATURE

Symbol	Description	Units	Reference
ASUBT	Thrust chamber throat area	in^2	/ENGINW/, 47, 60
CSTRN	Nominal thrust chamber characteristic velocity	ft/sec	/ENGINW/, 60
DTBUP	Time interval for thrust buildup	sec	/ENGINW/, 50, 59
EAE	Array of certain independent variables generated in Block I	—	/INFLTB/, 46, 47, 48
EPSI	Expansion area ratio	—	/ENGINW/, 47
ETACF	Nozzle efficiency	—	/ENGINW/, 48
FLGHD	Flag for hold-down	—	/ENGINW/, LOGICAL, 52
FLGSD	Flag for shut-down	—	/ENGINW/, LOGICAL, 53
FOPNOM	Nominal engine thrust	lb	/ENGINW/, 59
GSUB1	Weight to mass conversion factor	lb/slug	/CONST/, 60
INFIN	"Infinity" = 10^{10}	—	/CONST/, REAL, 51
ISTG	Indicator for influence coefficients generated by Block I	—	/STAGES/, 9
KPC	Array of input chamber pressure influence coefficients	—	/ENGINI/, REAL, 11

SUBROUTINE ENGSU NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
KWFC	Array of input chamber fuel flow rate influence coefficients	—	/ENGINI/, REAL, 13
KWFG	Array of input gaseous fuel flow rate influence coefficients	—	/ENGINI/, REAL, 15
KWOC	Array of input chamber oxidizer flow rate influence coefficients	—	/ENGINI/, REAL, 12
KWOG	Array of input gaseous oxidizer flow rate influence coefficients	—	/ENGINI/, REAL, 14
LSTAGE	Last stage set-up indicator	—	/ENGINW/, 1, 2
MISCD	Array of working values of miscellaneous data	—	EQUIV, DIM, REAL, 8
MISC	Array of input values of miscellaneous data	—	/ENGINI/, 8
PCBAR	Chamber pressure nominal value	psia	/ENGINW/, EQUIV, 60
PFNOM	Nominal fuel pump inlet pressure	psia	/ENGINW/, 55
PF	Fuel pump inlet pressure	psia	/ENGINO/, 55
PONOM	Nominal oxidizer pump inlet pressure	psia	/ENGINW/, EQUIV, 54
PO	Oxidizer pump inlet pressure	psia	/ENGINO/, 54

SUBROUTINE ENGSU NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
STIND	Stage indicator	—	/WORKD/, INTEGER, 1, 2, 4, 5, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 43, 44, 45
TANKD	Array of working tank data	—	EQUIV, DIM, 7
TANKS	Array of input tank data	—	/ENGINI/, 7
TBLDUP	Time when full thrust is reached	sec	/ENGINW/, 50
TEND	Time when engine shutdown is complete	sec	/ENGINW/, 51
THRST	Stage thrust	lb	/DATA/, 58, 59
TSUBS	Time at start of section	sec	/SAVE/, 50
VDEPA	Array of dependent variables	—	/INFLTB/, 35, 36, 37, 38, 39
VINDA	Array of independent variables	—	/INFLTB/, 43, 44, 45
WFCBR	Nominal value of chamber fuel flow rate	lb/sec	/ENGINW/, 57, 60
WFPDT	Fuel pump flow rate	lb/sec	/ENGINO/, 57
WKPC	Array of working chamber pressure influence coefficients	—	/ENGINW/, 11, 20, 25, 26

SUBROUTINE ENGSU NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WKWFC	Array of working chamber fuel flow rate influence coefficients	—	/ENGINW/, 13, 22, 29, 30
WKWFG	Array of working gaseous fuel flow rate influence coefficients	—	/ENGINW/, 15, 24, 33, 34
WKWOC	Array of working chamber oxidizer flow rate influence coefficients	—	/ENGINW/, 12, 21, 27, 28
WKWOG	Array of working gaseous oxidizer flow rate influence coefficients	—	/ENGINW/, 14, 23, 31, 32
WOCBR	Nominal value of chamber oxidizer flow rate	lb/sec	/ENGINW/, 56, 60
WOPDT	Oxidizer pump flow rate	lb/sec	/ENGINO/, 56
XBAR	Array of independent variables	—	/ENGINI/, 17
XBARW	Working array of independent variables	—	EQUIV, DIM, 17, 43, 44, 45
YBAR	Nominal dependent variable input values	—	/ENGINI/, 5
YBARW	Nominal dependent variable working values	—	EQUIV, DIM, 5, 41
YBBAR	Specific engine dependent variable input values	—	/ENGINI/, 4

SUBROUTINE ENGSU NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
YBBARW	Specific engine dependent variable working values	—	EQUIV, DIM, 4, 35, 36, 37, 38, 39, 41
YY	Array of influence coefficients generated by Block I	—	/INFLTB/, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34

```
$IBFTC ENGSU    FULIST,REF
      SUBROUTINE ENGSU
      COMMON /CONST/RSUB0,FLAT,GSUB0,GSUB1,OMEGA,PSL,CMAX,HMIN,HMAX,
     $              TAGS(7),INFIN,PI,PIO2,DTRAD,DPRAD
      EQUIVALENCE   (RSUB0,RSUBO),(GSUBO,GSUB0)
      REAL   INFIN
      COMMON /INPUT/ [illegible],LATI,LONGI,ELTAI,AZI,BETAO,VI,[illegible]PL,S[illegible]ETI,
     $              DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $              SECTS(20,12),STAGS(20,6),HEAD(11,15)
      REAL   LATI,LONGI
      INTEGER   TARGET
      INTEGER   FLTR,FLSC,FLPRT
      COMMON/WORKD/PARAM,VE,WIOWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $              [illegible]DOT,YDOT,PKIK,YKIK,VGX,VGY,VGZ,SECTNO,
     $              [illegible],VEGK,PADRT1,PADRT2,
     $           [illegible],WFIS,WPRPS,WBOS,FVAC,[illegible],EPS[illegible],
     $            [illegible],KSUB[illegible],KSUBP,KSUBY,KSUB[illegible],KSUBA([illegible])
      REAL   [illegible],KSUBY,KSUBN,KSUBA
      INTEGER   STIND,THIND,ATIND,SECTNO
      COMMON /SAVE/JD,[illegible],PHIO,SPHIO,CPHIO,THETAO,STH[illegible],CTHET[illegible],
     $              AZIR,ROX,ROY,ROZ,TSTG,TSUBS,PSUBS,YSUB[illegible],
     $              [illegible]OR(6),[illegible]FR(6),AVMR(6)
      DOUBLE PRECISION   JD
      COMMON /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $              [illegible],THETA,[illegible],VSUBI,GAMIN,AZIN,[illegible],AZ,BETA,
     $              [illegible],PRATE,[illegible],
     $              [illegible],ALPHAY,ALPHA,ALPHAD,[illegible],
     $              [illegible],THRST,GRAV,[illegible],RANGE,
     $              [illegible],ESUB[illegible],ERATIO
      REAL   LAT,[illegible],MACH,LIFT
      COMMON /[illegible]/ [illegible]BAR(5,6),YBAR(5,6),XBAR(6,6),KPC(6,6),KWOC(6,6),
     $           [illegible]FC(6,6),KWOG(6,6),KWFG(6,6),CTOX(6,6),CTFI(6,6),
     $           MISC(30,6),TANKS(30,6)
      REAL   [illegible],KWOC,[illegible]FC,KWOG,KWFG,MISC
      COMMON /[illegible]/[illegible],WOCBB,WFCBB,WOGBB,WFGBB,
     $              [illegible]CBAR,WOCBR,WFCBR,WOGBR,WFGBR,
     $              [illegible]ONOM,PFNOM,TONOM,TFNOM,FOPNOM,ETANOM,
     $              PIND,TREF,DREF,FPTPA,PTOG,PTFG,TSO1,PTO1,TSO2,PTO2,
     $              C1O,C2O,TSF1,PTF1,TSF2,PTF2,C1F,C2F,
     $              LO,RO,UHO,LHO,LF,RF,UHF,LHF(5),
     $              NOFE,FTURB,FOPT,KURF,DTBUP,VOCOFF,VFCOFF,EPVOL,
     $              DTDCY,WMSDT,WMOSD,WMFSD,WMOED,WMFED,RESO,RESF,
     $              ASUBT,EPSI,ETACF,ETACS,TOI,CTO,TFI,CTF,
     $              VOBB,VFBB,HOCOL,HFCOL,PARAML,VPARL,
     $              WKPC(6),WKWOC(6),WKWFC(6),WKWOG(6),[illegible](6),
     $              TBLDUP,TEND,FLGHD,FLGSD,LSTAGE,CSTRN
      LOGICAL   FLGHD,FLGSD
      INTEGER   PIND,FPTPA,UHO,UHF
      REAL   LO,LF,NOFE,KURF
      COMMON /ENGINO/ PO,PF,TO,TF,CSTR,RHOO,RHOF,PTO,PTF,VO,VF,HO,HF,
     $            WOCDT,WFCDT,MRC,PC,WOGDT,WFGDT,CVTH,CFTH,CSUBF,FCHMBR,
     $            FENGIN,WOEDT,WFEDT,MRE,WEDOT,ISPE,WOPDT,WFPDT,
```

FIGURE 25. SYMBOLIC LISTING OF SUBROUTINE ENGSU

```
     $             FS,WØSDT,WFSDT,MRS,WSDØT,ISPS
      REAL  MRC,MRE,MRS,ISPE,ISPS
      CØMMØN /STAGES/ISTG(6)
      CØMMØN /INFLTB/ YY(10,10,6),VINDA(10,6),VDEPA(10,6),NINDA(6),NIND,
     1NDA(6),Y(10,10),VDEP(10),VIND(10),ND ,EAE(3,6)
      REAL  MISCD
      DIMENSIØN  YBBARW(5),YBARW(5),XBARW(6),TANKD(30),MISCD(30)
      EQUIVALENCE  (YBBARW(1),PCBBR),(YBARW(1),PCBAR),(XBARW(1),PØNØM),
     $              (TANKD(1),PIND),(MISCD(1),NØFE)
C
      IF (STIND .EQ. LSTAGE)  RETURN
      LSTAGE = STIND
      DØ 10  I=1,5
      YBBARW(I) = YBBAR(I,STIND)
   10 YBARW(I) = YBAR(I,STIND)
C          MØVE TANK DATA
C          MØVE MISC.  DATA
      DØ 20  I=1,30
      TANKD(I) = TANKS(I,STIND)
   20 MISCD(I) = MISC(I,STIND)
C          MØVE INFLUENCE CØEFFICIENTS
      IF (ISTG(STIND) .GT. 0 .AND. ISTG(STIND) .LT. 5)  GØ TØ 55
      DØ 50  I=1,6
      WKPC(I) = KPC(I,STIND)
      WKWØC(I) = KWØC(I,STIND)
      WKWFC(I) = KWFC(I,STIND)
      WKWØG(I) = KWØG(I,STIND)
   50 WKWFG(I) = KWFG(I,STIND)
      DØ 51  I=1,6
   51 XBARW(I) = XBAR(I,STIND)
      GØ TØ 65
   55 DØ 60  I=1,4
      WKPC(I) = YY(I,5,STIND)
      WKWØC(I) = YY(I,1,STIND)
      WKWFC(I) = YY(I,2,STIND)
      WKWØG(I) = YY(I,3,STIND)
   60 WKWFG(I) = YY(I,4,STIND)
      WKPC(5) = YY(6,5,STIND)
      WKPC(6) = YY(5,5,STIND)
      WKWØC(5) = YY(6,1,STIND)
      WKWØC(6) = YY(5,1,STIND)
      WKWFC(5) = YY(6,2,STIND)
      WKWFC(6) = YY(5,2,STIND)
      WKWØG(5) = YY(6,3,STIND)
      WKWØG(6) = YY(5,3,STIND)
      WKWFG(5) = YY(6,4,STIND)
      WKWFG(6) = YY(5,4,STIND)
      YBBARW(1) = VDEPA(5,STIND)
      YBBARW(2) = VDEPA(1,STIND)
      YBBARW(3) = VDEPA(2,STIND)
```

FIGURE 25. SYMBOLIC LISTING OF SUBROUTINE ENGSU (cont.)

```
      YBBARW(4) = VDEPA(3,STIND)
      YBBARW(5) = VDEPA(4,STIND)
      DØ 61  I=1,5
   61 YBARW(I) = YBBARW(I)
      DØ 62  I=1,4
   62 XBARW(I) = VINDA(I,STIND)
      XBARW(5) = VINDA(6,STIND)
      XBARW(6) = VINDA(5,STIND)
      EPSI = EAE(1,STIND)
      ASUBT = EAE(2,STIND)
      ETACF = EAE(3,STIND)
   65 CØNTINUE
C
      TBLDUP = TSUBS + DTBUP
      TEND = INFIN
      FLGHD = .FALSE.
      FLGSD = .FALSE.
C
      PØ = PØNØM
      PF = PFNØM
      WØPDT = WØCBR
      WFPDT = WFCBR
      THRST = 0.0
      IF (DTBUP .EQ. 0.0)  THRST = FØPNØM
      CSTRN = PCBAR * ASUBT * GSUB1/(WØCBR + WFCBR)
C
      RETURN
      END
```

FIGURE 25. SYMBOLIC LISTING OF SUBROUTINE ENGSU (cont.)

XXII. SUBROUTINE VPOLY

This subroutine evaluates a polynomial of the form:

$$Y = C_1 + C_2X + C_3X^2 + \ldots . C_mX^{m-1}$$

CALLING SEQUENCE

The calling sequence is:

CALL VPOLY (M, C, X, Y)

C is an array containing the coefficients of the equation.

SOLUTION METHOD

Following is the appropriate set of equations:

```
1.  Y = C(1)
2.  XP = X
3.  Do 5, I = 2, M
4.  Y = Y + (C(I))(XP)
5.  XP = (XP)(X)
    Return
```

SUBROUTINE VPOLY NOMENCLATURE

Symbol	Description	Units	Reference
C	Table of coefficients	—	CALL, DIM, 1, 4
M	Table dimension	—	CALL, DIM, 3
X	Value of independent variable	—	CALL, 2, 5
XP	Successive powers of X	—	2, 4, 5
Y	Value of polynomial	—	CALL, 1, 4

```
$IBFTC VPØLY    FULIST,REF
      SUBRØUTINE VPØLY(M,C,X,Y)
C          M = DEGREE ØF PØLYNØMIAL + 1
C          C = VECTØR ØF CØEFFICIENTS ØF DIMENSIØN M
C          X = VALUE TØ USE TØ EVALUATE PØLYNØMIAL
C          Y = VALUE ØF PØLYNØMIAL
C               VPØLY EVALUATES A PØLYNØMIAL ØF THE FØRM
C                    Y = C(1) + C(2)*X + C(3)*X**2 ... + C(M)*X**(M-1)
      DIMENSIØN  C(M)
      Y = C(1)
      XP = X
      DØ 10  I=2,M
      Y = Y + C(I)*XP
   10 XP = XP * X
      RETURN
      END
```

FIGURE **26**. SYMBOLIC LISTING OF SUBROUTINE VPOLY

XXIII. SUBROUTINE VARYD

This subroutine computes the quantity $(X_i - \overline{X}_i)/(\overline{X}_i)$ which is used in engine influence coefficient calculations.

CALLING SEQUENCE

The calling sequence is:

CALL VARYD (XLIST, XBAR, PARSHL, N)

XLIST is an array containing the instantaneous values of the independent variables, X_i. XBAR is an array containing the nominal values of the variables, $\overline{X}_i$.

SOLUTION METHOD

Following is the appropriate set of equations:

```
1.  Do 2, I = 1, N
2.  PARSHL(I) = (XLIST(I) - XBAR(I))/(XBAR(I))
    Return
```

SUBROUTINE VARYD NOMENCLATURE

Symbol	Description	Units	Reference
N	Table dimension	—	CALL, 1
PARSHL	Table of $(X_i - \overline{X}_i)/(\overline{X}_i)$ values	—	CALL, DIM, 2
XBAR	Table of nominal values of independent variables	—	CALL, DIM, 2
XLIST	Table of instantaneous values of independent variables	—	CALL, DIM, 2

```
$IBFTC VARYD   FULIST,REF
      SUBROUTINE VARYD(XLIST,XBAR,PARSHL,N)
      DIMENSION XLIST(N),XBAR(N),PARSHL(N)
C         CALCULATES  (X - XBAR)/XBAR  FOR INFLUENCE COEFFICIENTS
      DO 10 I=1,N
   10 PARSHL(I) = (XLIST(I) - XBAR(I))/XBAR(I)
      RETURN
      END
```

FIGURE 27. SYMBOLIC LISTING OF SUBROUTINE VARYD

XXIV. SUBROUTINE GICEQ

This subroutine computes the value of the general influence coefficient equation of form:

$$Y_j = \bar{\bar{Y}}_j + \bar{Y}_j \sum K_{ij} \left(\frac{X_i - \bar{X}_i}{\bar{X}_i} \right)$$

CALLING SEQUENCE

The calling sequence is:

CALL GICEQ (Y, YBRBR, YBAR, COEFS, PARSHL, N)

COEFS is an array containing the non-dimensional influence coefficients, K_{ij} . PARSHL is an array containing the quantities $(X_i - \bar{X}_i)/(\bar{X}_i)$. The values of PARSHL come from Subroutine VARYD.

SOLUTION METHOD

Following is the appropriate set of equations:

```
1.  Y = 0
2.  Do 3, I = 1, N
3.  Y = Y + (COEFS(I))(PARSHL(I))
4.  Y = YBRBR + (YBAR)(Y)
    Return
```

SUBROUTINE GICEQ NOMENCLATURE

Symbol	Description	Units	Reference
COEFS	Table of non-dimensional influence coefficients	—	CALL, DIM, 3
N	Table dimension	—	CALL, 2
PARSHL	Table of $(X_i - \overline{X}_i)/(\overline{X}_i)$ values	—	CALL, DIM, 3
Y	Instantaneous value of dependent variable	—	CALL, 1, 3, 4
YBAR	Nominal value of dependent variable Y	—	CALL, 4
YBRBR	Specific engine value of dependent variable Y	—	CALL, 4

```
$IBFTC GICEQ   FULIST,REF
      SUBROUTINE GICEQ(Y,YBRBR,YBAR,COEFS,PARSHL,N)
      DIMENSION  COEFS(N),PARSHL(N)
C         CALCULATES GENERAL INFLUENCE COEFFICIENT EQUATION
C              Y = YBRBR + YBAR*(COEFS(I) * PARSHL(I),(I=1,N))
      Y =  0.0
      DO 10  I=1,N
   10 Y =  Y + COEFS(I) * PARSHL(I)
      Y = YBRBR + YBAR * Y
      RETURN
      END
```

FIGURE 28. SYMBOLIC LISTING OF SUBROUTINE GICEQ

XXV. SUBROUTINE INTEG

This subroutine provides a step-by-step procedure for the integration of first-order differential equations such as is used, for example, in trajectory analysis. The technique is based upon the Runge-Kutta integration analysis, as modified by the method of S. Gill (Ref. [1]).

The general computational method is as follows.

Given the system of M equations:

$$dy_i / dx = f_i(x , y_1 , y_2 , \ldots y_m) , \quad (i = 1, 2, \ldots m) ,$$

and the values of x , y_1 , y_2 , ... y_m for $x = x_o$, the auxiliary routine DERIV computes the values, x, y_1 , y_2 , ... y_m for $x = x_o + h$.

The determination of the values of the dependent variables requires four passes through the program. The equations used involve four coefficients, which are internally stored in a DATA statement as double precision constants. At the end of the fourth pass, and before control is returned to the main program, INTEG makes an additional pass through the auxiliary routine DERIV to compute the derivatives of the dependent variables for the final value of the independent variable,

$$f_i(x_o + h , y_{14} , y_{24} , \ldots y_{m4}) .$$

CALLING SEQUENCE

The calling sequence is:

CALL INTEG (DERIV, N, V, DER, Q, K)

In the above listing V, DER and Q are arrays with the following properties:

a. V is an array of dimension (2, N). The first element is the independent variable; the following N-1 elements are the M dependent variables. The variables are double precision within the integration routine.

b. DER is an array of dimension N. The first element is the integration interval for the independent variable. The following elements are the derivatives of the dependent variables.

c. Q is an array of dimension N, where the Q's are double precision summing variables.

The following double precision constants are internally stored in a DATA statement:

$B1 = 2 - \sqrt{2} = 0.585786438$

$B2 = 2 + \sqrt{2} = 3.414213562$

$C1 = -2 + 3/\sqrt{2} = 0.121320344$

$C2 = 2(-2 - 3/\sqrt{2}) = -8.242640688$

SOLUTION METHOD

The following set of equations is established:

```
1.   Q(1) = (DER(1))/(2)

2.   K = 1

3.   If K = 1, go to 4

     If K = 2, go to 9

     If K = 3, go to 13

     If K = 4, go to 18

First pass:

4.   Do 6, I = 2, N

5.   Q(I) = (DER(I))(Q(1))

6.   V(I) = V(I) + Q(I)

7.   V(1) = V(1) + Q(1)

8.   Go to 20

Second pass:

9.   Do 11, I = 2, N

10.  V(I) = V(I) + (B1)((DER(I))(Q(1)) - Q(I))

11.  Q(I) + (B1)(DER(I))(Q(1)) + (C1)(Q(I))

12.  Go to 20
```

Third pass:

```
13.  Do 15, I = 2, N
14.  V(I) = V(I) + (B2)((DER(I))(Q(1)) - Q(I))
15.  Q(I) = (2)(B2)(DER(I))(Q(1)) + (C2)(Q(I))
16.  V(1) = V(1) + Q(1)
17.  Go to 20
```

Fourth pass:

```
18.  Do 19, I = 2, N
19.  V(I) = V(I) + ((DER(I))(Q(1)) - Q(I))/(3)
20.  CALL DERIV
21.  K = K + 1
22.  If K ≤ 4, go to 3
     If K > 4, go to 23
23.  Return
```

SUBROUTINE INTEG NOMENCLATURE

Symbol	Description	Units	Reference
B1	Constant = 0.585786438	—	DATA, DP, 10, 11
B2	Constant = 3.414213562	—	DATA, DP, 14, 15
C1	Constant = 0.121320344	—	DATA, DP, 11
C2	Constant = -8.242640688	—	DATA, DP, 15
DER	Values for integration interval and derivatives	—	CALL, DIM, 1, 5, 10, 11, 14, 15, 19
DERIV	Auxiliary routine used to evaluate derivatives	—	CALL, 20
K	Pass counter, values 1 to 4	—	CALL, 2, 3, 21, 22
N	Total number of dependent variables + 1	—	CALL, DIM, 4, 9, 13, 18
Q	Summing variables	—	CALL, DIM, DP, 1, 5, 6, 7, 10, 11, 14, 15, 16, 19
V	Storage array for independent and dependent variables	—	CALL, DIM, DP, 6, 7, 10, 14, 16, 19

```
$IBFTC INTEG   REF,FULIST
      SUBROUTINE INTEG (DERIV,N,V,DER,Q,K)
      DOUBLE PRECISION  V(N),Q(N)
      DIMENSION  DER(N)
      DOUBLE PRECISION B1,B2,C1,C2
      DATA  B1,B2,C1,C2/.585786438D0,3.414213562,.121320344D0,
     $          -8.242640688/
      Q(1) = DER(1)/2.0
      K = 1
    5 GO TO (10,20,30,40),  K
C         FIRST PASS
   10  DO 11  I=2,N
      Q(I) = DER(I) * Q(1)
   11 V(I) = V(I) + Q(I)
      V(1) = V(1) + Q(1)
      GO TO 50
C         SECOND PASS
   20 DO 21  I=2,N
      V(I) = V(I) + B1 * (DER(I) * Q(1) - Q(I))
   21 Q(I) = B1 * DER(I) * Q(1) + C1 * Q(I)
      GO TO 50
C         THIRD PASS
   30 DO 31  I=2,N
      V(I) = V(I) + B2 * (DER(I) * Q(1) - Q(I))
   31 Q(I) = 2.0 * B2 * DER(I) * Q(1) + C2 * Q(I)
      V(1) = V(1) + Q(1)
      GO TO 50
C         FOURTH PASS
   40 DO 41  I=2,N
   41 V(I) = V(I) + (DER(I) * Q(1) - Q(I))/3.0
C
   50 CALL DERIV
      K = K + 1
      IF (K .LE. 4)  GO TO 5
      RETURN
      END
```

FIGURE **29**. SYMBOLIC LISTING OF SUBROUTINE INTEG

XXVI. SUBROUTINE NTERP

This subroutine will perform either a linear, quadratic or cubic interpolation by the use of Lagrange's interpolation formula.

Tables of the independent variable X and of the corresponding dependent variable Y are stored. The values of these tables are sequenced in ascending order, after the introduction of the maximum and minimum values as the first and second table values respectively. These latter values define the table limits for control purposes. The following constraints apply:

a. If X is outside the table limits, interpolation does not take place.

b. If quadratic or cubic interpolation is desired but the value of X is such that there is only one point below X in the table, a switch is made to linear interpolation. If cubic interpolation is desired but the value of X is such that there is only one point above X in the table, a switch is made to quadratic interpolation.

c. Linear interpolation requires two points X_1 and X_2 such that $X_1 < X \leq X_2$.

d. Quadratic interpolation requires three points X_1 , X_2 X_3 such that $X_1 < X_2 < X \leq X_3$.

e. Cubic interpolation requires four points X_1 , X_2 , X_3 and X_4 such that $X_1 < X_2 < X \le X_3 < X_4$.

CALLING SEQUENCE

The calling sequence is:

```
CALL NTERP(X, Y, MAXI, XO, YO, DYO, N, K, L)
```

SOLUTION METHOD

Initially, a check is made to determine if the argument is within the table limits. If so, L = 0. If the argument is less than the minimum table value, L is set to -1. If the argument is greater than the table limits, L is set to +1.

```
1.  L = 0
2.  If XO ≤ X(1), go to 6
3.  YO = Y(1)
4.  DYO = 0
5.  L = +1
    Return
6.  If XO ≥ X(2), go to 10
7.  YO = Y(2)
8.  DYO = 0
9.  L = -1
    Return
```

Next, find the interval for interpolation:

```
10.  If $K < 2$, $K = 2$, go to 13
11.  $K = K + 1$
12.  If $K > MAXI$, go to 3
13.  If $XO > X(K)$, go to 11
14.  Go to 17
15.  $K = K - 1$
16.  If $K < 2$, go to 7
17.  If $XO \leq X(K)$, go to 15
18.  If $N = 1$, go to 19
        $N = 2$, go to 25
        $N = 3$, go to 34
```

If linear interpolation is called for, the following equations are executed:

```
19.  $X1 = X(K)$
20.  $Y1 = Y(K)$
21.  $X2 = X(K + 1)$
22.  $Y2 = Y(K + 1)$
23.  $DYO = (Y2 - Y1)/(X2 - X1)$
24.  $YO = Y1 + (DYO)(XO - X1)$
     Return
```

If quadratic interpolation is called for by Eq. 18, then:

```
25.  If K = 2, go to 19
26.  X1 = X(K - 1)
27.  Y1 = Y(K - 1)
28.  X2 = X(K)
29.  Y2 = Y(K)
30.  X3 = X(K + 1)
31.  Y3 = Y(K + 1)
32.  DYO = (Y1)((2.0)(XO) - X2 - X3)/((X1 - X2)(X1 - X3))
          +(Y2)((2.0)(XO) - X1 - X3)/((X2 - X1)(X2 - X3))
          +(Y3)((2.0)(XO) - X1 - X2)/((X3 - X1)(X3 - X2))
33.  YO  = (XO - X2)(XO - X3)(Y1)/((X1 - X2)(X1 - X3))
          +(XO - X1)(XO - X3)(Y2)/((X2 - X1)(X2 - X3))
          +(XO - X1)(XO - X2)(Y3)/((X3 - X1)(X3 - X2))
     Return
```

If cubic interpolation is required, then:

```
34.  If K = 2, go to 19
35.  X1 = X(K - 1)
36.  Y1 = Y(K - 1)
37.  X2 = X(K)
38.  Y2 = Y(K)
```

```
39.  X3  =  X(K + 1)

40.  Y3  =  Y(K + 1)

41.  X4  =  X(K + 2)

42.  Y4  =  Y(K + 2)

43.  If X3 = X(1), go to 32

44.  DYO = (Y1)((XO - X2)(XO - X3) + (XO - X2)(XO - X4)
           +(XO - X3)(XO - X4))/((X1 - X2)(X1 - X3)(X1 - X4))
           +(Y2)((XO - X1)(XO - X3) + (XO - X1)(XO - X4)
           +(XO - X3)(XO - X4))/((X2 - X1)(X2 - X3)(X2 - X4))
           +(Y3)((XO - X1)(XO - X2) + (XO - X1)(XO - X4)
           +(XO - X2)(XO - X4))/((X3 - X1)(X3 - X2)(X3 - X4))
           +(Y4)((XO - X1)(XO - X2) + (XO - X1)(XO - X3)
           +(XO - X2)(XO - X3))/((X4 - X1)(X4 - X2)(X4 - X3))

45.  YO  = (Y1)(XO - X2)(XO - X3)(XO - X4)
           /((X1 - X2)(X1 - X3)(X1 - X4))
           +(Y2)(XO - X1)(XO - X3)(XO - X4)
           /((X2 - X1)(X2 - X3)(X2 - X4))
           +(Y3)(XO - X1)(XO - X2)(XO - X4)
           /((X3 - X1)(X3 - X2)(X3 - X4))
           +(Y4)(XO - X1)(XO - X2)(XO - X3)
           /((X4 - X1)(X4 - X2)(X4 - X3))

     Return
```

SUBROUTINE NTERP NOMENCLATURE

Symbol	Description	Units	Reference
DYO	Slope of Function at XO, YO	—	CALL, 4, 8, 23, 24, 32, 44
K	Location counter of last interpolation entry: initial setting zero	—	CALL, 10, 11, 12, 13, 15, 16, 17, 19, 20, 21, 22, 25, 26, 27, 28, 29, 30, 31, 34, 35, 36, 37, 38, 39, 40, 41, 42
L	Error indicator	—	CALL, 1, 5, 9
MAXI	Maximum size of tables	—	CALL, 12
N	Type of interpolation	—	CALL, 18
X	Independent variable table	—	CALL, DIM, 2, 6, 13, 17, 19, 21, 26, 28, 30, 35, 37, 39, 41, 43
XO	Interpolation value of independent variable	—	CALL, 2, 6, 13, 17, 24, 32, 33, 44, 45
X1	Specific input value	—	19, 23, 24, 26, 32, 33, 35, 44, 45
X2	Specific input value	—	21, 23, 28, 32, 37, 44, 45
X3	Specific input value	—	30, 32, 33, 39, 43, 44, 45
X4	Specific input value	—	41, 44, 45
Y	Dependent variable table	—	CALL, DIM, 3, 7, 20, 22, 27, 29, 31, 36, 38, 40, 42
YO	Interpolated value of dependent variable	—	CALL, 3, 7, 24, 33, 45
Y1	Specific input value	—	20, 23, 24, 27, 32, 33, 36, 44, 45

SUBROUTINE NTERP NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
Y2	Specific input value	—	22, 23, 29, 32, 33, 38, 44, 45
Y3	Specific input value	—	31, 32, 33, 40, 44, 45
Y4	Specific input value	—	42, 44, 45

```
$IBFTC NTERP   REF,FULIST
      SUBROUTINE NTERP(X,Y,MAXI,XO,YO,DYO,N,K,L)
      DIMENSION X(MAXI),Y(MAXI)
C          X = INDEPENDENT VARIABLE TABLE
C          Y = DEPENDENT VARIABLE TABLE
C          MAXI = MAXIMUM SIZE OF TABLES
C          XO = INTERPOLATION VALUE OF INDEPENDENT VARIABLE
C          YO = INTERPOLATED VALUE OF DEPENDENT VARIABLE
C          DYO = SLOPE OF FUNCTION AT (XO,YO)
C          N = TYPE OF INTERPOLATION
C          K    = LOCATION COUNTER OF LAST INTERPOLATION ENTRY
C          L = PSEUDO ERROR INDICATOR   IF ARGUMENT WITHIN TABLE
C          LIMITS L = 0, IF ARGUMENT LESS THAN MINIMUM TABLE VALUE
C          L = -1, IF ARGUMENT GREATER THAN TABLE LIMITS, L = +1
C               CHECK TO SEE IF WITHIN TABLE LIMITS
      L = 0
      IF (XO .LE. X(1))  GO TO 2
    1 YO = Y(1)
      DYO = 0.0
      L = +1
      RETURN
    2 IF (XO .GE. X(2))  GO TO 4
    3 YO = Y(2)
      DYO = 0.0
      L = -1
      RETURN
C          FIND INTERVAL FOR INTERPOLATION
    4 IF (K .LT. 2)  K = 2
      GO TO 6
    5 K = K + 1
      IF (K .GT. MAXI)  GO TO 1
    6 IF (XO .GT. X(K))  GO TO 5
      GO TO 8
    7 K = K - 1
      IF (K .LT. 2)  GO TO 3
    8 IF (XO .LE. X(K))  GO TO 7
      GO TO (10,20,30), N
C          LINEAR INTERPOLATION
   10 X1 = X(K)
      Y1 = Y(K)
      X2 = X(K+1)
      Y2 = Y(K+1)
      DYO = (Y2 - Y1)/(X2 - X1)
      YO = Y1 + DYO * (XO - X1)
      RETURN
C          QUADRATIC INTERPOLATION
   20 IF (K .EQ. 2)  GO TO 10
      X1 = X(K-1)
      Y1 = Y(K-1)
      X2 = X(K)
```

FIGURE 30. SYMBOLIC LISTING OF SUBROUTINE NTERP

```
      Y2 = Y(K)
      X3 = X(K+1)
      Y3 = Y(K+1)
   21 DY0 = Y1 * (2.0 * X0 - X2 - X3)/((X1 - X2) * (X1 - X3))
     $        + Y2 * (2.0 * X0 - X1 - X3)/((X2 - X1) * (X2 - X3))
     $        + Y3 * (2.0 * X0 - X1 - X2)/((X3 - X1) * (X3 - X2))
      Y0 = (X0 - X2) * (X0 - X3) * Y1/((X1 - X2) * (X1 - X3))
     $        + (X0 - X1) * (X0 - X3) * Y2/((X2 - X1) * (X2 - X3))
     $        + (X0 - X1) * (X0 - X2) * Y3/((X3 - X1) * (X3 - X2))
      RETURN
C           CUBIC INTERPOLATION
   30 IF (K .EQ. 2)  GO TO 10
      X1 = X(K-1)
      Y1 = Y(K-1)
      X2 = X(K)
      Y2 = Y(K)
      X3 = X(K+1)
      Y3 = Y(K+1)
      X4 = X(K+2)
      Y4 = Y(K+2)
      IF (X3 .EQ. X(1))  GO TO 21
      DY0 = Y1 * ((X0 - X2) * (X0 - X3) + (X0 - X2) * (X0 - X4)
     $        + (X0 - X3) * (X0 - X4))/((X1 - X2) * (X1 - X3)
     $             * (X1 - X4))
     $        + Y2 * ((X0 - X1) * (X0 - X3) + (X0 - X1) * (X0 - X4)
     $        + (X0 - X3) * (X0 - X4))/((X2 - X1) * (X2 - X3)
     $             * (X2 - X4))
     $        + Y3 * ((X0 - X1) * (X0 - X2) + (X0 - X1) * (X0 - X4)
     $        + (X0 - X2) * (X0 - X4))/((X3 - X1) * (X3 - X2)
     $             * (X3 - X4))
     $        + Y4 * ((X0 - X1) * (X0 - X2) + (X0 - X1) * (X0 - X3)
     $        + (X0 - X2) * (X0 - X3))/((X4 - X1) * (X4 - X2)
     $             * (X4 - X3))
      Y0 = Y1 * (X0 - X2) * (X0 - X3) * (X0 - X4)/((X1 - X2)
     $        * (X1 - X3) * (X1 - X4))
     $        + Y2 * (X0 - X1) * (X0 - X3) * (X0 - X4)/((X2 - X1)
     $        * (X2 - X3) * (X2 - X4))
     $        + Y3 * (X0 - X1) * (X0 - X2) * (X0 - X4)/((X3 - X1)
     $        * (X3 - X2) * (X3 - X4))
     *        + Y4 * (X0 - X1) * (X0 - X2) * (X0 - X3)/((X4 - X1)
     $        * (X4 - X2) * (X4 - X3))
      RETURN
      END
```

FIGURE 30. SYMBOLIC LISTING OF SUBROUTINE NTERP (cont.)

XXVII. SUBROUTINE STDATM

This subroutine calculates the atmospheric pressure, density, temperature and speed of sound, given the altitude in feet.

The computational method used first computes the geopotential height, and then the other values as a function of this height. The basic data are those of the ARDC 1956 Standard Atmosphere. The reference sea-level Earth radius is taken as 20855277.8928 feet.

CALLING SEQUENCE

The calling sequence is:

CALL STDATM (H, D, S, T, P)

SOLUTION METHOD

First the geopotential height is calculated:

1. GPH = (20855277.8928)(H)/(H + 20855277.8928)

Then the temperature and pressure are calculated as a function of the geopotential height interval:

2. If GPH $>$ 0, go to 5

   If GPH $\le$ 0, go to 3

3. TW = 288.16

```
4.  Go to 7
5.  If $GPH > 36089.24$, go to 9
    If $GPH \leq 36089.24$, go to 6
6.  $TW = 288.16 - (0.1981199)(10)^{-2}(GPH)$
7.  $PW = (2116.22)(1.0 - (0.6875347)(10)^{-5}(GPH))^{5.25612}$
8.  Go to 39
9.  If $GPH > 82020.997$, go to 13
    If $GPH \leq 82020.997$, go to 10
10. $TW = 216.66$
11. $PW = \exp(7.8929846 - (0.4806324)(10)^{-4}(GPH))$
12. Go to 39
13. If $GPH > 154199.475$, go to 17
    If $GPH \leq 154199.475$, go to 14
14. $TW = 141.66001 + (0.9143999)(10)^{-3}(GPH)$
15. $PW = (51.97536)(0.6538355$
            $+ (0.4220438)(10)^{-5}(GPH))^{-11.3883}$
16. Go to 39
17. If $GPH > 173884.514$, go to 21
    If $GPH \leq 173884.514$, go to 18
```

```
18. $TW = 282.66$
19. $PW = \exp(6.6032876 - (0.3684067)(10)^{-4}(GPH))$
20. Go to 39
21. If $GPH > 246062.992$, go to 25
    If $GPH \le 246062.992$, go to 22
22. $TW = 489.3959 - (0.11887199)(10)^{-2}(GPH)$
23. $PW = (1.2180285)(1.7312672 - (0.42054765)(10)^{-5}(GPH))^{8.7602036}$
24. Go to 39
25. If $GPH > 295275.59$, go to 29
    If $GPH \le 295275.59$, go to 26
26. $TW = 196.86$
27. $PW = \exp(10.04434 - (0.52897637)(10)^{-4}(GPH))$
28. Go to 39
29. If $GPH > 413385.82$, go to 33
    If $GPH \le 413385.82$, go to 30
30. $TW = -118.13999 + (0.0010668)(GPH)$
31. $PW = (0.37906765)(10)^{-2}(-0.6001219 + (0.54190795)(10)^{-5}(GPH))^{-9.76137}$
```

```
32. Go to 39
33. If $GPH > 574146.98$, go to 37
    If $GPH \leq 574146.98$, go to 34
34. $TW = -937.13956 + (0.30479999)(10)^{-2}(GPH)$
35. $PW = (0.30304527)(10)^{-4}(-2.9026203 + (0.9440624)(10)^{-5}(GPH))^{-3.4164794}$
36. Go to 39
37. $TW = -202.161527 + (0.176787752)(10)^{-2}(GPH)$
38. $PW = (0.12925894)(10)^{-5}(-0.2486775 + (0.21748395)(10)^{-5}(GPH))^{-5.8904818}$
```

The final values are determined from the relations:

```
39. $SQRTTW = TW^{0.5}$
40. $S = (65.766336)(SQRTTW)$
41. $D = (0.32365915)(10)^{-3}(PW)/(TW)$
42. $P = (PW)/(144.0)$
43. $T = (1.8)(TW) - 459.69$
    Return
```

SUBROUTINE STDATM NOMENCLATURE

Symbol	Description	Units	Reference
D	Density	$slugs/ft^3$	CALL, 41
GPH	Geopotential height	feet	1, 2, 5, 6, 7, 9, 11, 13, 14, 15, 17, 19, 21, 22, 23, 25, 27, 29, 30, 31, 33, 34, 35, 37, 38
H	Altitude	feet	CALL, 1
P	Pressure	lb/in^2	CALL, 42
PW	Interim pressure	lb/ft^2	7, 11, 15, 19, 23, 27, 31, 35, 38, 41, 42
S	Speed of sound	ft/sec	CALL, 40
SQRTTW	Square root of TW	—	39, 40
T	Temperature	°F	CALL, 43
TW	Interim temperature	°K	3, 6, 10, 14, 18, 22, 26, 30, 34, 37, 39, 41, 43

```
$IBFTC STDATM  REF,FULIST
C          STANDARD ATMOSPHERE ROUTINE
      SUBROUTINE STDATM(H,D,S,T,P)
C          H = GEOMETRIC ALTITUDE (FEET)
C          D = DENSITY (SLUGS/FEET**3)
C          S = SPEED OF SOUND (FEET/SECOND**2)
C          T = TEMPERATURE (DEGREES F)
C          P = PRESSURE (POUNDS/INCH**2)
      GPH = H * 20855277.8928/(H + 20855277.8928)
C          CALCULATE TEMPERATURE AND PRESSURE)
      IF (GPH .GT. 0.0)  GO TO 10
      TW = 288.16
      GO TO 11
   10 IF (GPH .GT. 36089.24)  GO TO 20
      TW = 288.16 - .1981199E-2 * GPH
   11 PW = 2116.22 * (1.0 - .6875347E-5 * GPH)**5.25612
      GO TO 100
   20 IF (GPH .GT. 82020.997)  GO TO 30
      TW = 216.66
      PW = EXP(7.8929846 - .4806324E-4 * GPH)
      GO TO 100
   30 IF (GPH .GT. 154199.475)  GO TO 40
      TW = 141.66001 + .9143999E-3 * GPH
      PW = 51.97536 * (.6538355 + .4220438E-5 * GPH)**(-11.3883)
      GO TO 100
   40 IF (GPH .GT. 173884.514)  GO TO 50
      TW = 282.66
      PW = EXP(6.6032876 - .3684067E-4 * GPH)
      GO TO 100
   50 IF (GPH .GT. 246062.992)  GO TO 60
      TW = 489.3599 - .11887199E-2 * GPH
      PW = 1.2180285 * (1.7312672 - .42054765E-5 * GPH)**8.7602036
      GO TO 100
   60 IF (GPH .GT. 295275.59)  GO TO 70
      TW = 196.86
      PW = EXP(10.04434 - .52897637E-4 * GPH)
      GO TO 100
   70 IF (GPH .GT. 413385.82)  GO TO 80
      TW = -118.13999 + .0010668 * GPH
      PW = .37906765E-2 * (-.6001219 + .54190795E-5 * GPH)**(-9.76137)
      GO TO 100
   80 IF (GPH .GT. 574146.98)  GO TO 90
      TW = -937.13956 + .30479999E-2 * GPH
      PW = .30304527E-4 * (-2.9026203 + .9440624E-5 * GPH)**
     $                                                       (-3.4164794)
      GO TO 100
   90 TW = -202.161527 + .176787752E-2 * GPH
      PW = .12925894E-5 * (-.2486775 + .21748395E-5 * GPH)**
```

FIGURE 31. SYMBOLIC LISTING OF SUBROUTINE STDATM

```
     $                                                       (-5.8904818)
 100 SQRTTW = SQRT(TW)
     S = 65.766336 * SQRTTW
     D = .32365915E-3 * PW/TW
     P = PW/144.0
     T = 1.8 * TW - 459.69
     RETURN
     END
```

FIGURE **31.** SYMBOLIC LISTING OF SUBROUTINE STDATM (cont.)

XXVIII. SUBROUTINE HITE

This subroutine computes the height of propellant in a tank, given the tank geometry and propellant volume.

The analysis is based upon a tank geometry which meets the following qualifications:

a. The tank may have a cylindrical section of radius R and length L, with hemispherical and/or semiellipsoidal headers.

b. The tank may be spherical, composed of two hemispherical headers.

c. The tank may be ellipsoidal composed of two semiellipsoidal headers.

d. The semiellipsoidal headers are limited to the type designated as "square root of two" headers. That is, the cross section of the tank is circular and has a maximum radius equal to R, while the depth of the header is equal to $R/\sqrt{2}$.

The 16 combinatorial possibilities of the various headers, with or without cylindrical tank sections, are shown in Figure 32.

The equations defining propellant height as a function of header geometry and propellant volume are given in Figures 33 through 40. The cubic equations are solved by calling Subroutine CROOT.

CALLING SEQUENCE

The calling sequence is:

CALL HITE (VOL, T, HT)

T is an array containing the description of the tank to be analyzed. This description contains:

Word 1: The length of the cylindrical part of the tank in feet

Word 2: Radius of the tank in feet

Word 3: Type of upper header (1 - 4)

Word 4: Type of lower header (1 - 4)

The numerical designators in Words 3 and 4 above have the following meanings:

A "1" specifies a hemispherical header

A "2" specifies a semiellipsoidal header

A "3" specifies an inverted hemispherical header

A "4" specifies an inverted semiellipsoidal header

The following constants are stored as permanent program data:

PI = 3.14159265
FRTHPI = (4/3) PI = 4.1887902
THROPI = 3/PI = 0.95492966
SQR2 = $2^{0.5}$ = 1.414214

SOLUTION METHOD

The following initial definitions are entered:

1. V = VOL
2. B = 0
3. C = 0
4. X1 = 0
5. XLGTH = T(1)
6. R = T(2)
7. NTOP = T(3)
8. NBOT = T(4)
9. XL = R

A test is made on the validity of the input data:

10. If NBOT $\leq$ 0, go to 217

    If NBOT $>$ 0, go to 11

11. If NTOP $\leq$ 0, go to 217

    If NTOP $>$ 0, go to 12

12. If NBOT $\leq$ 4, go to 13

    If NBOT $>$ 4, go to 217

13. If NTOP $\leq$ 4, go to 14

    If NTOP $>$ 4, go to 217

The following equations are established:

14. RSQ = (R)/(SQR2)

15. V1 = (PI)(R^2)(XLGTH)

16. V3 = (0.5)(FRTHPI)(R^3)

17. VH = (PI)$(R)^3$

The tank configuration is established:

18. If NTOP = 1, go to (19, 81, 55, 141), NBOT

If NTOP = 2, go to (103, 114, 123, 157), NBOT

If NTOP = 3, go to (42, 131, 72, 164), NBOT

If NTOP = 4, go to (173, 187, 196, 208), NBOT

The following equations define the elements of the tank volume, and propellant height. The sketches of Figure 32 refer to the 16 combinatorial possibilities expressed by Eq. 18 above.

19. V5 = V1 + (V3)(2)

20. V4 = V1 + V3

21. If $V < V5$, go to 22

If $V = V5$, go to 40

If $V > V5$, go to 215

22. If $V < V4$, go to 23

If $V = V4$, go to 38

If $V > V4$, go to 35

```
23. If V < V3, go to 27
    If V = V3, go to 33
    If V > V3, go to 24
24. DV = V - V3
25. HGT = $(DV)/((PI)(R)^2) + R$
26. HT = HGT
    Return
27. DV = V3 - V
28. N = 1
29. C = $(-3)(R)^2$
30. D = (THROPI)(DV)
```

To find X1, Subroutine CROOT is called. The calling sequence is:

```
31. CALL CROOT (1., B, C, D, XL, X1)
32. If N = 1, go to 33
    If N = 2, go to 38
    If N = 3, go to 64
    If N = 4, go to 79
    If N = 5, go to 101
    If N = 6, go to 53
    If N = 7, go to 185
    If N = 8, go to 204
    If N = 9, go to 137
```

```
33.  HGT = R - X1
34.  Go to 26
35.  N = 2
36.  DV = V - V4
37.  Go to 29
38.  HGT = XLGTH + R + X1
39.  Go to 26
40.  HGT = XLGTH + 2R
41.  Go to 26
42.  VX = VH - V3
43.  V4 = V1 - VH + V3
44.  V5 = V4 + VX
45.  If V < V5, go to 46
     If V = V5, go to 38
     If V > V5, go to 215
46.  If V < V4, go to 23
     If V = V4, go to 53
     If V > V4, go to 47
47.  N = 6
48.  DV = V - V4
49.  B = (-3)(R)
```

```
50.  C = (-B)(R)
51.  D = (-THROPI)(DV)
52.  Go to 31
53.  HGT = XLGTH + X1
54.  Go to 26
55.  VX = VH - V3
56.  V4 = VX + V1 - VH
57.  V5 = V4 + V3
58.  If V < V5, go to 59
     If V = V5, go to 38
     If V > V5, go to 215
59.  If V < V4, go to 60
     If V = V4, go to 53
     If V > V4, go to 66
60.  If V < VX, go to 61
     If V = VX, go to 33
     If V > VX, go to 68
61.  D = (-THROPI)(V)
62.  N = 3
63.  Go to 31
64.  HGT = X1
```

```
65.  Go to 26
66.  N = 6
67.  Go to 36
68.  DV = V + V3
69.  RSQ = 0
70.  $HGT = (DV)/((PI)(R^2)) + RSQ$
71.  Go to 26
72.  VX = VH - V3
73.  V4 = VX + (V1 - (2)(VH))
74.  V5 = V4 + VX
75.  If V < V5, go to 76
     If V = V5, go to 53
     If V > V5, go to 215
76.  If V < V4, go to 60
     If V = V4, go to 79
     If V > V4, go to 77
77.  N = 4
78.  Go to 48
79.  HGT = XLGTH - R + X1
80.  Go to 26
81.  V2 = V3
```

```
82.  V3 = (V3)/(SQR2)
83.  V4 = V1 + V3
84.  V5 = V4 + V2
85.  If V < V5, go to 86
     If V = V5, go to 100
     If V > V5, go to 215
86.  If V < V4, go to 87
     If V = V4, go to 101
     If V > V4, go to 97
87.  If V < V3, go to 88
     If V = V3, go to 95
     If V > V3, go to 93
88.  B = (-3)(RSQ)
89.  D = (3)(V)/((2)(PI))
90.  N = 3
91.  XL = RSQ
92.  Go to 31
93.  DV = V - V3
94.  Go to 70
95.  HGT = RSQ + X1
96.  Go to 26
```

```
97.   DV = V - V4
98.   N = 5
99.   Go to 29
100.  X1 = R
101.  HGT = XLGTH + X1 + RSQ
102.  Go to 26
103.  V2 = (V3)/(SQR2)
104.  V4 = V1 + V3
105.  V5 = V4 + V2
106.  If V < V5, go to 107
      If V = V5, go to 100
      If V > V5, go to 215
107.  If V < V4, go to 23
      If V = V4, go to 38
      If V > V4, go to 108
108.  N = 2
109.  DV = V - V4
110.  C = (-3)($R^2$)/(2)
111.  D = (3)(DV)/((2)(PI))
112.  XL = RSQ
113.  Go to 31
```

```
114.  V3 = (V3)/(SQR2)
115.  V4 = V1 + V3
116.  V5 = V4 + V3
117.  If V < V5, go to 118
      If V = V5, go to 121
      If V > V5, go to 215
118.  If V < V4, go to 87
      If V = V4, go to 101
      If V > V4, go to 119
119.  N = 5
120.  Go to 109
121.  R = RSQ
122.  Go to 40
123.  V2 = (V3)/(SQR2)
124.  VX = VH - V3
125.  V4 = VX + V1 - VH
126.  V5 = V4 + V2
127.  If V < V5, go to 128
      If V = V5, go to 101
      If V > V5, go to 215
```

```
128.  If V < V4, go to 60
      If V = V4, go to 53
      If V > V4, go to 129
129.  N = 6
130.  Go to 109
131.  VX = VH - V3
132.  V3 = (V3)/(SQR2)
133.  V4 = V3 + V1 - VH
134.  V5 = V4 + VX
135.  If V < V5, go to 136
      If V = V5, go to 101
      If V > V5, go to 215
136.  If V < V4, go to 87
      If V = V4, go to 137
      If V > V4, go to 139
137.  HGT = XLGTH + RSQ - R + X1
138.  Go to 26
139.  N = 9
140.  Go to 48
141.  V2 = V3
142.  VH = (VH)/(SQR2)
```

```
143.  VX = VH - (V3)/(SQR2)
144.  V4 = VX + V1 - VH
145.  V5 = V4 + V2
146.  If V < V5, go to 147
      If V = V5, go to 38
      If V > V5, go to 215
147.  If V < V4, go to 148
      If V = V4, go to 53
      If V > V4, go to 152
148.  If V < VX, go to 149
      If V = VX, go to 95
      If V > VX, go to 155
149.  D = (-3)(V)/((2)(PI))
150.  XL = RSQ
151.  Go to 62
152.  DV = V - V4
153.  N = 6
154.  Go to 29
155.  V3 = (V3)/(SQR2)
156.  Go to 68
157.  V2 = (V3)/(SQR2)
```

```
158.  VH  =  (VH)/(SQR2)

159.  VX  =  VH - V2

160.  V4  =  VX + V1 - VH

161.  V5  =  V4 + V2

162.  If V < V5, go to 163

      If V = V5, go to 101

      If V > V5, go to 215

163.  If V < V4, go to 148

      If V = V4, go to 53

      If V > V4, go to 129

164.  V2  =  V3

165.  VH1  =  VH

166.  VH  =  (VH)/(SQR2)

167.  VX  =  VH - (V3)/(SQR2)

168.  VX1  =  VH1 - V2

169.  V4  =  VX + V1 - VH - VH1

170.  V5  =  V4 + VX1

171.  If V < V5, go to 172

      If V = V5, go to 53

      If V > V5, go to 215
```

```
172.  If V < V4, go to 148
      If V = V4, go to 79
      If V > V4, go to 77
173.  VH = (VH)/(SQR2)
174.  V4 = V3 + V1 - VH
175.  VX = VH - (V3)/(SQR2)
176.  V5 = V4 + VX
177.  If V < V5, go to 178
      If V = V5, go to 38
      If V > V5, go to 215
178.  If V < V4, go to 23
      If V = V4, go to 185
      If V > V4, go to 179
179.  N = 7
180.  DV = V - V4
181.  B = (-3)(RSQ)
182.  C = $(3)(R^2)/(2)$
183.  DV = -DV
184.  Go to 111
185.  HGT = XLGTH + R + X1 - RSQ
186.  Go to 26
```

```
187.  V3  =  (V3)/(SQR2)

188.  VH  =  (VH)/(SQR2)

189.  VX  =  VH - V3

190.  V4  =  V3 + V1 - VH

191.  V5  =  V4 + VX

192.  If V < V5, go to 193

      If V = V5, go to 101

      If V > V5, go to 215

193.  If V < V4, go to 87

      If V = V4, go to 53

      If V > V4, go to 194

194.  N  =  6

195.  Go to 180

196.  V2  =  (V3)/(SQR2)

197.  VH1  =  (VH)/(SQR2)

198.  VX  =  VH - V3

199.  VX1  =  VH1 - V2

200.  V4  =  V1 - V3 - VH1

201.  V5  =  V4 + VX1
```

```
202.  If $V < V5$, go to 203
      If $V = V5$, go to 53
      If $V > V5$, go to 215
203.  If $V < V4$, go to 60
      If $V = V4$, go to 204
      If $V > V4$, go to 206
204.  $R = 0$
205.  Go to 185
206.  $N = 8$
207.  Go to 180
208.  $V2 = (V3)/(SQR2)$
209.  $VH = (VH)/(SQR2)$
210.  $VX = VH - V2$
211.  $V4 = VX + V1 - (2)(VH)$
212.  $V5 = V4 + VX$
213.  If $V < V5$, go to 214
      If $V = V5$, go to 53
      If $V > V5$, go to 215
214.  If $V < V4$, go to 148
      If $V = V4$, go to 204
      If $V > V4$, go to 206
```

```
215.  HGT = -1
216.  Go to 26
217.  CALL SETERR (4HHITE)
      Return
```

SUBROUTINE HITE NOMENCLATURE

Symbol	Description	Units	Reference
B	Coefficient of X^2 in polynomial equation	—	2, 31, 49, 50, 88, 181
C	Coefficient of X in polynomial equation	—	3, 29, 31, 50, 110, 182
D	Constant of polynomial equation	—	30, 31, 51, 61, 89, 111, 149
DV	Differential volume	ft^3	24, 25, 27, 30, 36, 48, 51, 68, 70, 93, 97, 109, 111, 152, 180, 183
FRTHPI	4/3 PI	—	DATA, 16
HGT	Propellant height	ft	25, 26, 33, 38, 40, 53, 64, 70, 79, 95, 101, 137, 185, 215
HT	Propellant height	ft	CALL, 26
N	Control variable	—	28, 32, 35, 47, 62, 66, 77, 90, 98, 108, 119, 129, 139, 153, 179, 194, 206
NBOT	Lower header reference	—	8, 10, 12, 18
NTOP	Upper header reference	—	7, 11, 13, 18
PI	Constant	—	DATA, 15, 17, 25, 70, 89, 111, 149
R	Tank radius	ft	6, 9, 14, 15, 16, 17, 25, 29, 33, 38, 40, 49, 50, 70, 79, 100, 110, 121, 137, 182, 185, 204

SUBROUTINE HITE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RSQ	Tank radius / $\sqrt{2}$	ft	14, 69, 70, 88, 91, 95, 101, 112, 121, 137, 150, 181, 185
SQR2	Square root of 2	—	DATA, 14, 82, 103, 114, 123, 132, 142, 143, 155, 157, 158, 166, 167, 173, 175, 187, 188, 196, 197, 208, 209
T	Table of tank geometry	—	CALL DIM, 5, 6, 7, 8
THROPI	3/PI	—	DATA, 30, 51, 61
V	Element of tank volume	ft^3	1, 21, 22, 23, 24, 27, 36, 45, 46, 48, 58, 59, 60, 61, 68, 75, 76, 85, 86, 87, 89, 93, 97, 106, 107, 109, 117, 118, 127, 128, 135, 136, 146, 147, 148, 149, 152, 162, 163, 171, 172, 177, 178, 180, 192, 193, 202, 203, 213, 214
V1	Element of tank volume	ft^3	15, 19, 20, 43, 56, 73, 83, 104, 115, 125, 133, 144, 160, 169, 174, 190, 200, 211
V2	Element of tank volume	ft^3	81, 84, 103, 105, 123, 126, 141, 145, 157, 159, 161, 164, 168, 196, 199, 208, 210

SUBROUTINE HITE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
V3	Element of tank volume	ft^3	16, 19, 20, 23, 24, 27, 42, 43, 55, 57, 68, 72, 81, 82, 83, 87, 93, 103, 104, 114, 115, 116, 123, 124, 131, 132, 133, 141, 143, 155, 157, 164, 167, 174, 175, 187, 189, 190, 196, 198, 200, 208
V4	Element of tank volume	ft^3	20, 22, 36, 43, 44, 46, 48, 56, 57, 59, 73, 74, 76, 83, 84, 86, 97, 104, 105, 107, 109, 115, 116, 118, 125, 126, 128, 133, 134, 136, 144, 145, 147, 152, 160, 161, 162, 169, 170, 172, 174, 176, 178, 180, 190, 191, 193, 200, 201, 203, 211, 212, 214
V5	Element of tank volume	ft^3	19, 21, 44, 45, 57, 58, 74, 75, 84, 85, 105, 106, 116, 117, 126, 127, 134, 135, 145, 146, 161, 162, 170, 171, 176, 177, 191, 192, 201, 202, 212, 213
VH	Element of tank volume	ft^3	17, 42, 43, 55, 56, 72, 73, 124, 125, 131, 133, 142, 143, 144, 158, 159, 160, 165, 166, 167, 169, 173, 174, 175, 188, 189, 190, 197, 198, 209, 210, 211
VH1	Element of tank volume	ft^3	165, 168, 169, 197, 199, 200

SUBROUTINE HITE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
VOL	Propellant volume	ft^3	CALL, 1
VX	Element of tank volume	ft^3	42, 44, 55, 56, 60, 72, 73, 74, 124, 125, 131, 134, 143, 144, 148, 159, 160, 167, 169, 175, 176, 189, 191, 198, 210, 211, 212
VX1	Element of tank volume	ft^3	168, 170, 199, 201
X1	Propellant height	ft	4, 31, 33, 38, 53, 64, 79, 95, 100, 101, 137, 185
XL	Maximum possible value of propellant height	ft	9, 31, 91, 112, 150
XLGTH	Length of cylindrical part of tank	ft	5, 15, 38, 40, 53, 79, 101, 137, 185

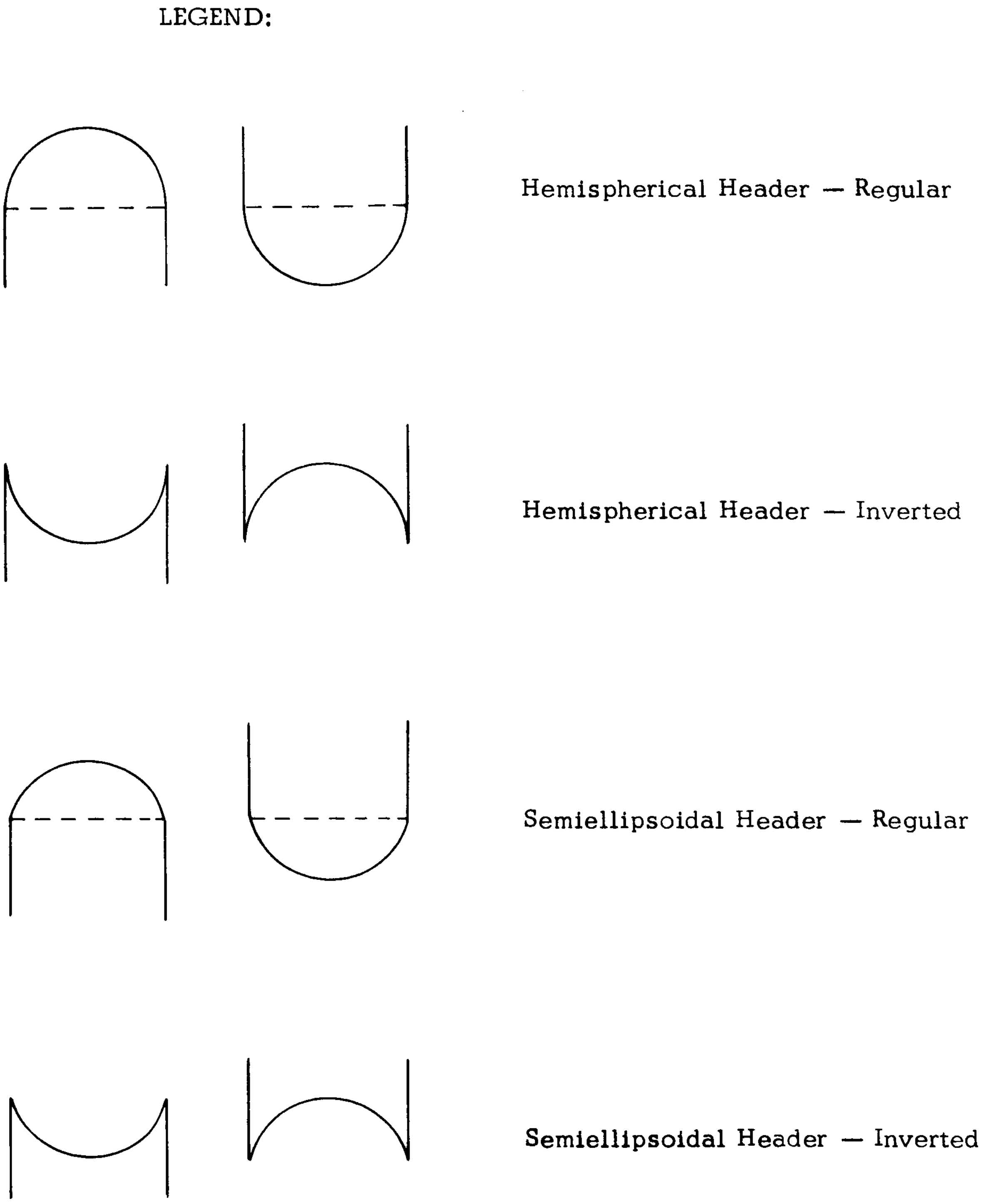
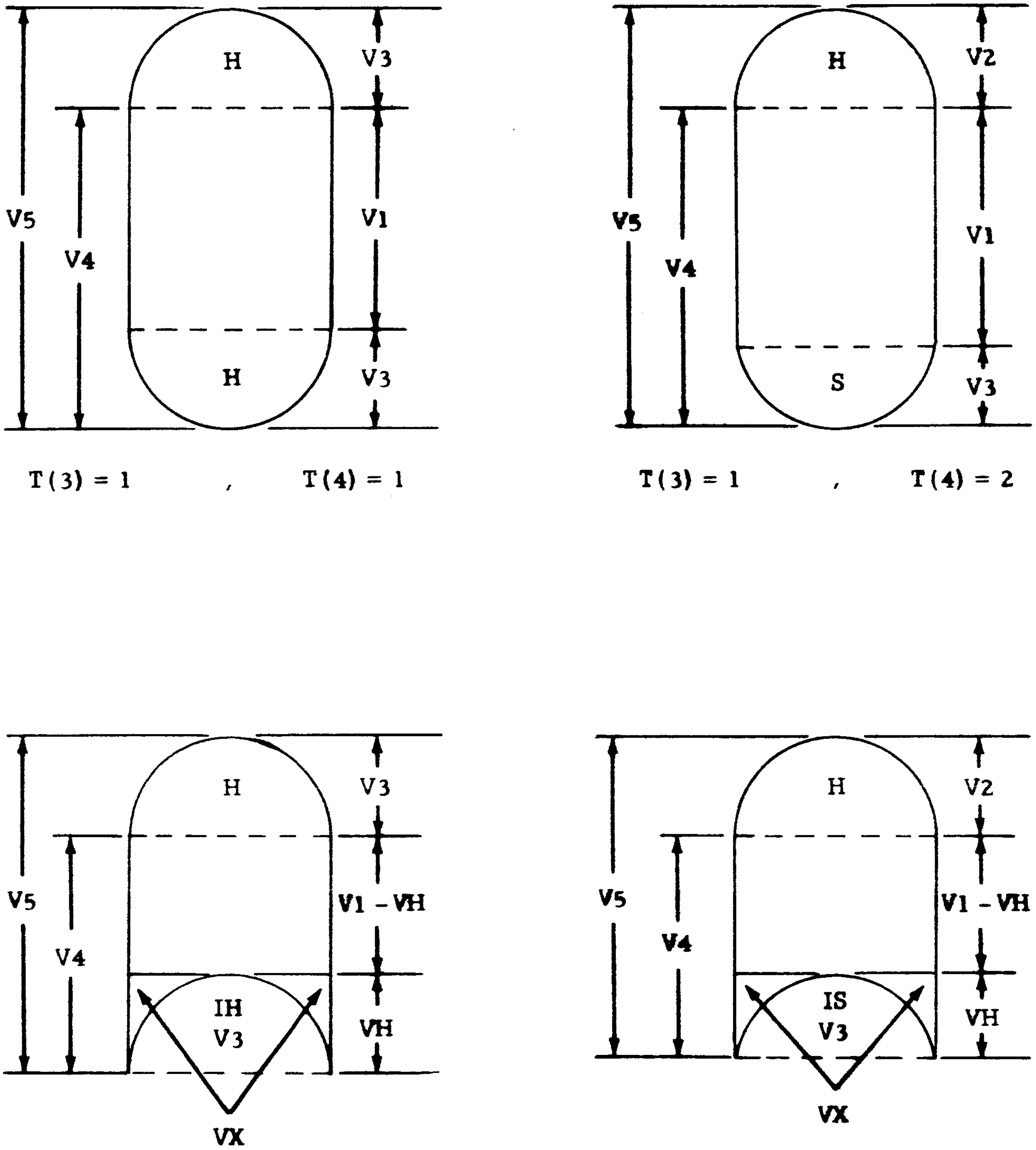


FIGURE 32. TANK CONFIGURATIONS



T(3) = 1 , T(4) = 1

T(3) = 1 , T(4) = 2

T(3) = 1 , T(4) = 3

T(3) = 1 , T(4) = 4

FIGURE 32. TANK CONFIGURATIONS (Cont.)

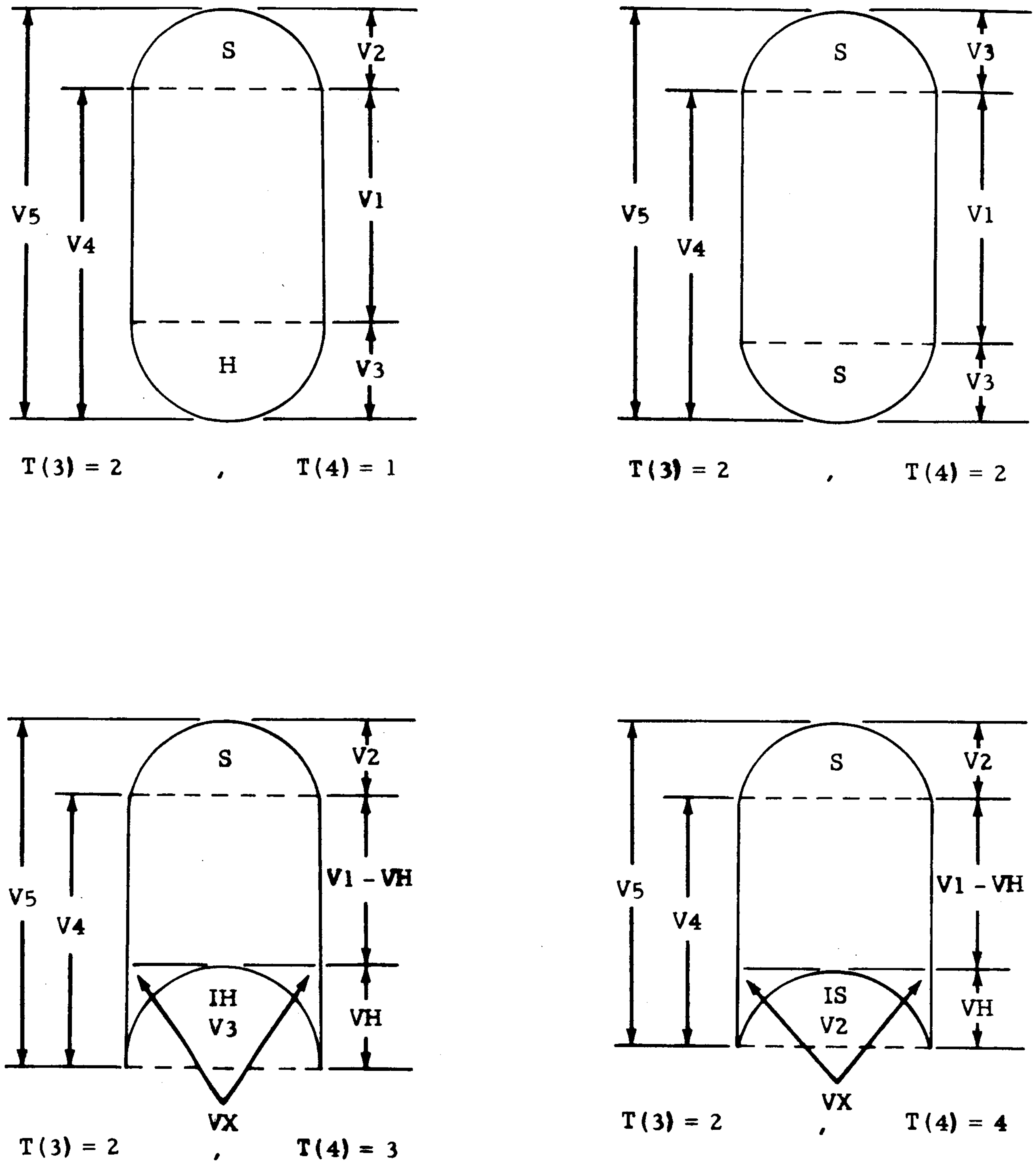
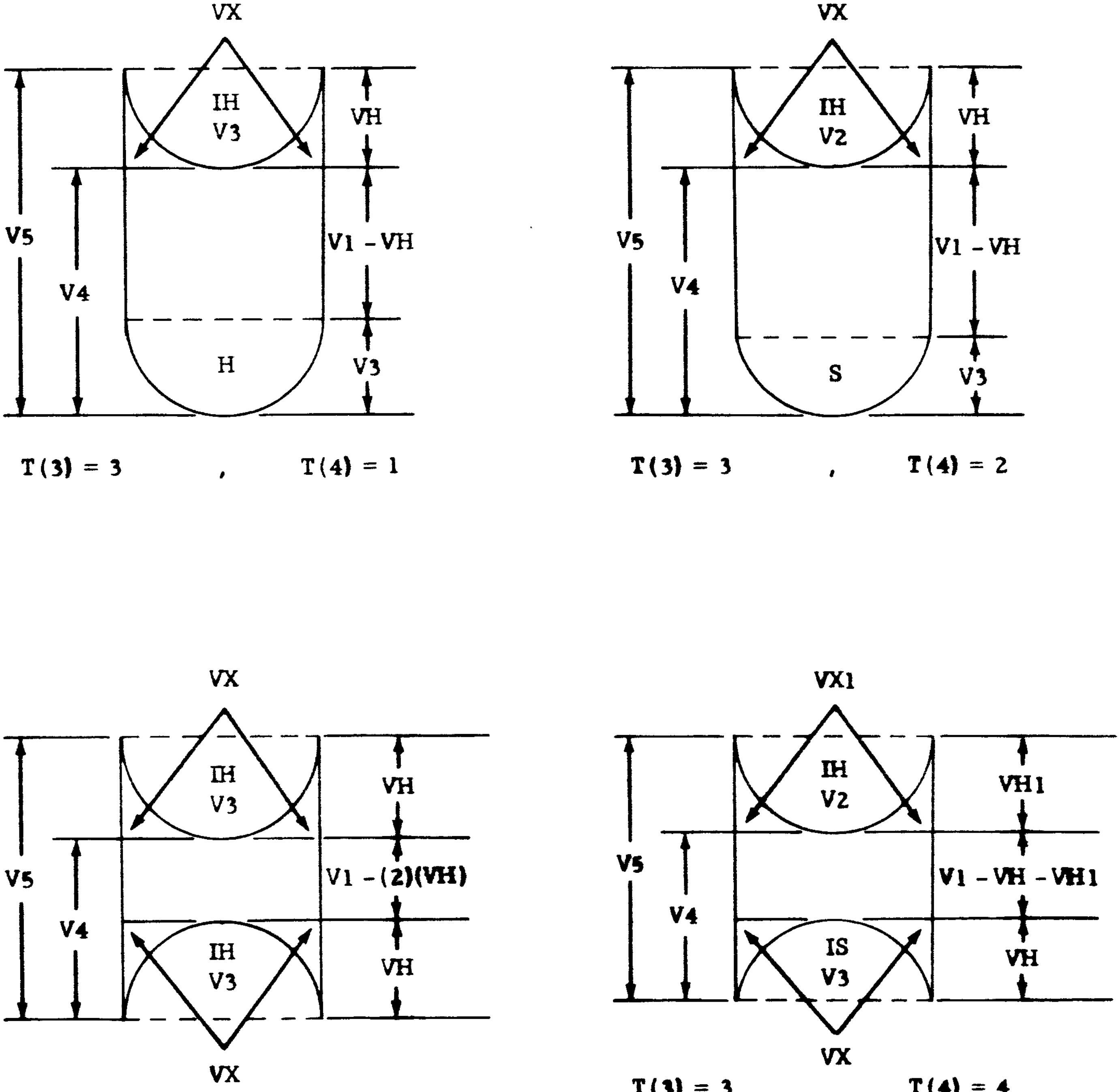


FIGURE 32. TANK CONFIGURATIONS (Cont.)



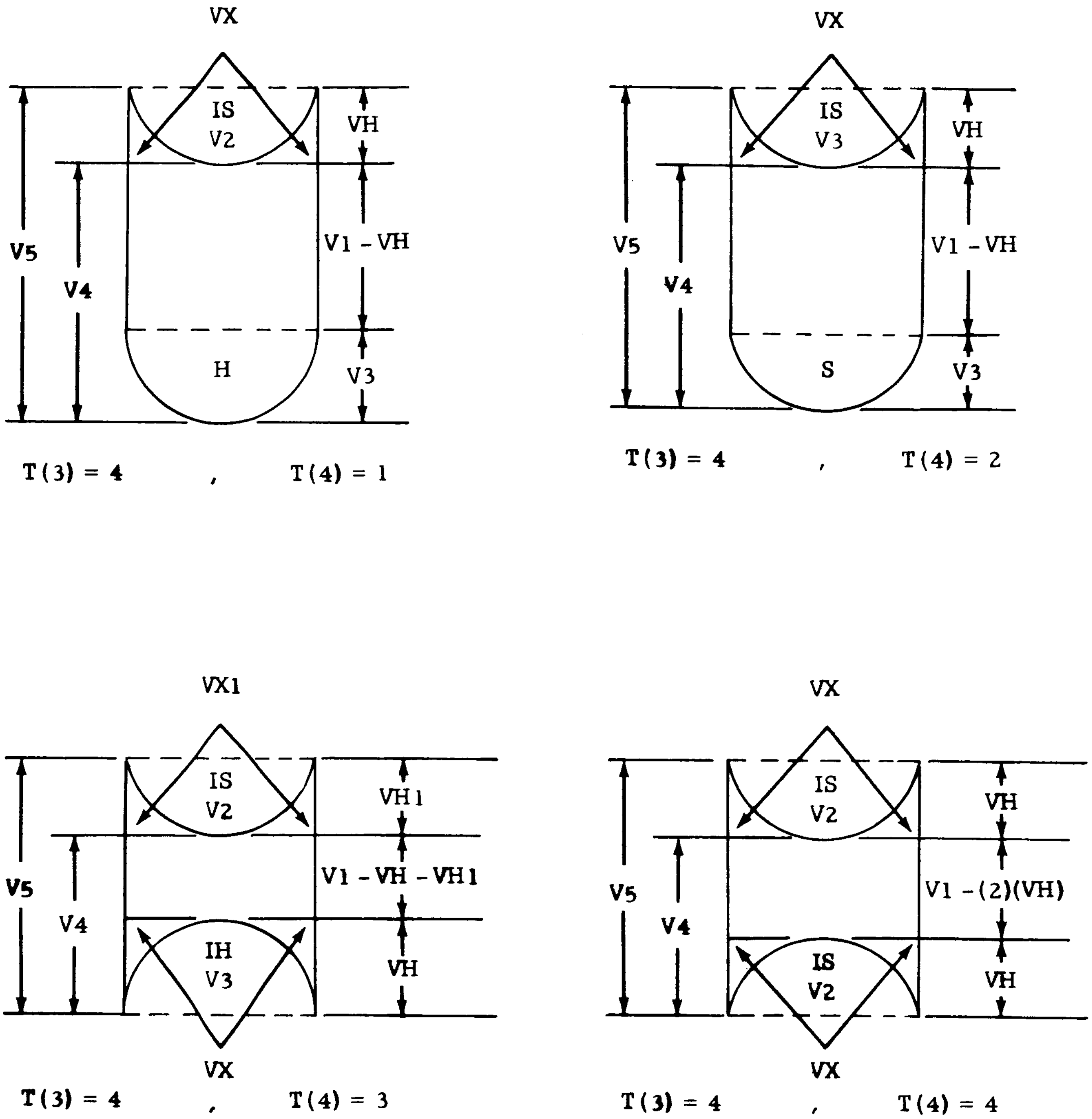
$T(3) = 3$, $T(4) = 1$

$T(3) = 3$, $T(4) = 2$

$T(3) = 3$, $T(4) = 3$

$T(3) = 3$, $T(4) = 4$

FIGURE 32. TANK CONFIGURATIONS (Cont.)



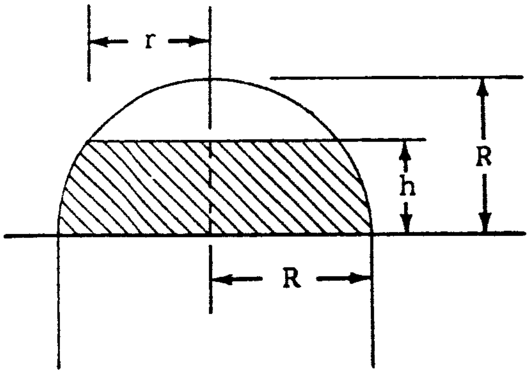
T(3) = 4 , T(4) = 1

T(3) = 4 , T(4) = 2

T(3) = 4 , T(4) = 3

T(3) = 4 , T(4) = 4

FIGURE 32. TANK CONFIGURATIONS (Cont.)



$$V = \frac{\pi h}{6}(3R^2 + 3r^2 + h^2)$$

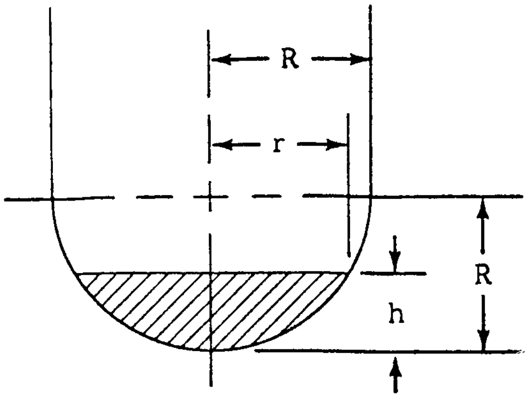
$$r^2 = R^2 - h^2$$

$$V = \frac{\pi h}{6}(3R^2 + 3R^2 - 3h^2 + h^2) = \frac{\pi h}{6}(6R^2 - 2h^2)$$

$$V = \frac{\pi h}{3}(3R^2 - h^2)$$

$$h^3 - 3R^2 h + \frac{3V}{\pi} = 0$$

FIGURE 33. HEMISPHERICAL UPPER HEADER, REGULAR



$$V = \frac{\pi h}{6}(3r^2 + h^2)$$

$$r^2 = R^2 - (R-h)^2$$

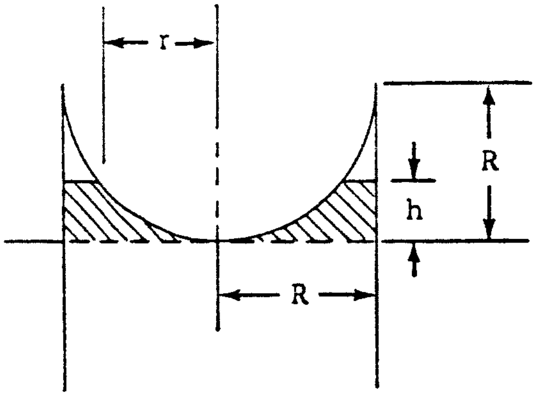
$$V = \frac{\pi h}{6}\left[3R^2 - 3(R-h)^2 + h^2\right]$$

$$V = \frac{\pi h}{6}(3R^2 - 3R^2 + 6Rh - 3h^2 + h^2)$$

$$V = \frac{\pi h}{6}(6Rh - 2h^2) = \frac{\pi h}{3}(3Rh - h^2)$$

$$h^3 - 3Rh^2 + \frac{3V}{\pi} = 0$$

FIGURE 34. HEMISPHERICAL LOWER HEADER, REGULAR



$$V = \pi R^3 - \pi R^2 (r - h) - \frac{\pi h}{6} (3r^2 + h^2)$$

$$r^2 = R^2 - (R - h)^2$$

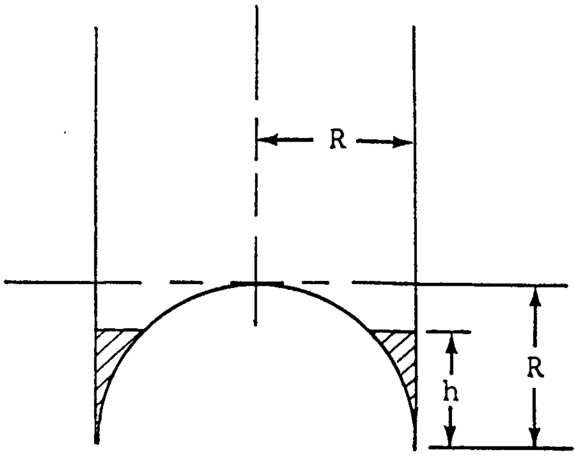
$$V = \pi R^3 - \pi R^3 + \pi R^2 h - \frac{\pi h}{6} \left[3R^2 - 3(R-h)^2 + h^2 \right]$$

$$V = \pi R^2 h - \frac{\pi R^2 h}{2} + \frac{\pi R^2 h}{2} - \pi R h^2 + \frac{\pi h^3}{2} - \frac{\pi h^3}{6}$$

$$V = \pi R^2 h - \pi R h^2 + \frac{\pi h^3}{3}$$

$$h^3 - 3Rh^2 + 3R^2 h - \frac{3V}{\pi} = 0$$

FIGURE 35. HEMISPHERICAL UPPER HEADER, INVERTED

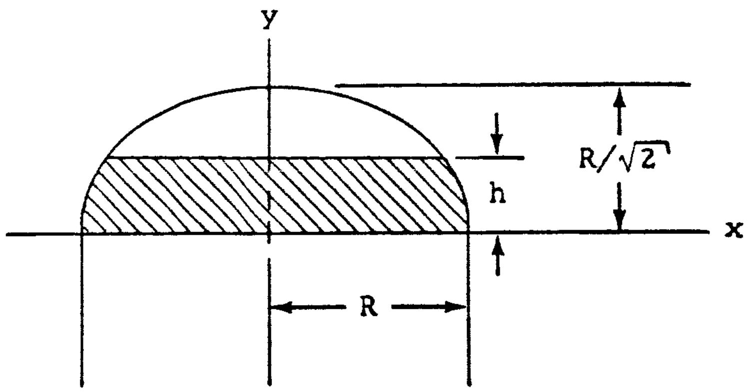


$$V = \pi R^2 h - \frac{\pi h}{3}(3R^2 - h^2)$$

$$V = \pi R^2 h - \pi R^2 h + \frac{\pi h^3}{3} = \frac{\pi h^3}{3}$$

$$h^3 = \frac{3V}{\pi}$$

FIGURE 36. HEMISPHERICAL LOWER HEADER, INVERTED



$$V = \pi \int_0^h f^2(y)\,dy$$

$$f(y) = \sqrt{R^2 - 2y^2}$$

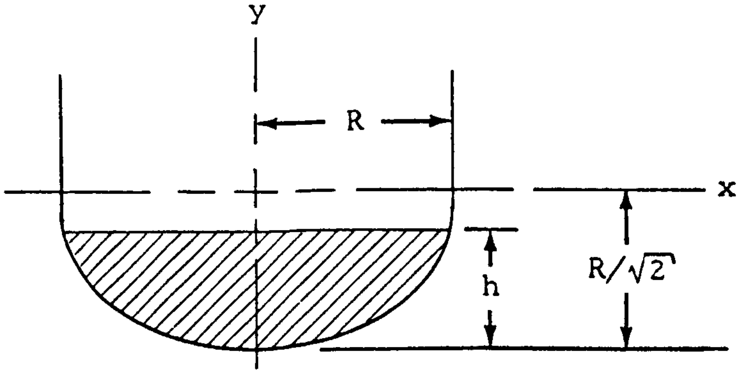
$$V = \pi \int_0^h (R^2 - 2y^2)\,dy$$

$$V = \pi \left[R^2 y - \frac{2y^3}{3} \right]_0^h$$

$$V = \pi \left[R^2 h - \frac{2h^3}{3} \right]$$

$$h^3 - \frac{3R^2 h}{2} + \frac{3V}{2\pi} = 0$$

FIGURE 37. SEMIELLIPSOIDAL UPPER HEADER



$$V = \pi \int_{-R/\sqrt{2}}^{-(R/\sqrt{2}-h)} f^2(y)\,dy$$

$$V = \pi \int_{-R/\sqrt{2}}^{-(R/\sqrt{2}-h)} (R^2 - 2y^2)\,dy$$

$$V = \pi \left[R^2 y - \frac{2y^3}{3} \right]_{-R/\sqrt{2}}^{-(R/\sqrt{2}-h)}$$

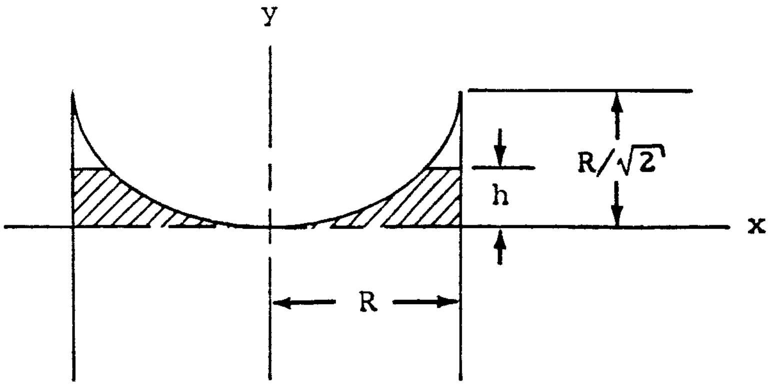
$$V = \pi \left[R^2 \left(-\frac{R}{\sqrt{2}} + h + \frac{R}{\sqrt{2}} \right) - \frac{2}{3} \left(-\left(\frac{R}{\sqrt{2}} - h \right)^3 + \left(\frac{R}{\sqrt{2}} \right)^3 \right) \right]$$

$$V = \pi \left[R^2 h - \frac{2}{3} \left(\frac{3R^2 h}{2} - \frac{3Rh^2}{\sqrt{2}} + h^3 \right) \right]$$

$$V = \pi \left[\frac{2Rh^2}{\sqrt{2}} - \frac{2h^3}{3} \right]$$

$$h^3 - \frac{3Rh^2}{\sqrt{2}} + \frac{3V}{2\pi} = 0$$

FIGURE 38. SEMIELLIPSOIDAL LOWER HEADER

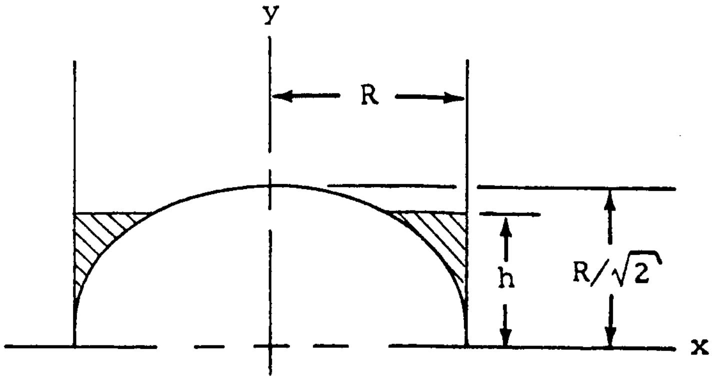


$$V = \pi R^2 h - \pi \left[\frac{2Rh^2}{\sqrt{2}} - \frac{2h^3}{3} \right]$$

$$V = \pi \left[R^2 h - \sqrt{2}\; Rh^2 + \frac{2h^3}{3} \right]$$

$$h^3 - \frac{3Rh^2}{\sqrt{2}} + \frac{3R^2 h}{2} - \frac{3V}{2\pi} = 0$$

FIGURE 39. SEMIELLIPSOIDAL UPPER HEADER, INVERTED



$$V = \pi R^2 h - \pi \left[R^2 h - \frac{2h^3}{3} \right]$$

$$V = \pi \left[R^2 h - R^2 h + \frac{2h^3}{3} \right]$$

$$V = \frac{2\pi h^3}{3}$$

$$h^3 = \frac{3V}{2\pi}$$

To find the height of propellant in a cylinder:

$$h = V/\pi R^2 .$$

FIGURE 40. SEMIELLIPSOIDAL LOWER HEADER, INVERTED

```
$IBFTC HEIGHT  FULIST,REF
      SUBROUTINE HITE(VOL,T,HT)
      DIMENSION T(4)
      DATA PI/3.14159265/,FRTHPI/4.1887902/,THROPI/.95492966/,
     1SQR2/1.414214/
      V = VOL
      DO 9999 I = 1,2
 9999 B = 0.
      C = 0.
      X1= 0.
      XLGTH = T(1)
      R = T(2)
      NTOP = T(3)
      NBOT = T(4)
      XL = R
      IF(NBOT)9000,9000,8
    8 IF(NTOP)9000,9000,5
    5 IF(NBOT-4)6,6,9000
    6 IF(NTOP-4)7,7,9000
    7 RSQ = R/SQR2
      V1 = PI* R*R *XLGTH
      V3 = .5*FRTHPI*R**3
      VH = PI* R**3
      GO TO(1,2,3,4),NTOP
    1 GO TO(1020,1060,1040,1110),NBOT
    2 GO TO(1070,1080,1090,1120),NBOT
    3 GO TO(1030,1100,1050,1130),NBOT
    4 GO TO(1140,1150,1160,1170),NBOT
 1020 V5 = V1+V3*2.
      V4 = V1+V3
      IF(V -V5)10, 29,1010
   10 IF(V -V4)12, 21,20
   12 IF(V -V3)25, 27,14
   14 DV = V-V3
   15 HGT =(DV/(        PI   *(R*R)))+R
 1000 HT = HGT
      RETURN
   25 DV = V3 - V
      N = 1
   26 C = -3.*(R*R)
      D =   THROPI  *DV
   24 CALL CROOT(1.,B,C,D,XL,X1)
      GO TO(27,21,44,57,67,38,148,163,102),N
   27 HGT = R-X1
      GO TO 1000
   20 N=2
   23 DV = V - V4
      GO TO 26
   21 HGT =  XLGTH+ R    +X1
      GO TO 1000
```

FIGURE 41. SYMBOLIC LISTING OF SUBROUTINE HITE

```
  29 HGT = XLGTH + R + R
     GO TO 1000
1030 VX = VH-V3
     V4 = V1 - VH  +V3
     V5 = V4 + VX
     IF(V-V5)30,21,1010
  30 IF(V-V4)12,38,35
  35 N = 6
  36 DV =  V - V4
     B = -3.*R
     C = -B*R
     D = -THROPI     * DV
     GO TO 24
  38 HGT = XLGTH + X1
     GO TO 1000
1040 VX = VH - V3
     V4 = VX +   V1-VH
     V5 = V4 + V3
     IF (V-V5)40,21,1010
  40 IF (V-V4)42,38,48
  42 IF (V-VX)43,27,49
  43 D = -THROPI * V
  46 N = 3
     GO TO 24
  44 HGT = X1
     GO TO 1000
  48 N = 6
     GO TO 23
  49 DV = V+ V3
     RSQ = 0.
  45 HGT = (DV/(PI*R**2)) +RSQ
     GO TO 1000
1050 VX = VH - V3
     V4 = VX +  (V1-2.*VH)
     V5 = V4 + VX
     IF(V-V5)50,38,1010
  50 IF(V-V4)42,57,56
  56 N=4
     GO TO 36
  57 HGT = XLGTH -R +X1
     GO TO  1000
1060 V2= V3
     V3 = V3/SQR2
     V4 = V1 + V3
     V5 = V4 + V2
     IF (V-V5)60,69,1010
  60 IF (V-V4)61,67,66
  61 IF (V-V3)62,64,63
  62 B = -3.*RSQ
     D = (3.*V)/(2.*PI)
```

FIGURE 41. SYMBOLIC LISTING OF SUBROUTINE HITE (cont.)

```
      N = 3
      XL = RSQ
      GO TO 24
   63 DV = V-V3
      GO TO 45
   64 HGT = RSQ + X1
      GO TO 1000
   66 DV = V -V4
      N = 5
      GO TO 26
   69 X1 = R
   67 HGT = XLGTH +X1 + RSQ
      GO TO 1000
 1070 V2 = V3/SQR2
      V4 = V1 + V3
      V5 = V4 + V2
      IF (V-V5)70,69,1010
   70 IF(V-V4)12,21,76
   76 N=2
   77 DV = V- V4
   79 C =(-3.*R*R)/2.
   78 D = (3.*DV) / (2.*PI)
      XL = RSQ
      GO TO 24
 1080 V3 = V3/SQR2
      V4 = V1 + V3
      V5 = V4 + V3
      IF (V-V5)80,89,1010
   80 IF(V-V4)61,67,86
   86 N = 5
      GO TO 77
   89 R = RSQ
      GO TO 29
 1090 V2 = V3/SQR2
      VX =  VH - V3
      V4 = VX+  V1-VH
      V5 =  V4 + V2
      IF (V-V5)90,67,1010
   90 IF (V-V4)42,38,96
   96 N = 6
      GO TO 77
 1100 VX = VH - V3
      V3 = V3 /SQR2
      V4 = V3 +  V1-VH
      V5 =   V4 + VX
      IF(V-V5)100,67,1010
  100 IF(V-V4)61,102,106
  102 HGT = XLGTH + RSQ- R +X1
      GO TO 1000
  106 N = 9
```

FIGURE 41. SYMBOLIC LISTING OF SUBROUTINE HITE (cont.)

```
      GO TO 36
1110  V2 = V3
      VH = VH/SQR2
      VX = VH - V3 /SQR2
      V4 = VX +  V1-VH
      V5 = V4 + V2
      IF (V - V5)110,21, 1010
 110  IF (V - V4)111,38, 113
 111  IF(V - VX)112,64,115
 112  D =(-3.*V)/(2.*PI)
      XL = RSQ
      GO TO 46
 113  DV = V - V4
      N = 6
      GO TO 26
 115  V3 = V3/SQR2
      GO TO 49
1120  V2 = V3/SQR2
      VH = VH/SQR2
      VX = VH-V2
      V4 = VX +  V1-VH
      V5 = V4 + V2
      IF(V-V5)120,67,1010
 120  IF(V-V4)111,38,96
1130  V2 = V3
      VH1 = VH
      VH = VH/SQR2
      VX = VH - V3/SQR2
      VX1= VH1 -V2
      V4 = VX + V1-VH-VH1
      V5 = V4 + VX1
      IF(V-V5)130,38,1010
 130  IF(V-V4)111,57,56
1140  VH = VH/SQR2
      V4 = V3 +  V1-VH
      VX = VH - V3/SQR2
      V5 =  V4 + VX
      IF(V-V5)140,21,1010
 140  IF(V-V4)12,148,147
 147  N =7
 149  DV = V - V4
      B = -3.*RSQ
      C=  3.*R*R/2.
      DV = -DV
      GO TO 78
 148  HGT = XLGTH + R + X1 - RSQ
      GO TO 1000
1150  V3 = V3/SQR2
      VH = VH/SQR2
      VX =  VH - V3
```

FIGURE 41. SYMBOLIC LISTING OF SUBROUTINE HITE (cont.)

```
      V 4 = V 3 +    V1- VH
      V5 =  V4 + VX
      IF(V-V5)150,67,1010
  150 IF(V-V4)61 ,38,152
  152 N=6
      GØ TØ 149
 1160 V2 = V3/SQR2
      VH1 = VH/SQR2
      VX  = VH-V3
      VX1 = VH1-V2
      V4 = V1 - V3-VH1
      V5 = V4 + VX1
      IF(V-V5)160,38,1010
  160 IF(V-V4)42,163,165
  163 R = 0.
      GØ TØ 148
  165 N=8
      GØ TØ 149
 1170 V2 = V3/SQR2
      VH = VH/SQR2
      VX =  VH - V2
      V4 = VX +  V1- 2.*VH
      V5 = V4 + VX
      IF(V-V5)170,38,1010
  170 IF (V-V4)111,163,165
 1010 HGT = -1.
      GØ TØ 1000
 9000 CALL SETERR(4HHITE)
      RETURN
      END
```

FIGURE 41. SYMBOLIC LISTING OF SUBROUTINE HITE (cont.)

XXIX. SUBROUTINE CROOT

This subroutine finds the real and positive roots of a polynomial of the form:

$$AX^3 + BX^2 + CX + D = 0$$

The use of the program is subject to the following constraints:

a. The coefficient (A) in the above equation cannot be zero.

b. There is only one positive root (X_i), which lies within a specified limit (XL); that is, the relation $0 \leq X_i \leq XL$ must be satisfied.

If the above criteria are not met, then the subroutine will set an error flag and return to the calling program.

The method of analysis generally follows that given in pages 7 through 9 of Ref. [2] for cubic equations.

Subroutine QUAD is called for analyses pertaining to the trigonometric portion of the solution.

CALLING SEQUENCE

The calling sequence is:

CALL CROOT (AA, B, C, D, XL, XI)

SOLUTION METHOD

The function CUBRT(XX) is defined.

1. $\text{CUBRT}(XX) = (|XX|^{1/3})(XX)/(|XX|)$

A counter is initially set to zero:

2. $I = 0$

The basic cubic equation is simplified by dividing it through by the coefficient of the first term:

3. $P = (B)/(AA)$

4. $Q = (C)/(AA)$

5. $R = (D)/(AA)$

The resulting equation is further reduced to eliminate the quadratic term, resulting in the coefficients:

6. $AP = ((3)(Q) - P^2)/(3)$

7. $BP = ((2)(P^3) - (9)(P)(Q) + (27)(R))/(27)$

A term of the resolvent equation is defined:

8. $DK = (AP)^{1/9} + (BP)^{1/2}$

The roots of the resolvent equation are then determined:

9. If $DK < 0$, go to 21

   If $DK = 0$, go to 10

   If $DK > 0$, go to 13

10. XY(1) = (2)(CUBRT(-(BP)/(2))) - (P) / (3)

11. XY(2) = CUBRT((BP)/(2)) - (P) / (3)

12. XY(3) = XY(2)

Go to 29.

13. SRTD = $(DK)^{1/2}$

14. BPSD = (-BP)/(2) + SRTD

15. BMSD = (-BP)/(2) - SRTD

If a real root exists, then it is found by the Cardan formula:

16. XY(1) = CUBRT(BPSD) + CUBRT(BMSD) - (P)/(3)

17. XI = XY(1)

18. If XY(1) $<$ 0 , go to 35

If XY(1) $\geq$ 0 , go to 19

19. If XL $<$ XY(1) , go to 35

If XL $\geq$ XY(1) , go to 20

20. Go to 36

If DK $<$ 0 , Eq. 9 above, then the roots of the resolvent equation are determined by the trigonometric technique:

21. YFACT = $(2)((-AP)/(3))^{1/2}$

22. CFE = $((-BP)/(2))/(-(AP^3)/(27))^{1/2}$

23. SFE = $(1 - CFE^2)^{1/2}$

Subroutine QUAD is called to solve for the trigonometric relationships:

```
24.   CALL QUAD (CFE, SFE, FE)
25.   DTR    =  0.017453292
26.   XY(1)  =  (YFACT)(Cos((FE)/(3))) - (P)/(3)
27.   XY(2)  =  (YFACT)(Cos((FE)/(3) + (120)(DTR))) - (P)/(3)
28.   XY(3)  =  (YFACT)(Cos((FE)/(3) + (240)(DTR))) - (P)/(3)
29.   I      =  I + 1
30.   If  I > 3 ,  go to 35
      If  I ≤ 3 ,  go to 31
31.   If  XY(I) ≥ 0 ,  go to 33
      If  XY(I) < 0 ,  go to 32
32.   Go to 29
33.   If  XL < XY(I) ,  go to 29
      If  XL ≥ XY(I) ,  go to 34
34.   XI  =  XY(I)
      Return
35.   CALL SETERR (5HCROOT)
36.   Return
```

SUBROUTINE CROOT NOMENCLATURE

Symbol	Description	Units	Reference
AA	Coefficient of X^3	—	CALL, 3, 4, 5
AP	Terms involving normalized coefficients	—	6, 8, 21, 22
B	Coefficient of X^2	—	CALL, 3
BMSD	—	—	15, 16
BP	Terms involving normalized coefficients	—	7, 8, 10, 11, 14, 15, 22
BPSD	—	—	14, 16
C	Coefficient of X	—	CALL, 4
CFE	Cosine term	—	22, 23, 24
D	Constant term	—	CALL, 5
DK	Term of resolvent equation	—	8, 9, 13
DTR	Conversion factor	radians/degrees	25, 27, 28
FE	Angle defined by CFE and SFE	radians	24, 26, 27, 28
I	Control variable	—	2, 29, 30
P	Normalized coefficient	—	3, 6, 7, 10, 11, 16, 26, 27, 28
Q	Normalized coefficient	—	4, 6, 7
R	Normalized coefficient	—	5, 7

SUBROUTINE CROOT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
SFE	Sine term	—	23, 24
SRTD	Square root of DK	—	13, 14, 15
XI	Desired solution	—	CALL, 17, 34
XL	Maximum value of XI	—	CALL, 19, 33
XX	Argument of function CUBRT	—	1
XY	Root of cubic equation	—	DIM, 10, 11, 12, 16, 17, 18, 19, 26, 27, 28, 31, 33, 34
YFACT	—	—	21, 26, 27, 28

```
$IBFTC CRØØT   FULIST,REF
      SUBRØUTINE CRØØT(AA,B,C,D,XL,XI)
      DIMENSIØN   XY(3)
      CUBRT(XX)=ABS(XX)**.3333333*XX/ABS(XX)
      I=0
   30 P=B/AA
      Q=C/AA
      R=D/AA
      AP= .33333333*(3.0*Q-P**2)
      BP=.03703704*(2.0*P**3-9.0*P*Q+27.0*R)
      DK= AP**3/27. +BP**2/4.
      IF (DK)60,40,50
   40 XY(1)= 2.0*CUBRT(-BP/2.0)-P/3.0
      XY(2)= CUBRT(BP/2.0)-P/3.0
      XY(3)=XY(2)
      GØ TØ 70
   50 SRTD= SQRT(DK)
      BPSD= -BP/2.+SRTD
      BMSD= -BP/2.-SRTD
      XY(1)= CUBRT(BPSD) +CUBRT(BMSD)-P/3.0
      XI=XY(1)
      IF (XY(1) .LT. 0.0) GØ TØ 130
      IF(XL .LT. XY(1)) GØ TØ 130
      GØ TØ 150
   60 YFACT= 2.0*SQRT(-AP/3.0)
      CFE= (-BP/2.0)/SQRT(-AP**3/27.0)
      SFE=SQRT(1.0-CFE**2)
      CALL QUAD (CFE,SFE,FE)
   69 DTR=.017453292
      XY(1)=YFACT* CØS(FE/3.0)-P/3.0
      XY(2)= YFACT* CØS(FE/3.0+120.0*DTR)-P/3.0
      XY(3)= YFACT*CØS(FE/3.0+240.0*DTR)-P/3.0
   70 I=I+1
      IF (I .GT. 3 ) GØ TØ 130
      IF ( XY(I) .GE. 0.0 ) GØ TØ 110
      GØ TØ 70
  110 IF(XL .LT. XY(I)) GØ TØ 70
      XI = XY(I)
      RETURN
  130 CALL SETERR(5HCRØØT)
  150 RETURN
      END
```

FIGURE 42. SYMBOLIC LISTING OF SUBROUTINE CROOT

XXX. SUBROUTINE G1OB

This subroutine performs the coordinate transformation from the local geocentric coordinates of a rotating oblate spheroid to inertial Cartesian coordinates. More specifically, the position and velocity of an object in space, as defined by two angles, the height above the spheroid's surface, and the velocity magnitude and orientation, are transformed to inertial position and velocity components. These components are related to x, y and z axes which form a right-handed orthogonal system, with the x-axis toward the principal direction and the z-axis perpendicular to the fundamental plane. The latter is defined to be the spheroid's equatorial plane. In the case of the Earth, if the x-axis points to the vernal equinox, then the meridian of the object in space is measured by the local sidereal time.

Double precision arithmetic is used internal to this subroutine.

CALLING SEQUENCE

The calling sequence is:

```
CALL G1OB (RE, FLAT, PHI, THETA, H, V, GAMMA,
           AZ, RX, RY, RZ, VX, VY, VZ, M )
```

If control variable M $\neq$ 0, the angles PHI, THETA, GAMMA and AZ are expressed in radians. If M = 0, these values are given in degrees.

SOLUTION METHOD

The following equations are established:

1. $CONV = .01745329252$

2. If $M \neq 0$, $CONV = 1$

   If $M = 0$, $CONV = .01745329252$

The distance of the object from the spheroid's center is computed:

3. $SPHI = \text{Sin}((PHI)(CONV))$

4. $CPHI = \text{Cos}((PHI)(CONV))$

5. $DF = FLAT$

6. $DH = H$

7. $DRE = RE$

8. $R = DH + (DRE)(1 - DF)$

   $$/((1 - ((2)(DF) - (DF^2))(CPHI^2))^{0.5})$$

The Cartesian position coordinates of the object are computed:

9. $STHETA = \text{Sin}((THETA)(CONV))$

10. $CTHETA = \text{Cos}((THETA)(CONV))$

11. $RX = (R)(CPHI)(CTHETA)$

12. $RY = (R)(CPHI)(STHETA)$

13. $RZ = (R)(SPHI)$

The Cartesian velocity coordinates of the object are computed:

```
14.  SGAM  =  Sin((GAMMA)(CONV))

15.  CGAM  =  Cos((GAMMA)(CONV))

16.  SAZ  =  Sin(AZ)(CONV))

17.  CAZ  =  Cos((AZ)(CONV))

18.  DV  =  V

19.  VX  =  (DV)((-CGAM)(CAZ)(SPHI)(CTHETA)

              +  (SGAM)(CPHI)(CTHETA)

              -  (CGAM)(SAZ)(STHETA))

20.  VY  =  (DV)((CGAM)(SAZ)(CTHETA)

              -  (CGAM)(CAZ)(SPHI)(STHETA)

              +  (SGAM)(CPHI)(STHETA))

21.  VZ  =  (DV)((SGAM)(SPHI) + (CGAM)(CAZ)(SPHI))

     Return
```

SUBROUTINE G1OB NOMENCLATURE

Symbol	Description	Units	Reference
AZ	Angle of velocity vector from local meridian North	deg. or radians	CALL, 16, 17
CAZ	Cosine of Azimuth	—	DP, 17, 19, 20, 21
CGAM	Cosine of GAMMA	—	DP, 15, 19, 20, 21
CONV	Conversion factor	deg/rad	DP, 1, 2, 3, 4, 9, 10, 14, 15, 16, 17
CPHI	Cosine of PHI	—	DP, 4, 8, 11, 12, 19, 20, 21
CTHETA	Cosine of THETA	—	DP, 10, 11, 19, 20
DF	Double precision FLAT	—	DP, 5, 8
DH	Double precision H	ft	DP, 6, 8
DRE	Double precision RE	ft	DP, 7, 8
DV	Double precision V	ft/sec	DP, 18, 19, 20, 21
FLAT	Flattening or ellipticity of spheroid	—	CALL, 5
GAMMA	Angle of velocity vector with local horizontal	deg. or radians	CALL, 14, 15
H	Height above reference spheroid	ft	CALL, 6
M	Control variable	—	CALL, 2
PHI	Geocentric latitude	deg. or radians	CALL, 3, 4

SUBROUTINE G1OB NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
R	Distance of object from spheroid's center	ft	DP, 8, 11, 12, 13
RE	Equatorial radius of oblate spheroid	ft	CALL, 7
RX	X-component of R	ft	CALL, 11
RY	Y-component of R	ft	CALL, 12
RZ	Z-component of R	ft	CALL, 13
SAZ	Sine of Azimuth	—	DP, 16, 19, 20
SGAM	Sine of GAMMA	—	DP, 14, 19, 20, 21
SPHI	Sine of PHI	—	DP, 3, 13, 19, 20, 21
STHETA	Sine of THETA	—	DP, 9, 12, 19, 20
THETA	Angle from X-Z plane to local meridian measured counter-clockwise in X-Y plane	deg. or radians	CALL, 9, 10
V	Velocity magnitude	ft/sec	CALL, 18
VX	X-component of velocity	ft/sec	CALL, 19
VY	Y-component of velocity	ft/sec	CALL, 20
VZ	Z-component of velocity	ft/sec	CALL, 21

```
$IBFTC G1ØB    FULIST,REF
      SUBRØUTINE G1ØB(RE,FLAT,PHI,THETA,H,V,GAMMA,AZ,RX,RY,RZ,VX,VY,VZ,
     $           M)
C          GIVEN --
C                1)  ØBLATE SPHERØID DEFINED BY
C                     RE - EQUATØRIAL RADIUS
C                     FLAT - FLATTENING
C                2)  PØSITIØN DEFINED BY
C                     PHI - GEØCENTRIC LATITUDE
C                     THETA - ANGLE FRØM X-Z PLANE TØ LØCAL MERIDIAN
C                            MEASURED CØUNTER-CLØCKWISE IN X-Y PLANE
C                            (I.E. LØCAL SIDEREAL TIME IF X-AXIS PØINTS
C                            TØ VERNAL EQUINØX)
C                     H - HEIGHT ABØVE REFERENCE SPHERØID
C                3)  VELØCITY DEFINED BY
C                     V - VELØCITY MAGNITUDE
C                     GAMMA - ANGLE ØF VELØCITY FRØM LØCAL HØRIZØNTAL
C                     AZ - ANGLE ØF VELØCITY FRØM LØCAL MERIDIAN NØRTH
C          CALCULATES --
C               1)  CARTESIAN CØMPØNENTS ØF PØSITIØN DEFINED BY
C                     RX - X CØMPØNENT ØF R
C                     RY - Y CØMPØNENT ØF R
C                     RZ - Z CØMPØNENT ØF R
C                2)  CARTESIAN CØMPØNENTS ØF VELØCITY DEFINED BY
C                     VX - X CØMPØNENT ØF V
C                     VY - Y CØMPØNENT ØF V
C                     VZ - Z CØMPØNENT ØF V
C          IF M=0, PHI, THETA, GAMMA, AZ IN DEGREES INSTEAD ØF RADIANS
C
      DØUBLE PRECISIØN  CØNV,SPHI,CPHI,STHETA,CTHETA,SGAM,CGAM,SAZ,CAZ,
     $         DF,DH,DRE,DV,R
C
      CØNV = .01745329252
      IF (M .NE. 0)  CØNV = 1.0D0
C          CALCULATE SEA LEVEL RADIUS + H TØ GET R
      SPHI = DSIN(PHI*CØNV)
      CPHI = DCØS(PHI*CØNV)
      DF = FLAT
      DH = H
      DRE = RE
      R = DH + DRE * (1.0D0 - DF)/DSQRT(1.0D0 - (2.0D0*DF - DF**2)
     $                                                     * CPHI**2)
C
      STHETA = DSIN(THETA*CØNV)
      CTHETA = DCØS(THETA*CØNV)
C          CØMPØNENTS ØF R
      RX = R * CPHI * CTHETA
      RY = R * CPHI * STHETA
      RZ = R * SPHI
C
```

FIGURE 43. SYMBOLIC LISTING OF SUBROUTINE G1OB

```
      SGAM = DSIN(GAMMA*CØNV)
      CGAM = DCØS(GAMMA*CØNV)
      SAZ = DSIN(AZ*CØNV)
      CAZ = DCØS(AZ*CØNV)
C          CØMPØNENTS ØF V
      DV = V
      VX = DV * (-CGAM * CAZ * SPHI * CTHETA + SGAM * CPHI * CTHETA
     $             - CGAM * SAZ * STHETA)
      VY = DV * (CGAM * SAZ * CTHETA - CGAM * CAZ * SPHI * STHETA
     $             + SGAM * CPHI * STHETA)
      VZ = DV * (SGAM * SPHI + CGAM * CAZ * CPHI)
      RETURN
      END
```

FIGURE 43. SYMBOLIC LISTING OF SUBROUTINE G1OB (cont.)

XXXI. SUBROUTINE G2OB

Given a rotating oblate spheroid, and the position and velocity of an object in space, as referenced to an inertial Cartesian coordinate system, this subroutine will determine the object's height above the spheroid's surface, its position in terms of the local geocentric coordinates, and its velocity magnitude and orientation.

The right-handed orthogonal Cartesian coordinate system has its x-axis toward the principal direction and its z-axis perpendicular to the spheroid's equatorial plane. In the case of the Earth, if the x-axis points to the vernal equinox, then the meridian of the object in space is measured by the local sidereal time.

This program calls Subroutine QUAD for the determination of the value of an angle, given its sine and cosine. The arcsine and arctangent functions, ASIN and ATAN, are also used.

Double precision arithmetic is used internal to this subroutine.

CALLING SEQUENCE

The calling sequence is:

```
CALL G2OB (RX, RY, RZ, VX, VY, VZ, RE, FLAT, PHI,
           THETA, H, V, GAMMA, AZ, M)
```

If the control variable M = 0, PHI, THETA, GAMMA and AZ are measured in degrees. If $M \neq 0$, these values are measured in radians.

SOLUTION METHOD

The following function is defined:

1. $\mathrm{DASIN(X) = DSIGN\ (DATAN((X)/(1 - X^2)^{.5}),\ X)}$

   DATAN is a FORTRAN double precision arctangent function. DSIGN is a FORTRAN double precision function which transfers the sign of the second argument to the first argument.

2. CONV = 57.2957795

3. If $M \neq 0$, CONV = 1

   If M = 0, CONV = 57.2957795

4. DRX = RX

5. DRY = RY

6. DRZ = RZ

The object's distance from the spheroid's center is determined:

7. $\mathrm{R = (DRX^2 + DRY^2 + DRZ^2)^{0.5}}$

Values are determined for the geocentric latitude and its sine and cosine:

8. DPHI = DASIN ((DRZ)/(R))

9. CPHI = Cos (DPHI)

10. SPHI = Sin (DPHI)

Values are determined for the angle THETA and its sine and cosine:

11. STHETA = (DRY)/((R)(CPHI))

12. CTHETA = (DRX)/((R)(CPHI))

13. CALL QUAD (CTHETA, STHETA, THETA)

The object's height above the spheroid's surface, and its velocity, are computed:

14. DRE = RE

15. DF = FLAT

16. $H = R - (DRE)(1 - DF)/((1 - ((2)(DF)$

$$- (DF^2))(CPHI^2))^{0.5}$$

17. DVX = VX

18. DVY = VY

19. DVZ = VZ

20. $DV = (DVX^2 + DVY^2 + DVZ^2)^{0.5}$

21. V = DV

The north, east and down components of the velocity are computed:

22. VN = (-DVX)(CTHETA)(SPHI)

- (DVY)(STHETA)(SPHI) + (DVZ)(CPHI)

23. VE = (-DVX)(STHETA) + (DVY)(CTHETA)

24. VD = (-DVX)(CTHETA)(CPHI)

- (DVY)(STHETA)(CPHI) - (DVZ)(SPHI)

Values are determined from the flight path and azimuth angles:

25. If | (VD)/(DV) | ≥ .9999998, go to 34

If | (VD)/(DV) | < .9999998, go to 26

26. DGAMMA = DATAN(((-VD)/ (DV))/(1 - $((VD)/(DV))^2)^{.5}$)

27. SAZ = (VE)/((DV)(Cos(DGAMMA)))

28. CAS = (VN)/((DV)(Cos(DGAMMA)))

29. CALL QUAD (CAZ, SAZ, AZ)

The angular measurements are converted:

30. PHI = (DPHI)(CONV)

31. THETA = (THETA)(CONV)

32. GAMMA = (DGAMMA)(CONV)

33. AZ = (AZ)(CONV)

Return

If | VD | is equal to | V |, the vehicle is in vertical flight.

34. DGAMMA = DSIGN ((90)(.01745329252), -VD)

35. AZ = 0

36. Go to 30

SUBROUTINE G2OB NOMENCLATURE

Symbol	Description	Units	Reference
AZ	Angle of velocity vector from local meridian north	deg. or radians	CALL, 29, 33, 35
CAZ	Cosine of Azimuth	—	28, 29
CONV	Conversion factor	deg/rad	DP, 2, 3, 30, 31, 32, 33
CPHI	Cosine of PHI	—	DP, 9, 11, 12, 16, 22, 24
CTHETA	Cosine of THETA	—	DP, 12, 13, 22, 23, 24
DF	Double precision FLAT	—	DP, 15, 16
DGAMMA	Double precision GAMMA	radians	DP, 26, 27, 28, 32, 34
DPHI	Double precision PHI	radians	DP, 8, 9, 10, 30
DRE	Double precision RE	ft	DP, 14, 16
DRX	Double precision RX	ft	DP, 4, 7, 12
DRY	Double precision RY	ft	DP, 5, 7, 11
DRZ	Double precision RZ	ft	DP, 6, 7, 8
DV	Double precision V	ft/sec	DP, 20, 21, 25, 26, 27, 28
DVX	Double precision VX	ft/sec	DP, 17, 20, 22, 23, 24
DVY	Double precision VY	ft/sec	DP, 18, 20, 22, 23, 24
DVZ	Double precision VZ	ft/sec	DP, 19, 20, 22, 24
FLAT	Flattening or ellipticity of spheroid	—	CALL, 15

SUBROUTINE G2OB NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
GAMMA	Angle of velocity vector with local horizontal	deg. or radians	CALL, 32
H	Height of object above reference spheroid	ft	CALL, 16
M	Control variable	—	CALL, 3
PHI	Geocentric latitude	deg. or radians	CALL, 30
R	Distance of object from spheroid's center	ft	DP, 7, 8, 11, 12, 16
RE	Equatorial radius of oblate spheroid	ft	CALL, 14
RX	X-component of R	ft	CALL, 4
RY	Y-component of R	ft	CALL, 5
RZ	Z-component of R	ft	CALL, 6
SAZ	Sine of Azimuth	—	27, 29
SPHI	Sine fo PHI	—	DP, 10, 22, 24
STHETA	Sine of THETA	—	DP, 11, 13, 22, 23, 24
THETA	Angle from X-Z plane to local meridian measured counter-clockwise in X-Y plane	deg. or radians	CALL, 13, 31
V	Velocity magnitude	ft/sec	CALL, 21
VD	Down component of velocity	ft/sec	DP, 24, 25, 26, 34

SUBROUTINE G2OB NOMENCLATURE (cont.)

Symbol	Description	Units	Description
VE	East component of velocity	ft/sec	DP, 23, 27
VN	North component of velocity	ft/sec	DP, 22, 28
VX	X-component of velocity	ft/sec	CALL, 17
VY	Y-component of velocity	ft/sec	CALL, 18
VZ	Z-component of velocity	ft/sec	CALL, 19

```
$IBFTC G2OB    FULIST,REF
      SUBROUTINE G2OB(RX,RY,RZ,VX,VY,VZ,RE,FLAT,PHI,THETA,H,V,GAMMA,AZ,
     $           M)
C      GIVEN --
C          1)  CARTESIAN POSITION R DEFINED BY
C                RX - X COMPONENT OF R
C                RY - Y COMPONENT OF R
C                RZ - Z COMPONENT OF R
C           2)  CARTESIAN VELOCITY V DEFINED BY
C                VX - X COMPONENT OF V
C                FLAT - FLATTENING
C      CALCULATES --
C           1)  POSITION DEFINED BY
C                PHI - GEOCENTRIC LATITUDE
C                THETA - ANGLE FROM X-Z PLANE TO LOCAL MERIDIAN
C                       MEASURED COUNTER-CLOCKWISE IN X-Y PLANE
C                       (I.E. LOCAL SIDEREAL TIME IF X-AXIS POINTS
C                       TO VERNAL EQUINOX)
C                H - HEIGHT ABOVE REFERENCE SPHEROID
C           2)  VELOCITY DEFINED BY
C                V - VELOCITY MAGNITUDE
C                GAMMA - ANGLE OF VELOCITY VECTOR FROM LOCAL HORIZONTAL
C                AZ - ANGLE OF VELOCITY VECTOR FROM LOCAL MERIDIAN NORTH
C      IF M=0, PHI, THETA, GAMMA, AZ IN DEGREES INSTEAD OF RADIANS
C                VY - Y COMPONENT OF V
C                VZ - Z COMPONENT OF V
C           3)  OBLATE SPHEROID DEFINED BY
C                RE - EQUATORIAL RADIUS
C
      DOUBLE PRECISION  DASIN,CONV,R,DPHI,CPHI,SPHI,STHETA,CTHETA,DV,
     $          VN,VE,VD,DGAMMA,X,DRX,DRY,DRZ,DRE,DF,DVX,DVY,DVZ
C
      DASIN(X) = DSIGN(DATAN(X/DSQRT(1.0D0-X**2)),X)
C
      CONV = 57.2957795D0
      IF (M .NE. 0)  CONV = 1.0D0
      DRX = RX
      DRY = RY
      DRZ = RZ
      R = DSQRT(DRX**2 + DRY**2 + DRZ**2)
C          CALCULATE PHI
      DPHI = DASIN(DRZ/R)
      CPHI = DCOS(DPHI)
      SPHI = DSIN(DPHI)
C          CALCULATE THETA
      STHETA = DRY/(R * CPHI)
      CTHETA = DRX/(R * CPHI)
      CALL  QUAD(CTHETA,STHETA,THETA)
C          CALCULATE H AND V
      DRE = RE
```

FIGURE 44. SYMBOLIC LISTING OF SUBROUTINE G2OB

```
      DF = FLAT
      H = R - DRE * (1.0D0 - DF)/DSQRT(1.0D0 - (2.0D0*DF - DF**2)
     $                                                 * CPHI**2)
      DVX = VX
      DVY = VY
      DVZ = VZ
      DV = DSQRT(DVX**2 + DVY**2 + DVZ**2)
      V = DV
C          CALCULATE NØRTH, EAST AND DØWN CØMPØNENTS ØF V
      VN = -DVX * CTHETA * SPHI - DVY * STHETA * SPHI + DVZ * CPHI
      VE = -DVX * STHETA + DVY * CTHETA
      VD = -DVX * CTHETA * CPHI - DVY * STHETA * CPHI - DVZ * SPHI
C          CALCULATE GAMMA AND AZIMUTH
      IF (DABS(VD/DV) .GE. .9999998D0)  GØ TØ 20
      DGAMMA = DATAN((-VD/DV)/DSQRT(1.0D0 - (VD/DV)**2))
      SAZ = VE/(DV * DCØS(DGAMMA))
      CAZ = VN/(DV * DCØS(DGAMMA))
      CALL QUAD(CAZ,SAZ,AZ)
C
   10 PHI = DPHI * CØNV
      THETA = THETA * CØNV
      GAMMA = DGAMMA * CØNV
      AZ = AZ * CØNV
      RETURN
C        IF ABS(VD) = ABS(V)  -  GAMMA = 90.0 DEGREES
   20 DGAMMA = DSIGN(90.0D0*.01745329252,-VD)
      AZ = 0.0
      GØ TØ 10
      END
```

FIGURE **44**. SYMBOLIC LISTING OF SUBROUTINE G2OB (cont.)

XXXII. SUBROUTINE FINPUT

This subroutine reads a set of Hollerith-punched data and/or header cards into core with one CALL statement using the FORTRAN IV input routine FRWD. It converts the data fields to binary and stores them in core according to their associated conversion codes. There are 2 entry points to this routine, FINP and FINK.

CALLING SEQUENCES

Entry FINP has two type of CALL statements:

1. CALL FINP (IUNIT, OUNIT, EFLAG, ±n, X, Y, ZETA, . . . ,

   mHX⑤ Y⑤ ZETA② . . .) where:

   a. IUNIT specifies the FORTRAN symbolic unit for the input card images. If IUNIT = 0, cards are read on-line.

   b. OUNIT specifies the FORTRAN symbolic unit for the output card listing. If OUNIT = 0, cards are listed on-line. If OUNIT < 0, cards are listed both on-tape and on-line.

c. EFLAG is a logical error flag. If an error is encountered, this is set .TRUE. and successive cards will be read and processed, but not stored. When an END card is encountered, control returns to the calling program. If EFLAG is .FALSE. upon entry to FINPUT, no data will be stored.

d. n is the number of variables and/or arrays in the list, excluding n itself. If $n < 0$, cards will be listed; if $n > 0$, input cards will not be listed. The n^{th} variable is assumed to be the origin for header cards (see HEAD under CALL FINK).

e. X, Y, ZETA, ... are the names of variables and/or arrays whose Hollerith names are restricted to at most 5 characters each, one character of which is non-numeric.

f. m is 6 times $|n|$. Hence, mH allows for 6n Hollerith characters to follow.

g. X⑤ Y⑤ ZETA② ... is a list of Hollerith names of the items previously named in exactly the same order with ⓘ indicating the number, i, of blanks necessary to provide six Hollerith characters for each item. Since each name is restricted to 5 characters, the minimum value of ⓘ is ① .

2. CALL FINP (IUNIT, OUNIT, EFLAG, 0) where the number of items is given as zero. This CALL statement must be used only after a CALL statement of type 1 has been executed. When the subroutine encounters a zero for the number of items, it immediately refers to the last executed CALL FINP with a non-zero number of items for the names of the items to be loaded.

Entry FINK has only one type of call:

CALL FINK (IUNIT, OUNIT, EFLAG, TABLE, HEAD) where:

a. IUNIT, OUNIT, and EFLAG are as described above.

b. HEAD refers to the origin of the array where header cards are to be stored.

c. TABLE is the origin of a table of symbolic information of the form:

```
TABLE  ***  N          ***  = MZE lists card images
       BCI  1,XXXXX         = PZE  no list
       AAA  YYYYY
```

where N is the number of words in the table excluding the word containing N, and XXXXX is any collection of 1 to 5 alphanumeric characters, left justified, at least one of which must be non-numeric. When conversion of XXXXX is encountered, the table is searched for this symbol and the contents of the next location used as the converted value.

The same table is used for symbolic locations and symbolic data. When used with locations, the low order 15 bits of the word following the word containing the symbol is used. For a typical entry of this type AAA=PZE and YYYYY= XXXXX where XXXXX is some symbolic location. When the table is used with symbolic data defined by conversion code A, loading occurs as described under A code.

There must always be at least one entry in TABLE, i.e., the location of the data block.

If cards are listed, a page is ejected before listing the first card image. Where only a few items are to be read in as input, the FINP call will be easier to use. But, where more data is to be read in than can be identified in the FINP call, which is limited to 19 continuation cards in FORTRAN programs, or if "bucket" input is desired, the FINK call should be used.

USAGE

Card Format

The data card format consists of four data fields each of which is composed of four subfields containing the conversion code, location, value, and exponent, respectively (see Figure 45). The conversion code is one of the alphabetic characters defined below which specifies the type of conversion to be used on the value field; the location specifies an actual, array subscript, or symbolic cell into which the converted value field is to be stored; the value subfield contains the data to be converted, and the exponent contains the power of ten by which floating data is to be scaled or the location of the binary point of fixed point data.

The header card format consists of a conversion code in Column 1, a sequence number in Columns 2-6 and any Hollerith information in Columns 7-72.

Decimal Points

Decimal points may be placed anywhere in the value field except that they may not occur in the same column as a minus sign (11 punch) since this results in a non-Hollerith character. If the decimal point would normally appear at the right of the number punched in the value field, then it is optional.

Minus Signs

Minus signs are 11 punches over any digit of the field. If all of the available columns of the field are not used, minus signs may be punched as the left character of the field.

Plus Signs

Plus signs are 12 punches. They are optional but may not be punched over a digit or decimal point.

Commas

Commas in the value field are ignored for decimal conversion options.

	Card Columns			
	Data Field 1	Data Field 2	Data Field 3	Data Field 4
Subfield				
Conversion code	1	19	37	55
Location	2-6	20-24	38-42	56-60
Value	7-16	25-34	43-52	61-70
Exponent	17-18	35-36	53-54	71-72

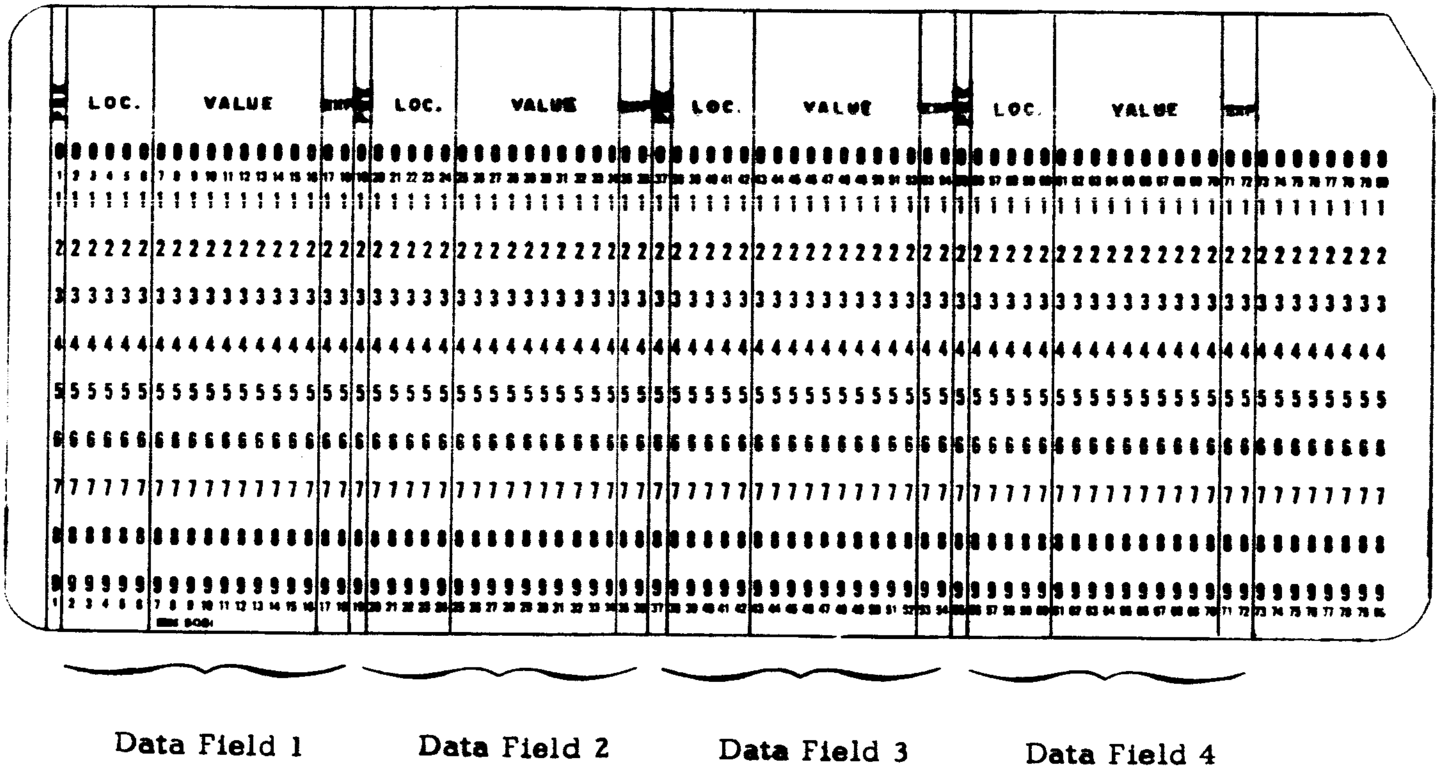


FIGURE 45. CARD FORMAT

Values

Values must always be written to the extreme left of a field, except for numerical data which will be left justified before conversion. It is not necessary that the entire field be filled, as the first blank denotes the end of value. Superfluous low-order zeros will be edited out.

The only exception to partial fields is BCI data where the entire field, including blanks, is stored.

Location

The location may be either absolute octal, array subscript, symbolic, or matrix i, j notation. If the location contains five digits, it is interpreted as octal. If the location contains four or fewer digits, it is interpreted as an array subscript, and the actual location is found relative to the current origin. This rule does not apply to H cards. It should be noted that negative subscripts are not permissible. All five columns must be punched for octal locations. Only significant digits need be punched for array subscripts, i.e., zeros on the left need not be punched. However, the location must be punched to the extreme left of the location field. If the location is left blank, then the location counter is increased by 1 and the associated number is stored in the cell immediately following the cell where the last number was stored. Thus, an entire array may be read in by specifying the initial location only.

If the location contains at least 1 non-numeric character other than a comma in the second, third, or fourth position, it is interpreted as symbolic. The contents of the number and exponent fields are stored in the location associated with the symbolic name in the symbol table or call statement. This location then becomes the current origin for all succeeding array subscripts until another symbolic location is encountered. Caution must be taken to load a current origin definition prior to array subscripts.

If the location field contains a comma in the second, third, or fourth position, it is interpreted as matrix i, j notation. The location field is defined to consist of three subfields containing i, a comma, and j respectively. The i and j subfields must contain at least one digit decimal integer written to the left of the subfields. The contents of the value and exponent fields are stored in the (i, j) array subscript of the current origin. If the subscript exceeds the previously defined matrix size (i_{max} , j_{max}) an error results.

Conversion Codes

Blank: Floating Decimal

The number in the value field times the power of ten in the exponent field is converted to floating binary. Checks are made for overflow and format errors. If a format error is detected, an attempt at A-type conversion is made before an error is flagged.

F: Fixed Decimal

The number in the value field is converted to fixed point binary and stored with the binary point located at the position specified by the number in the exponent field. An overflow error check is made.

I: Decimal Integer

The number in the value field is converted to a fixed point binary integer. The exponent field is ignored.

B: Octal

The value plus exponent fields are converted as a logical octal word.

It is not necessary to include leading zeros but the first octal digit must always occupy the left most position of the field.

D: BCI Data

The contents of the value plus exponent fields are interpreted as two BCI words and stored in two consecutive cells beginning at the location specified by the location field.

H: Heading Card

A card with an H in Column 1 is considered a BCI heading card. The location field contains the sequence number for storing the heading card. This number must be a positive integer I with a maximum value of four digits. Each heading card (Columns 7-72) are stored in 11 consecutive words beginning at HEAD (11 · I - 10).

A: Symbolic

The value plus exponent fields are interpreted in a MAP instruction format AAAAA T DDDDD P where the fields to replace are address, tag, decrement and prefix, respectively. The address and decrement fields are defined normally to be 5 characters and the tag and prefix, one character. Any field containing less than the normal number of characters must end with a comma while fields of normal length must not. Any field containing less than 5 characters, all of which are numeric, is converted as decimal, while any field of 5 all-numeric characters is converted as octal. Fields containing at least one non-numeric character are interpreted as symbolic. Symbolic addresses load the entire word from TABLE or CALL into the location word. Symbolic decrement fields load the low-order 18 bits from TABLE or CALL into the location decrement-prefix. Symbolic tag and prefix fields replace the high-order 3 bits of the respective address-tag and decrement-prefix fields. Null fields are not loaded. Thus, it is possible to load most any desired word by careful combination of input and table formats. In particular, 18-bit instruction codes can be loaded by writing a symbolic notation in the decrement and omitting a prefix if the FINK-type CALL is used.

G: Temporary Origin

The value in the location field (either symbolic, array subscript, or actual octal) is used as a temporary origin for tables. The location is saved and if data cards follow with blank location fields, the corresponding data is stored consecutively beginning with the cell specified in the location in the G card. The first non-blank location starts a new origin.

If this non-blank is a subscript, it references the last variable or array named.

Columns 7-72 are ignored and may be used to identify the table.

K: Omit Loading the Next N Cells

The integer N in the value field is used to increment the next storage location by N.

This conversion code is available only when the FINK-type CALL is used.

C: Preserve the Location of the Next Storage Cell

The location of the next available sequential storage cell is stored into the TABLE of symbolic information in the address of the word following the word containing the BCI representation of the symbol in the value field.

This conversion code is available only when the FINK-type CALL is used.

J: Transfer

The location specified with this prefix is the only part of the data field that is interpreted. The subroutine causes a transfer to the location specified and does not interpret the remaining fields on the card.

L: Two-Dimension Array i_{max}, j_{max} Definition

The location field contains the name of the array to be loaded. The value field is defined to consist of 2 subfields, separated by a comma, containing the decimal integers for i_{max} and j_{max} respectively where i_{max} and j_{max} generally appear in a DIMENSION statement. The i_{max} and j_{max} values are retained to compute the successive subscripted locations until redefined. Blank address fields may follow this array definition if successive elements of the array are to be loaded.

M: Two-Dimension Array i_{max}, j_{max} Definition

Conversion is identical to L except the entire array is preset to zero.

E: End Case

This defines an end-of-case and control is returned to the FORTRAN object program. The rest of this field and the remaining fields on the card are ignored.

EXPLANATORY NOTE

"Bucket" input is a space-saving device whereby the dimensions of an array are not defined until the input is read in. FORTRAN coding using bucket input is slightly different than normal FORTRAN coding, as these arrays cannot be manipulated directly in FORTRAN source language. A MAP driver is necessary for routines using "bucket" tables.

For bucket input, the symbol table should be a control block in Named COMMON. Those inputs which go in the bucket are identified in the symbol table as follows:

```
BCIPZE      CONTRL      BCIPZE
            USE         BCIPZE
            EVEN
TABLE       PZE         6
            BCI         1,XTABLE   )
XTABLE      PZE         **         }  In Bucket
            BCI         1,YTABLE   }
YTABLE      PZE         **         )
            BCI         1,Z        }  Not in Bucket
            PZE         Z          }
```

The FORTRAN routines which use XTABLE and YTABLE then find them in COMMON as follows:

```
COMMON/BCIPZE/TABLE(2), XTABLE(2), YTABLE(3)
```

Instead of XTABLE and YTABLE being the start of the XTABLE and YTABLE arrays, they give the location of the actual beginning of the arrays.

The FORTRAN Program must set aside a large array to be used for bucket input. The name of this array and its location must be in the FINK symbol table. At input time before any data can be read into the bucket, the tables to be read into the bucket must first be dimensioned.

Dimensioning is accomplished as follows:

a. Set the storage location counter to the first available location in the bucket by means of a "G" card.

b. Store this location in the FINK symbol table by means of a "C" card(s) which also gives the name(s) of the table to be stored here. (If two or more tables contain the same values, only one table need be input if the table names are defined to be equivalent by C cards.)

c. Increment the storage location counter by means of a "K" card which also gives the number of locations to skip.

d. Repeat Steps b and c until all tables to be read into the bucket have locations assigned.

e. Read in input as normal.

When using bucket input, it may also be necessary to read in an array of dimensions corresponding to those defined by "C" and "K" cards if the program needs to know the dimensions of arrays in the bucket. The "C" and "K" prefix conversion codes do not transmit data to the calling program. They only set up the FINK symbol table.

```
$IBMAP  FINPUT
*       CALL     FINP(INPT,ØTPT,FLAG,N,A,B,. .,6NANAM BNAM ...)
*                INPT - INPUT UNIT  0-ØN-LIN
*                ØTPT - ØUTPUT UNIT  0-ØN-LIN  -I-I AND ØN-LIN
*                N - NUMBR ØF ITMS IN LIST
*                A,B,C - ITMS IN LIST --LAST IS HADR BLØCK
*                ANAM,BNAM,CNAM - HØLLRITH NAMES ØF ITEMS IN LIST
*
FINP    SAVE     (4,1,2)I              ENTRY FØR LØNG FØRTRAN CALL
        CLA*     6,4
        STØ      CØMMØN+28           FLAG TØ USE PREVIØUS CALL
        TZE      INPT2
        SXD      FINPT,4
INPT2   LXD      FINPT,4
        CLA      3,4
        STA      LINPUT
        CLA      4,4
        STA      LØUTPT
        CLA      5,4
        STA      GIN13+1
        CLA*     5,4
        STØ      ERRØR
        PXA      ,4
        SBM*     6,4
        ADD      M6B35
        CØM
        STA      TBL02                 ALPHA + N
        CLA      6,4
        STA      TBL00
        CLA*     6,4
        STØ      CØMMØN+29             N
        ALS      18
        CØM
        STD      *+1
        TXI      *+1,4,**
        CLA      5,4                   LAST ITEM IN LIST
        STA      GIN01                 ADDRESS ØF HEADER
        CLA      6,4
        ADM      CØMMØN+29
        STA      TBL01
        STZ      FINKY
        TRA      GINZ
*
*       CALL     FINK(INPT,ØTPT,EFLAG,TABLE,HEADER)
*                INPT - INPUT UNIT  0-ØN-LINE
*                ØTPT - ØUTPUT UNIT  0-ØN-LIN  -I-I AND ØN-LINE
*                TABL - TABLE ØF NAMES AND ITEMS
*                HEADER - HEADER BLØCK
FINK    SAVE     (4,1,2)I              ENTRY FØR MAP TABLE ØPTIØN AND BUCKET
        CLA      3,4
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT

```
        STA     LINPUT
        CLA     4,4
        STA     LØUTPT
        CLA     5,4
        STA     GIN13+1
        CLA*    5,4
        STØ     ERRØR
        CLA     6,4
        STA     TBLØØ
        ADM*    6,4
        ADD     1B35
        STA     TBLØ1
        STA     TBLØ2
        CLA     7,4
        STA     GINO1
        CLA*    6,4
        STØ     CØMMØN+29
        STØ     FINKY
*
GINZ    CLA     CØMMØN+29
        TPL     GINA
        CAL*    LØUTPT
        STØ     ØUNIT
        TZE     GINZ1
        CALL    .FVIØ.(ØUNIT,ØTAPE)
        CALL    .FWRD.(ØTAPE,=H(A6)    )
        CLA     =H1
        TSX     .FCNV.,4
        TSX     .FFIL.,4
        CLA*    LØUTPT
        TPL     GINA
GINZ1   CALL    .FPRN.(=H(A6)    )
        CLA     =H1
        TSX     .FCNV.,4
        TSX     .FFIL.,4
*       READ INPUT CARD
GINA    CLA     CØUNT
        ADD     1B35
        STØ     CØUNT
        CAL*    LØUTPT
        STØ     ØUNIT
        CAL*    LINPUT
        STØ     IUNIT
        TZE     GINA2
        CALL    .FVIØ.(IUNIT,ITAPE)
        CALL    .FRDD.(ITAPE,=H(14A6))
        AXT     14,1
        TSX     .FCNV.,4
        STØ     CØMMØN+14,1
        TIX     *-2,1,1
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
         TSX      .FRTN.,4
         CLA      =H00
         ØRS      CØMMØN+13
         TRA      GINA3
GINA2    CALL     .FRCD.(=H(12A6))
         AXT      12,1
         TSX      .FCNV.,4
         STØ      CØMMØN+12,1
         TIX      *-2,1,1
         TSX      .FRTN.,4
         STZ      CØMMØN+13
         STZ      CØMMØN+14
GINA3    CLA      CØMMØN+29
         TPL      GINA1                 NØ INPUT LIST
*        WRITE CARD IMAGE ØN ØUTPUT TAP
         NZT      ØUNIT
         TRA      GINA4
         CALL     .FVIØ.(ØUNIT,ØTAPE)
         CALL     .FWRD.(ØTAPE,FØRMAT)
         CLA      =H
         TSX      .FCNV.,4
         AXT      14,1
         CLA      CØMMØN+14,1
         TSX      .FCNV.,4
         TIX      *-2,1,1
         CLA      CØUNT
         TSX      .FCNV.,4
         TSX      .FFIL.,4
         CLA*     LØUTPT
         TPL      GINA1
GINA4    CALL     .FPRN.(FØRMAT)
         CLA      =H
         TSX      .FCNV.,4
         AXT      14,1
         CLA      CØMMØN+14,1
         TSX      .FCNV.,4
         TIX      *-2,1,1
         CLA      CØUNT
         TSX      .FCNV.,4
         TSX      .FFIL.,4
*        CHECK FØR BLANK CARD
GINA1    AXC      CØMMØN,1
         TXI      *+1,1,-12
         SXD      GINC,1
         SXD      GIN11+1,1
         AXC      CØMMØN,1
         CLA      =H
GINB     CAS      0,1
         TRA      GINC+2                NØT BLANK
         TXI      *+2,1,-1
         TRA      GINC+2                NØT BLANK
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
GINC    TXH     GINB,1,**
        TRA     GINA
*      START  INPUT CØNVERSIØN
        AXC     CØMMØN,1
        LDQ     0,1
GIN01   PXD     **,0                    SAVE PREFIX CØDE
        LGL     6
        STØ     PREFIX
        STQ     0,1
        SUB     =H00000E
        TZE     GIN13                   END
        LGL     30                      SAVE LAST DIGIT ØF ADDRESS
        ANA     =Ø77
        STØ     ADDIND
        STZ     SUM                     TEST FØR BLANK ADDRESS
        STZ     SIGN
        CLA     0,1
        CAS     =H     0
        TRA     *+2
        TRA     GIN06                   BLANK
        LDQ     0,1                     DETERMINE CØMMA PØSITIØN
        LGL     6
        AXT     2,4
        PXD     ,0
        LGL     6
        SUB     =H00000,
        TZE     *+3
        TXH     GIN17-2,4,3             NØT MATRIX
        TXI     *-5,4,1
        PXA     ,4                      PØSITIØN ØF CØMMA
        TSX     MTX00,4
        TRA     GINER                   ØUTSIDE MATRIX
        TRA     GIN20
        AXT     5,4
        LDQ     0,1
GIN17   PXD     ,0
        LGL     6
        PAX     ,2
        TXL     GIN18,2,9
        TXH     GIN19,2,48
        TXL     GIN19,2,47
GIN18   TIX     GIN17,4,1
        TRA     GIN16
GIN19   CLA     0,1       SYMBØLIC
        ØRA     =H00000
        TSX     TBL,4
        LLS     35
        STA     GIN00
GIN20   CLA     =H00000
        STØ     ADDIND
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        TRA     GIN07
GIN16   CLA     ADDIND              TEST FØR ØCTAL
        SUB     =H00000
        TZE     GIN04
        AXT     5,2                 SET UP ØCTAL ADDRESS
        TSX     ØCTAL,4
        TXH     5,,**
GIN03   TXL     GIN07,,**
GIN04   AXT     100,2               SET UP DECIMAL ADDRESS
        CLA     4B35
        TSX     DECIM,4
        TRA     GINER
        TXH     6,,**
        TXL     GINER,2,100         ERRØR RETURN
        TRA     GIN07
GIN06   LDQ     =H     0
GIN07   STQ     CRDADD
        LXA     PREFIX,4            SET UP TRANSFER TØ SUBPRØGRAM
        TXH     GINER,4,48
        TXL     GIN08,4,47
        AXT     0,4
        TRA     GIN09
GIN08   TXH     GINER,4,36
        TXL     *+2,4,32
        TXI     GIN09,4,-23
        TXH     GINER,4,25
        TNX     GINER,4,16
GIN09   TXI     *+1,1,-1
        TRA     GIN10,4
        TRA     MCD00               M - MATRIX  IJ ZERØ
        TRA     MCD00               L - MATRIX  IJ NØ ZERØ
        TRA     KGIN00              DIMENSIØN TABLE
        TRA     JGIN00              J -- TRANSFER
        TRA     IGIN00              I - INTEGER
        TRA     HGIN00              H - HEADER
        TRA     GGIN00              G - TABLE ØRIGIN
        TRA     FGIN00              F - FIXED PØINT
        TRA     GINER               E
        TRA     DGIN00              D - BCD
        TRA     CGIN00              SAVE LØCATIØN CØUNTER FØR BUCKET
        TRA     BGIN00              B - ØCTAL
        TRA     ACD0                A - SYMBØLIC
GIN10   TXL     LGIN00,,12          BLANK - FLØATING
GIN11   TXI     *+1,1,-1            TEST END CARD
        TXH     GIN01-1,1,**
GIN12   TRA     GINA                TØ NEXT CARD
GIN13   CLA     ERRØR
        STØ     **
GINY    ZET     FINKY
        RETURN  FINK
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        RTURN   FINP
GINER   STL      ERRØR
        NZT      ØUNIT
        TRA      GINERZ
        CALL     .FVIØ.(ØUNIT,ØTAPE)
        CALL     .FWRD.(ØTAPE,=H(14A6))
        AXT      10,1
        CLA      =H
        TSX       FCNV.,4
        TIX      *-2,1,1
        AXT      4,1
        CLA      GINRC+4,1
        TSX      .FCNV.,4
        TIX      *-2,1,1
        TSX      .FFIL.,4
        CLA*     IØUTPT
        TMI      GINERZ
        TRA      GIN12
GINERZ  CALL     .FPRN.(=H(14A6))
        AXT      10,1
        CLA      =H
        TSX      .FCNV.,4
        TIX      *-2,1,1
        AXT      4,1
        CLA      GINRC+4,1
        TSX      .FCNV.,4
        TIX      *-2,1,1
        TSX      .FFIL.,4
        TRA      GIN12
GINRC   BCI      4,               CARD IN ERRØR
*     B - ØCTAL CØNVERSIØN
BGINOO  STZ      SUM
        STZ      SIGN
        AXT      6,2
        TSX      ØCTAL,4
        TXI      BGINO4,1,-1
        TXI      BGINO1,1,-1
BGINO1  AXT      6,2
        TSX      ØCTAL,4
        TXH      4,0,0
BGINO2  TSX      SGINOO,4
BGINO3  TXL      GIN11,,**              EXIT FØR NEW PREFIX
BGINO4  TXH      BGINO3,2,5             BLANK
        TRA      BGINO2
*     ØCTAL CØNVERSIØN
ØCTAL   LDQ      0,1                    SET-UP ØCTAL INTEGER
ØCTAL1  PXD      ,0
        LGL      3
        TNZ      ØCTAL2
        CAL      SUM
        LGL      3
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        SLW     SUM
        TIX     ØCTAL1,2,1
        TXI     ØCTAL3,4,-1
ØCTAL2  LGL     3
        SUB     =H00000
        TNZ     GINER
ØCTAL3  LDQ     SUM
        TRA     1,4
*      D - BCD CØNVERSIØN
DGIN00  LDQ     0,1
        TSX     SGIN00,4
        CLA     =H     0
        STØ     CRDADD
        LDQ     1,1
        TXI     BGIN02,1,-1
*      G - TABLE ØRIGIN
GGIN00  CLA     ADDIND
        SUB     =H00000
        TZE     GGIN01
        PXD     ,0
        TRA     GGIN02
GGIN01  CAL     GIN00
GGIN02  SUB     1B35
        ADD     CRDADD
        STA     SGIN03
        TRA     GIN12
*      H -- HEADER CØNVERSIØN
HGIN00  CLA     CRDADD
        CAS     =H     0
        TRA     *+2
        TRA     GIN12                       BLANK
        LDQ     CRDADD
        MPY     =11
        LLS     35
        ACL     GIN01
        STA     HGIN01
        AXT     11,1
        CLA     CØMMØN+12,1
HGIN01  STØ     **,1
        TIX     HGIN01-1,1,1
        TRA     GIN12
*      I - INTEGER CØNVERSIØN
IGIN00  TSX     DECNØ,4
        TXH     GINER,2,1                   DECIMAL PLACES
        TXL     BGIN02,2,0                  STØRE INTEGER
        CLM
        DVP     10B35
        TNZ     GINER                       NØT .0
        TRA     BGIN02
*      F - FIXED PØINT CØNVERSIØN
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
FGINOO  TSX     DECEN,4
        STQ     CØMMØN                  EXPØNENT
        LDC     DECN2,2
        TXI     *+1,2,1
        CLA     CØMMØN
        SUB     BREF,2
        TMI     GINER
        STA     FGINO1
        LDQ     CØMMØN+13
        MPY     FREF,2
FGINO1  LRS     **
        TNZ     GINER
        TXL     BGINO2,,**
*        - FLØATING PØINT CØNVERSIØN
LGINOO  TSX     DECEN,4
        STQ     CØMMØN                  EXPØNENT
        LXD     DECN2,2
        PXA     ,2
        SSM
        ADD     CØMMØN                  ACTUAL EXPØNENT FØR B ØF 35
        LRS     35
        DVP     10B35
        SUB     M9B35
        PAX     ,2                      FRACTIØNAL PART -- MØD 10
        CLM
        LLS     35
        ADD     =5                      HIGH DIGIT
        TMI     GINER
        PAX     ,4                      INTEGRAL PART -- MØD 10
        SXA     LGINO3,1
        CLA     XREF1,2
        ADD     XREF2,4
        ADD     =126
        TPL     LGINOA
        STZ     CØMMØN+13               UNDERFLØW
LGINOA  PAX     0,1
        LDQ     FREF1,2
        MPR     FREF2,4
        LRS     35
        MPY     CØMMØN+13
        LLS     2
        TZE     LGINO3
LGINO1  TZE     LGINO2
        LRS     1
        TXI     LGINO1,1,1              RØUND CAUSED 1 BIT ØVERFLØW
LGINO2  LLS     27
        RND
        LRS     27
        TZE     *+3
        LRS     1
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        TXI       *+1,1,1              ADD 1 TØ BINARY CHARACTERISTIC
        PXA       ,1
        LLS       0                    SHIFT IN CHARACTERISTIC
        LRS       8
BGINO3  AXT       **,1                 CHARACTERISTIC TØØ LARGE
        TNZ       GINER                EXIT TØ STØRE
        TRA       BGINO2
*
        DEC       133                  10**40    - BINARY EXPØNENT
        DEC       100                  10**30
        DEC       67                   10**20
        DEC       34                   10**10
1B35    DEC       1                    10**0
        DEC       -33                  10**-10
        DEC       -66                  10**-20
        DEC       -99                  10**-30
        DEC       -132                 10**-40
XREF2   DEC       .9183549616B0        2**-133/10**40
        DEC       .7888609053B0        2**-100/10**30
        DEC       .6776263579B0        2**-67/10**20
        DEC       .5820766092B0        2**-34/10**10
        DEC       .5B0                 2**-1/10**0
        DEC       .8589934592B0        2**33/10**-10
        DEC       .7378697630B0        2**66/10**-20
        DEC       .6338253002B0        2**9/10**-30
        DEC       .5444517871B0        2**132/10**-40
FREF2   DEC       30                   10**9    - BINARY EXPØNENT
        DEC       27                   10**8
        DEC       24                   10**7
        DEC       20                   10**6
        DEC       17                   10**5
        DEC       14                   10**4
10B35   DEC       10                   10**3
        DEC       7                    10**2
4B35    DEC       4                    10**1
BREF    DEC       1                    10**0
        DEC       -3                   10**-1
M6B35   DEC       -6                   10**2
M9B35   DEC       -9                   10**-3
        DEC       -13                  10**-4
        DEC       -16                  10**-5
        DEC       -19                  10**-6
        DEC       -23                  10**-7
        DEC       -26                  10**-8
XREF1   DEC       -29                  10**-9
        DEC       .9313225747B0        2**-30/10**-9
        DEC       .7450580597B0        2**-27/10**-8
        DEC       .5960464478B0        2**-24/10**-7
        DEC       .9536743165B0        2**-20/10**-6
        DEC       .7629394531B0        2**-17/10**-5
```

FIGURE **46**. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        DC      .6103515625B0       2**-14/10**-4
        DEC     .9765625B0          2**-10/10**-3
        DEC     .78125B0            2**-7/10**-2
        DEC     .625B0              2**-4/10**-1
FREF    DEC     .5B0                2**-1/10**0
        DEC     .8B0                2**3/10**1
        DEC     .64B0               2**6/10**2
        DEC     .512B0              2**9/10**3
        DEC     .8192B0             2**13/10**4
        DEC     .65536B0            2**16/10**5
        DEC     .524288B0           2**19/10**6
        DEC     .8388608B0          2**23/10**7
        DEC     .67108864B0         2**26/10**8
FREF1   DEC     .536870912B0        2**29/10**9
*       DECEN CØNVERSIØN
DECEN   SXD     GINO3,4
        TSX     DECNØ,4             CØNVERT VALUE FIELD
        STQ     CØMMØN+13
        SXD     DECN2,2
        LDQ     CØMMØN+20
        STQ     0,1
        STZ     SUM
        STZ     SIGN
        CLA     DECN2               2
        TSX     DECIM,4             CØNVERT EXPØNENT FIELD
        TXI     DECER,1,1           ERRØR
        NØP
        LXD     GINO3,4
        TRA     1,4
*       RESET FØR A-PREFIX ØPTIØN CØNVERSIØN
DECER   CLA     CØMMØN+33
        STØ     0,1
        CLA     CØMMØN+34
        STØ     1,1
        TRA     ACDO
*       DECNØ CØNVERSIØN ØF 10 CHARACTER VALUE FIELD USING DECIM
DECNØ   SXA     DECN4,4
        CAL     0,1
        SLW     CØMMØN+33           SAVE FØR ERRØR RETURN
        CAL     1,1
        SLW     CØMMØN+34           SAVE FØR ERRØR RETURN
        LDQ     =H
        LGR     12
        STQ     CØMMØN+20           SHIFT ØFF EXPØNENT AND SAVE
        LDQ     =H
        LGL     12
        SLW     1,1
        LAS     =H
        TRA     *+6
        TRA     *+2                 BLANKS
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        TRA     *+4
        LAS     0,1
        TRA     *+2
        TRA     DECN3              ALL BLANKS
*      LEFT JUSTIFY FIELD
        AXT     10,2
        LDQ     0,1
        LGL     6
        SUB     =Ø60
        TNZ     DECN5
        LGL     30
        LDQ     1,1
        LGL     6
        SLW     0,1
        LGL     30
        LDQ     =H
        LGL     6
        SLW     1,1
        TIX     *-12,2,1
*      EDIT ØUT EXTRA ZERØS AFTER DECIMAL PØINT - IF ANY
DECN5   AXT     6,2
        AXT     0,4
        LDQ     0,1
        CLM
        LGL     6
        SUB     =Ø60
        TZE     DECN6              STØP ØN BLANK
        ADD     =Ø25
        TNZ     *+2
        TXI     DECN6,4,1          DECIMAL PØINT FØUND
        TIX     *-7,2,1
        AXT     4,2
        LDQ     1,1
        CLM
        LGL     6
        SUB     =Ø60
        TZE     DECN6              STØP ØN BLANK
        ADD     =Ø25
        TNZ     *+2
        TXI     DECN6,4,1          DECIMAL PØINT FØUND
        TIX     *-7,2,1
DECN6   TXL     DECN3,4,0          NØ DECIMAL PØINT -- NØ EDIT
        AXT     4,2                EDIT FRØM RIGHT
        CAL     1,1
        LGR     12                 SKIP EXPØNENT -- BLANKS
        AXT     0,4
        LGR     6
        SLW     1,1
        CLM
        LGL     6
        TZE     *+4
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        SUB    =O60
        TZE    *+2
        TXI    *+1,4,1          SET FLAG TO STOP  NON-ZERO, NON-BLANK
        ADD    =O60             RESET OR EDIT
        LGR    6
        CAL    1,1
        TXH    *+3,4,0
        TIX    *-12,2,1
        TRA    *+4
        TNX    *+3,2,1
        LGR    6
        TIX    *-1,2,1
        STQ    1,1
        TXH    DECN3,4,0
        AXT    6,2
        CAL    0,1
        LGR    6
        SLW    0,1
        CLM
        LGL    6
        TZE    *+4
        SUB    =O60
        TZE    *+2
        TXI    *+1,4,1          SET FLAG TO STOP  NON-ZERO, NON-BLANK
        ADD    =O60
        LGR    6
        CAL    0,1
        TXH    *+3,4,0
        TIX    *-12,2,1
        TRA    GINER
        TNX    *+3,2,1
        LGR    6
        TIX    *-1,2,1
        STQ    0,1
DECN3   STZ    SUM
        STZ    SIGN
        AXT    100,2
        CLS    M6B35            6
        TSX    DECIM,4          CONVERT FIRST 6 CHARACTERS
        TRA    DECER            RROR
        TXI    DECN2+1,1,-1     END ON BLANK
        TXI    DECN1,1,-1
DECN1   CLA    4B35
        TSX    DECIM,4          CONVERT LAST 4 CHARACTERS
        TXI    DECER,1,1        ERROR
DECN2   TXH    2,,**
        TXL    DECN4,2,99
        TXL    GIN11,2,100
        AXT    0,2
DECN4   AXT    **,4
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        TRA     1,4
*      DECIM CØNVERSIØN
*      DECIM CØNVERSIØN ØF BCD TØ INTEGER
*               RETURNS WITH CØNVERTED SUM IN MQ -- SUM NØT CLEARED
*               XRZ IS TALLY ØF NUMBER ØF SHIFTS
*                       XR2 SET TØ ZERØ FØR DECIMAL PØINT
*               NUMBR TØ BE CØNVERTED IN 0,1
*       AXT     100,2
*       CLA     N                       NØ ØF CHARACTERS TØ CØNVERT
*       TSX     DECIM,4
*       ERRØR RETURN
*       BLANK CHARACTER RETURN
*       NØRMAL RETURN
*
DECIM   SXA     DECIM8,4
        PAX     ,4
DECIM1  CLM
        LDQ     0,1
        LGL     6                       SHIFT ØFF 1 BCD CHARACTER
        STQ     0,1
        SUB     10B35
        TMI     DECIM6                  0,1,2,3,4,5,6,7,8,9
        SBM     M6B35
        TZE     DECIM7                  + IGNØRE PLUS SIGN
        SUB     =11
        TNZ     DECIM3                  NØT A DECIMAL PØINT .
        AXT     0,2
        TRA     DECIM7
DECIM3  SUB     =5
        TMI     DECIM9                  =,@,A,B,C,D,E,F,G,H,I,),
        SUB     10B35
        TNZ     DECIM4                  NØT A -0
        SUB     10B35                   FØRCE SIGN NEGATIVE
DECIM4  TPL     DECIM5                  NØT AN ØVERPUNCH  J,K,L,M,N,Ø,P,Q,R
        STØ     SIGN
        TRA     DECIM6
DECIM5  ADD     M6B35                   $,*
        TZE     DECIM8                  BLANK
        SUB     =11
        TNZ     DECIM9                  /,S,T,U,V,W,X,Y,Z,(
        TRA     DECIM7                  , IGNØRE CØMMA
DECIM8  AXT     **,4
        LDQ     SUM
        CLA     SIGN
        LRS     0
        TRA     2,4                     BLANK CHARACTER RETURN
DECIM6  ADD     10B35
        STØ     CØMMØN+14
        LDQ     SUM
        MPY     10B35
```

FIGURE **46**. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
       XCA
       ADD     CØMMØN+14
       STØ     SUM
       TXI     *+1,2,1
DECIM7 TIX     DECIM1,4,1
       LXA     DECIM8,4                END ØN DIGIT CØUNT
       TXI     DECIM8+1,4,-1
DECIM9 LXA     DECIM8,4                ERRØR RETURN
       TRA     1,4
*
*     DIMENSIØN TABLE
KGIN00 NZT     FINKY
       TRA     GINER
       TSX     DECNØ,4
       TXH     GINER,2,0
       XCA
       TMI     GINER
       ADM     SGIN03
       STA     SGIN03
       TRA     GIN11
*     SAVE LØCATIØN FØR BUCKET
CGIN00 NZT     FINKY
       TRA     GINER
       CLA     0,1
       TSX     TBLØØ,4
       CAL     SGIN03
       ADD     1B35
CGIN01 AXT     **,2
       STA*    TBLØ2
       TXI     GIN11,1,-1
*     TRANSFER
JGIN00 CLA     CRDADD
       SUB     =H     0
       TZE     JGIN01
       CLA     ADDIND
       SUB     =H00000
       TZE     JAA1
       CLA     CRDADD
       TRA     JGIN02
JAA1   CLA     CRDADD
       TZE     JGINOA
       CAL     CRDADD
       SUB     1B35
       STA     CRDADD
JGINOA CAL     GIN00
       ACL     CRDADD
       TRA     JGIN02
JGIN01 CAL     SGIN03
       ACL     1B35
JGIN02 STA     SGIN03
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        TRA*    SGIN03
*      STØRE RØUTINE
SGIN00 CLA      CRDADD
        SUB     =H     0
        TZE     SGIN01
        CLA     ADDIND
        SUB     =H00000
        TZE     SAA1
        CLA     CRDADD
        TRA     SGIN02
SAA1    CLA     CRDADD                  ØNLY IF CARDADD HAS
        TZE     SGIN0A
        CAL     CRDADD
        SUB     1B35                    RELATIVE LØCATIØN IN
        STA     CRDADD
SGIN0A  CAL     GIN00
        ACL     CRDADD
        TRA     SGIN02
SGIN01  CAL     SGIN03                  ADD 1 TØ ALD ADDRESS
        ACL     1B35
SGIN02  STA     SGIN03
        NZT     ERRØR
SGIN03  STQ     **
        TRA     1,4
*      MATRIX SET-UP
MTX00   SXA     MTX10,4                 TSX MTX00,4
        STØ     CØMMØN+35               NØ CHAR I
        SSM
        ADD     =5
        STØ     CØMMØN+36               NØ CHAR J
        AXT     2,4                     IJ RETURN
MTX01   SXA     MTX02,4                 NØRMAL RETURN
        STZ     SIGN
        STZ     SUM
        AXT     100,2
        CLA     CØMMØN+37,4
        TSX     DECIM,4
        TRA     GINER
        TRA     MTX11                   LESS THAN N CHARS
        TXL     GINER,2,100
MTX02   AXT     **,4
        STQ     I+2,4                   I,J
        TIX     MTX01,4,1
MTX10   AXT     **,4
        CLA     J
        SUB     1B35
        XCA
        MPY     IMAX
        XCA
        ADD     I
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        LDQ     IJ
        TLQ     1,4                 LØC GTR MAX
        XCA
        TRA     2,4
MTX11   LXA     MTX02,4
        TXH     GINER,4,1           I ERRØR
        TRA     MTX02-1
*     L,M - MATRIX INPUT
*       SET UP IJ WØRD
MCD00   SXA     MCD02,4
        CAL     GIN00
        SUB     1B35
        STA     SGIN03
        LDQ     0,1                 DETERMINE CØMMA PØSITIØN
        LGL     6
        AXT     2,4
        PXD     ,0
        LGL     6
        SUB     =H00000,
        TZE     *+3
        TXH     GINER,4,3
        TXI     *-5,4,1
        PXA     ,4
        TSX     MTX00,4
MCD02   NØP     **
        LDQ     I
        STQ     IMAX
        MPY     J
        STQ     IJ
        LXA     MCD02,4
        TXL     MCD04,4,12
        LXA     IJ,4
        CAL     GIN00
        ADM     IJ
        STA     *+1         ZERØ MATRIX
        STZ     **,4
        TIX     *-1,4,1
MCD04   TXI     GIN11,1,-1
*     A - SYMBØLIC CØNVERSIØN
ACD0    STZ     CTR
        STZ     PASS
ACD1    STZ     PF
        AXT     5,2
        TSX     SYM00,4
        AXT     0,4                 BLANK
        TXH     ACD6,2,1
        TXL     ACD12,4,0           NULL AND BLANK
        TXL     ACD6+1,,0           NULL
ACD6    STQ     PF
        TXL     ACD10,4,0           BLANK
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        AXT     1,2
        TSX     SYM00,4
        AXT     0,4                 BLANK
        TXL     ACD10,2,1           NULL
        XCL
        ALS     15
        STT     PF
ACD10   LDQ     PF
        CLS     PASS
        STØ     PASS
        TPL     ACD15
        STQ     WØRD
        TXH     ACD1,4,0            NØT BLANK
ACD12   LXA     CTR,4
        TXH     ACD14,4,6
        TXI     *+1,1,-1
        TXL     *+3,4,1             BLANK
ACD14   LDQ     WØRD
        TSX     SGIN00,4
        TXL     GIN11,,0
ACD15   RQL     18
        SLQ     WØRD
        TXL     ACD12,,0
*
SYM00   SXD     SYM02,2             LXA L(NØ CHARACTERS),2
        SXA     SYM14,4             TSX SYM00,4
        STZ     PSYM                BALNK
        AXT     1,2                 NØRMAL IR2=1 IF NULL
SYM01   CLA     CTR
        ADD     1B35
        STØ     CTR
        SUB     10B35+1             7
        TNZ     *+2
        TXI     *+1,1,-1
        LDQ     0,1
        CAL     PSYM
        LGL     6
        SLW     PSYM
        STQ     0,1
        ANA     =Ø77
        SUB     =H00000
        TZE     SYM20               BLANK
        SUB     =11
        TZE     SYM21               CØMMA
        TXI     *+1,2,1
SYM02   TXL     SYM01,2,**
        CAL     PSYM
SYM03   TXI     *+1,2,-1
        SXA     SYM06,2             NØ CHARACTERS
        LDQ     =H     0
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
SYM04   LGL     6                       LEFT JUSTIFY
        TXI     *+1,2,1
        TXL     SYM04,2,5
        SLW     SYMBL
        CAL     PSYM
        ANA     =H
        TNZ     SYM10                   SYMBØLIC
SYM06   AXT     **,2                    NØ CHARACTERS
        LDQ     SYMBL
        STZ     SUM
        STZ     SIGN
        TXL     SYM08,2,4               DECIMAL
        TSX     ØCTAL1,4                ØCTAL
        TXL     GINER,,0
        TSX     SYM14,2
SYM08   CLA     0,1
        STQ     0,1
        STØ     SYMBL
        PXA     ,2
        TSX     DECIM,4
        TXL     GINER,,0
        TRA     GINER
        CLA     SYMBL
        STØ     0,1
        TSX     SYM14,2
SYM10   CLA     SYMBL
        TSX     TBL,4
SYM14   AXT     **,4
        TRA     2,4                     EXIT
SYM20   TXI     *+1,4,1                 BLANK
        SXA     SYM14,4
SYM21   TXL     SYM14+1,2,1             NULL FIELD
        CAL     PSYM
        ARS     6
        SLW     PSYM
        TRA     SYM03
*       TABLE LØØK-UP
TBL     NZT     FINKY
        TRA     TBL00
        TRA     TBLØØ
*
TBLØØ   LXA     **,2
TBLØ1   CAS     **,2
        TRA     *+2
        TXI     TBLØ2,2,-1
        TIX     TBLØ1,2,2
        TRA     GINER
TBLØ2   LDQ     **,2
        SXA     CGIN01,2
        LXD     TBLØ1+2,2
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

```
        TRA     1,4
*
TBL00   LXA     **,2                    N
TBL01   CAS     **,2
        TXL     *+2,,0
        TXI     TBL02,2,-2
        TIX     TBL01,2,1
TBL03   TXL     GINER,0,-1
TBL02   LDQ     **,2                    ALPHA + N - (N-2)
        IXD     TBL03,2
        TRA     1,4
*      MISCELLANY
IMAX    PZE
IJ      PZE
CØMMØN  BSS     35
GINOO   SYN     CØMMØN+30
PF      SYN     CØMMØN+15
PASS    SYN     CØMMØN+16
CTR     EQU     CØMMØN+17
SYMBL   EQU     CØMMØN+18
PSYM    EQU     CØMMØN+19
CRDADD  EQU     CØMMØN+21
SIGN    EQU     CØMMØN+22
SUM     EQU     CØMMØN+23
ADDIND  EQU     CØMMØN+24
PREFIX  EQU     CØMMØN+25
WØRD    SYN     CØMMØN+26
ERRØR   SYN     CØMMØN+27
I       SYN     PASS
J       SYN     CTR
FINPT   PZE
FINKY   PZE
LINPUT  PZE
LØUTPT  PZE
IUNIT   PZE
ØUNIT   PZE
ITAPE   PZE
ØTAPE   PZE
CØUNT   PZE
FØRMAT  BCI     2,(15A6,I6)
        END
```

FIGURE 46. SYMBOLIC LISTING OF SUBROUTINE FINPUT (cont.)

XXXIII. SUBROUTINE FXINP

This subroutine converts a list or array of floating point numbers to fixed point numbers. If any number in the list or array is already in fixed point form, the conversion of that number is skipped. There are 2 entry points.

CALLING SEQUENCES

CALL FXINP (N, ITEM1, ITEM2, ITEM3, ... ITEMN)

N = number of items in list

ITEM(i) = location(s) of numbers to be fixed

or:

CALL FXINQ (N, ARRAY)

N = dimension of ARRAY

USAGE

FXINP is used in order that all numerical input data may be input as floating point numbers (no I-prefix necessary). Those items which are desired to be fixed point integers for use within the program are fixed using this routine.

```
$IBMAP FXINPT
*
*      SUBRØUTINE TØ PRØCESS A LIST ØF DATA, CØNVERTING FLØATING PAINT
*               NUMBERS TØ FIXED PØINT NUMBERS WITHØUT ALTERING ANY
*               NUMBERS WHICH ARE ALREADY FIXED
*
*       CALL    FXINP(N,ITEM1,ITEM2,...,ITEMN)
*
*       CALL    FXINQ(N,ARRAY)
*               DIMENSIØN ØF ARRAY IS N
*
FXINP   SAVE    (4,2,1)I
        CLA*    3,4
        PAX     ,1
        TXI     *+1,4,-1
        CLA*    3,4
        TSX     FIX00,2
        STØ*    3,4
        TIX     *-4,1,1
        RETURN  FXINP
*
FXINQ   SAVE    (4,2,1)I
        CAL*    3,4
        PAX     ,1
        ADD     4,4
        STA     *+1
        CLA     **,1
        TSX     FIX00,2
        STØ*    *-2
        TIX     *-3,1,1
        RETURN  FXINQ
*
FIX00   STØ     FIX02
        TZE     1,2                 ZERØ
        ANA     =Ø377000000000
        TNZ     FIX01
        CLA     FIX02
        TRA     1,2                 ALREADY FIXED
FIX01   CLA     FIX02
        UFA     =Ø233000000000
        ANA     =Ø777777777
        LDQ     FIX02
        LLS     0
        TRA     1,2
FIX02   PZE
*
        END
```

FIGURE 47. SYMBOLIC LISTING OF SUBROUTINE FXINP

XXXIV. SUBROUTINE SYMTAB

This routine is not an executable routine. It is the symbol table used by FINPUT for symbolic storage locations and the "A"-prefix conversion code option. It is also used by LOOKUP to look up names of locations.

SYMTAB is the deck name; BCIPZE is the referenced "entry-point." BCIPZE can be referenced either by an EXTERNAL statement, as is done in GUIDON, or by a named COMMON block, (i.e., COMMON/BCIPZE/TABLE(1)) as is done in LOOKUP.

The format of the table is as follows:

```
TABLE   XXX   ENDT-TABLE
        BCI   1, (5-character Hollerith name)
        PZE   (location)
         .
         .
         .
        BCI   1,
ENDT    PZE
```

if

XXX = MZE input cards are listed

XXX = PZE input cards are not listed

ENDT must be the last entry in the table in order for ENDT-TABLE to be compiled correctly as the length of the table.

The locations in the PZE word reference the named COMMON block and relative location corresponding to the parameter named in the preceding BCI word. Alternate names for the same location are permissible.

For the dump routine, each named COMMON block has its own entry in the table where the PZE word specifies not only the location, but the length.

```
$IBMAP SYMTAB
BCIPZE CØNTRL  BCIPZE
       USE     BCIPZE
       EVEN
TABLE  MZE     ENDT-TABLE        NØ. ØF ENTRIES IN TABLE
*     DATA IN /TAPES/ CØMMØN BLØCK
       BCI     1,TAPES           CØMMØN BLØCK /TAPES/
       PZE     TAPES,,3          LØCATIØN,,LENGTH
       BCI     1,ITAPE           INPUT TAPE
       PZE     TAPES+0
       BCI     1,ØTAPE           ØUTPUT TAPE
       PZE     TAPES+1
       BCI     1,STAPE           SCRATCH TAPE
       PZE     TAPES+2
*     DATA IN /CØNST/ CØMMØN BLØCK
       BCI     1,CØNST           CØMMØN BLØCK /CØNST/
       PZE     CØNST,,21         LØCATIØN,,LENGTH
       BCI     1,RSUBO           EQUATØRIAL RADIUS - FT
       PZE     CØNST+0
       BCI     1,RSUBØ
       PZE     CØNST+0
       BCI     1,FLAT            FLATTENING
       PZE     CØNST+1
       BCI     1,GSUBO           SURFACE GRAVITY - FT/SEC**2
       PZE     CØNST+2
       BCI     1,GSUBØ
       PZE     CØNST+2
       BCI     1,GSUB1           WEIGHT TØ MASS CØNVERSIØN
       PZE     CØNST+3
       BCI     1,ØMEGA           SPIN RATE - RAD/SEC
       PZE     CØNST+4
       BCI     1,HMIN            MINIMUM ALLØWABLE ALTITUDE - FT
       PZE     CØNST+7
       BCI     1,HMAX            MAXIMUM ALTITUDE ØF ATMØSPHERE - FT
       PZE     CØNST+8
       BCI     1,TAGS            CØNVERSIØN FACTØRS
       PZE     CØNST+9
*     DATA IN /INPUT/ CØMMØN BLØCK
       BCI     1,INPUT           CØMMØN BLØCK /INPUT/
       PZE     INPUT,,555
       BCI     1,TI              INITIAL TIME - SEC
       PZE     INPUT+0
       BCI     1,VI              INITIAL VELØCITY (RELATIVE) - FT/SEC
       PZE     INPUT+1
       BCI     1,HI              INITIAL HEIGHT - FT
       PZE     INPUT+2
       BCI     1,LATI            INITIAL LATITUDE (GEØDETIC) - DEGREES
       PZE     INPUT+3
       BCI     1,LØNGI           INITIAL LØNGITUDE - DEGREES
       PZE     INPUT+4
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB

```
BCI     1,BETAI        INITIAL BETA - DEGREES
PZE     INPUT+5
BCI     1,AZI          INITIAL (FIRING) AZIMUTH - DEGREES
PZE     INPUT+6
BCI     1,BETAO        INITIAL KICK-ØVER - DEGREES
PZE     INPUT+7
BCI     1,BETAØ
PZE     INPUT+7
BCI     1,WI           INITIAL MISSILE WEIGHT
PZE     INPUT+8
BCI     1,WPL          ADDITIØNAL PAYLØAD WEIGHT
PZE     INPUT+9
BCI     1,SIDTI        INITIAL LØCAL SIDEREAL TIME
PZE     INPUT+10
BCI     1,DATE         LAUNCH DATE VECTØR
PZE     INPUT+11
BCI     1,YEAR         YEAR ØF LAUNCH
PZE     INPUT+11
BCI     1,MØNTH        MØNTH ØF LAUNCH
PZE     INPUT+12
BCI     1,DAY          DAY ØF LAUNCH
PZE     INPUT+13
BCI     1,HØUR         HØUR ØF LAUNCH
PZE     INPUT+14
BCI     1,MIN          MINUTE ØF LAUNCH
PZE     INPUT+15
BCI     1,SEC          SECØND ØF LAUNCH
PZE     INPUT+16
BCI     1,ZØNE         TIME ZØNE ØF LAUNCH
PZE     INPUT+17
BCI     1,TARGT        TARGET
PZE     INPUT+18
BCI     1,TRANS        TRANSF
PZE     INPUT+19
BCI     1,FLTR         FLAG FØR TYPE ØF RUN
PZE     INPUT+20
BCI     1,FLSC         NØ. ØF SECTIØNS IN FLIGHT
PZE     INPUT+21
BCI     1,FLPRT        PRINT ØPTIØN FLAG
PZE     INPUT+22
BCI     1,SEXO1        SECTIØN 1 DATA
PZE     INPUT+30
BCI     1,SEXOI
PZE     INPUT+30
BCI     1,SEXØ1
PZE     INPUT+30
BCI     1,SEXØI
PZE     INPUT+30
BCI     1,SEXO2        SECTIØN 2 DATA
PZE     INPUT+50
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
BCI     1,SEXØ2
PZE     INPUT+50
BCI     1,SEX03              SECTIØN 3 DATA
PZE     INPUT+70
BCI     1,SEXØ3
PZE     INPUT+70
BCI     1,SEX04              SECTIØN 4 DATA
PZE     INPUT+90
BCI     1,SEXØ4
PZE     INPUT+90
BCI     1,SEX05              SECTIØN 5 DATA
PZE     INPUT+110
BCI     1,SEXOS
PZE     INPUT+110
BCI     1,SEXØ5
PZE     INPUT+110
BCI     1,SEXØS
PZE     INPUT+110
BCI     1,SEX06              SECTIØN 6 DATA
PZE     INPUT+130
BCI     1,SEXØ6
PZE     INPUT+130
BCI     1,SEX07              SECTIØN 7 DATA
PZE     INPUT+150
BCI     1,SEXØ7
PZE     INPUT+150
BCI     1,SEX08              SECTIØN 8 DATA
PZE     INPUT+170
BCI     1,SEXØ8
PZE     INPUT+170
BCI     1,SEX09              SECTIØN 9 DATA
PZE     INPUT+190
BCI     1,SEXØ9
PZE     INPUT+190
BCI     1,SEX10              SECTIØN 10 DATA
PZE     INPUT+210
BCI     1,SEX1Ø
PZE     INPUT+210
BCI     1,SEXI0
PZE     INPUT+210
BCI     1,SEXIØ
PZE     INPUT+210
BCI     1,SEX11              SECTIØN 11 DATA
PZE     INPUT+230
BCI     1,SEX1I
PZE     INPUT+230
BCI     1,SECI1
PZE     INPUT+230
BCI     1,SEXII
PZE     INPUT+230
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
        BCI     1,SEX12                 SECTION 12 DATA
        PZE     INPUT+250
        BCI     1,SEXI2
        PZE     INPUT+250
        BCI     1,STAG1                 STAGE 1 DATA
        PZE     INPUT+270
        BCI     1,STAGI
        PZE     INPUT+270
        BCI     1,STAG2                 STAGE 2 DATA
        PZE     INPUT+290
        BCI     1,STAG3                 STAGE 3 DATA
        PZE     INPUT+310
        BCI     1,STAG4                 STAGE 4 DATA
        PZE     INPUT+330
        BCI     1,STAG5                 STAGE 5 DATA
        PZE     INPUT+350
        BCI     1,STAGS
        PZE     INPUT+350
        BCI     1,STAG6                 STAGE 6 DATA
        PZE     INPUT+370
*      DATA IN /TABLES/ COMMON BLOCK
        BCI     1,TABLE                 COMMON BLOCK /TABLES/
        PZE     TABLES,,2100            LOCATION,,LENGTH
        BCI     1,MACH1                 MACH NO. - MACH VS CSUBA AND CSUBN
        PZE     TABLES+0
        BCI     1,MACH2
        PZE     TABLES+30
        BCI     1,MACH3
        PZE     TABLES+60
        BCI     1,MACH4
        PZE     TABLES+90
        BCI     1,MACH5
        PZE     TABLES+120
        BCI     1,MACH6
        PZE     TABLES+150
        BCI     1,CSBN1                 CSUBN - MACH VS CSUBN
        PZE     TABLES+180
        BCI     1,CSBN2
        PZE     TABLES+210
        BCI     1,CSBN3
        PZE     TABLES+240
        BCI     1,CSBN4
        PZE     TABLES+270
        BCI     1,CSBN5
        PZE     TABLES+300
        BCI     1,CSBN6
        PZE     TABLES+330
        BCI     1,CSBA1                 CSUBA - MACH VS CSUBA
        PZE     TABLES+360
        BCI     1,CSBA2
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
PZE     TABLES+390
BCI     1,CSBA3
PZE     TABLES+420
BCI     1,CSBA4
PZE     TABLES+450
BCI     1,CSBA5
PZE     TABLES+480
BCI     1,CSBA6
PZE     TABLES+510
BCI     1,TPIT1                 TIME - TIME VS PITCH ANGLE
PZE     TABLES+540
BCI     1,TPIT2
PZE     TABLES+570
BCI     1,TPIT3
PZE     TABLES+600
BCI     1,TPIT4
PZE     TABLES+630
BCI     1,TPIT5
PZE     TABLES+660
BCI     1,TPIT6
PZE     TABLES+690
BCI     1,PIT1                  PITCH - TIME VS PITCH ANGLE
PZE     TABLES+720
BCI     1,PIT2
PZE     TABLES+750
BCI     1,PIT3
PZE     TABLES+780
BCI     1,PIT4
PZE     TABLES+810
BCI     1,PIT5
PZE     TABLES+840
BCI     1,PIT6
PZE     TABLES+870
BCI     1,TYAW1                 TIME - TIME VS YAW ANGLE
PZE     TABLES+900
BCI     1,TYAW2
PZE     TABLES+930
BCI     1,TYAW3
PZE     TABLES+960
BCI     1,TYAW4
PZE     TABLES+990
BCI     1,TYAW5
PZE     TABLES+1020
BCI     1,TYAW6
PZE     TABLES+1050
BCI     1,YAW1                  YAW - TIME VS YAW ANGLE
PZE     TABLES+1080
BCI     1,YAW2
PZE     TABLES+1110
BCI     1,YAW3
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
PZE      TABLES+1140
BCI      1,YAW4
PZE      TABLES+1170
BCI      1,YAW5
PZE      TABLES+1200
BCI      1,YAW6
PZE      TABLES+1230
BCI      1,TTRS1              TIM - TIME VS THRUST
PZE      TABLES+1260
BCI      1,TTRS2
PZE      TABLES+1290
BCI      1,TTRS3
PZE      TABLES+1320
BCI      1,TTRS4
PZE      TABLES+1350
BCI      1,TTRS5
PZE      TABLES+1380
BCI      1,TTRS6
PZE      TABLES+1410
BCI      1,TRUS1              THRUST - TIME VS THRUST
PZE      TABLES+1440
BCI      1,TRUS2
PZE      TABLES+1470
BCI      1,TRUS3
PZE      TABLES+1500
BCI      1,TRUS4
PZE      TABLES+1530
BCI      1,TRUS5
PZE      TABLES+1560
BCI      1,TRUS6
PZE      TABLES+1590
BCI      1,TWDT1              TIME - TIME VS WEIGHT FLØW
PZE      TABLES+1620
BCI      1,TWDT2
PZE      TABLES+1650
BCI      1,TWDT3
PZE      TABLES+1680
BCI      1,TWDT4
PZE      TABLES+1710
BCI      1,TWDT5
PZE      TABLES+1740
BCI      1,TWDT6
PZE      TABLES+1770
BCI      1,WDØT1              WDØT - TIME VS WEIGHT FLØW
PZE      TABLES+1800
BCI      1,WDØT2
PZE      TABLES+1830
BCI      1,WDØT3
PZE      TABLES+1860
BCI      1,WDØT4
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
         PZE      TABLES+1890
         BCI      1,WDØT5
         PZE      TABLES+1920
         BCI      1,WDØT6
         PZE      TABLES+1950
         BCI      1,JCSBN
         PZE      TABLES+1980              INTERPØLATIØN ØPTIØN
         BCI      1,JCSBA
         PZE      TABLES+1986
         BCI      1,JPIT
         PZE      TABLES+1992
         BCI      1,JYAW
         PZE      TABLES+1998
         BCI      1,JTRUS
         PZE      TABLES+2004
         BCI      1,JWDØT
         PZE      TABLES+2010
         BCI      1,LPIT                   ARGUMENT ØPTIØN
         PZE      TABLES+2016
         BCI      1,LYAW
         PZE      TABLES+2022
         BCI      1,LTRUS
         PZE      TABLES+2028
         BCI      1,LWDØT
         PZE      TABLES+2034
*        DATA IN /ENGINI/ CØMMØN BLØCK
         BCI      1,ENGNI                  CØMMØN BLØCK /ENGINI/
         PZE      ENGINI,,708              LØCATIØN,,LENGTH
         BCI      1,YBBR1                  BAR-BAR VALUES ØF DEP ENGINE PARAMETERS
         PZE      ENGINI+0
         BCI      1,YBBR2
         PZE      ENGINI+5
         BCI      1,YBBR3
         PZE      ENGINI+10
         BCI      1,YBBR4
         PZE      ENGINI+15
         BCI      1,YBBR5
         PZE      ENGINI+20
         BCI      1,YBBR6
         PZE      ENGINI+25
         BCI      1,YBAR1                  BAR VALUES ØF DEP ENGINE PARAMETERS
         PZE      ENGINI+30
         BCI      1,YBAR2
         PZE      ENGINI+35
         BCI      1,YBAR3
         PZE      ENGINI+40
         BCI      1,YBAR4
         PZE      ENGINI+45
         BCI      1,YBAR5
         PZE      ENGINI+50
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
BCI     1,YBAR6
PZE     ENGINI+55
BCI     1,XNØM1          NØMINAL VALUES ØF INDEP ENGINE PARAMS
PZE     ENGINI+60
BCI     1,XNØM2
PZE     ENGINI+66
BCI     1,XNØM3
PZE     ENGINI+72
BCI     1,XNØM4
PZE     ENGINI+78
BCI     1,XNØM5
PZE     ENGINI+84
BCI     1,XNØM6
PZE     ENGINI+90
BCI     1,KPC1           PC INFLUENCE CØEFFICIENTS
PZE     ENGINI+96
BCI     1,KPC2
PZE     ENGINI+102
BCI     1,KPC3
PZE     ENGINI+108
BCI     1,KPC4
PZE     ENGINI+114
BCI     1,KPC5
PZE     ENGINI+120
BCI     1,KPC6
PZE     ENGINI+126
BCI     1,KWØC1          CHAMBER ØXIDIZER FLØW INFL CØEFFS
PZE     ENGINI+132
BCI     1,KWØC2
PZE     ENGINI+138
BCI     1,KWØC3
PZE     ENGINI+144
BCI     1,KWØC4
PZE     ENGINI+150
BCI     1,KWØC5
PZE     ENGINI+156
BCI     1,KWØC6
PZE     ENGINI+162
BCI     1,KWFC1          CHAMBER FUEL FLØW INFL CØEFFS
PZE     ENGINI+168
BCI     1,KWFC2
PZE     ENGINI+174
BCI     1,KWFC3
PZE     ENGINI+180
BCI     1,KWFC4
PZE     ENGINI+186
BCI     1,KWFC5
PZE     ENGINI+192
BCI     1,KWFC6
PZE     ENGINI+198
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
BCI     1,KW0G1            GAS GEN 0XIDIZER FL0W INFL C0EFFS
PZE     ENGINI+204
BCI     1,KW0G2
PZE     ENGINI+210
BCI     1,KW0G3
PZE     ENGINI+216
BCI     1,KW0G4
PZE     ENGINI+222
BCI     1,KW0G5
PZE     ENGINI+228
BCI     1,KW0G6
PZE     ENGINI+234
BCI     1,KWFG1            GAS GEN FUEL FL0W INFL C0EFFS
PZE     ENGINI+240
BCI     1,KWFG2
PZE     ENGINI+246
BCI     1,KWFG3
PZE     ENGINI+252
BCI     1,KWFG4
PZE     ENGINI+258
BCI     1,KWFG5
PZE     ENGINI+264
BCI     1,KWFG6
PZE     ENGINI+270
BCI     1,CT0X1            0XIDIZER TEMPERATURE P0LYN0MIAL C0EFFS
PZE     ENGINI+276
BCI     1,CT0X2
PZE     ENGINI+282
BCI     1,CT0X3
PZE     ENGINI+288
BCI     1,CT0X4
PZE     ENGINI+294
BCI     1,CT0X5
PZE     ENGINI+300
BCI     1,CT0X6
PZE     ENGINI+306
BCI     1,CTFL1            FUEL TEMPERATURE P0LYN0MIAL C0EFFS
PZE     ENGINI+312
BCI     1,CTFL2
PZE     ENGINI+318
BCI     1,CTFL3
PZE     ENGINI+324
BCI     1,CTFL4
PZE     ENGINI+330
BCI     1,CTFL5
PZE     ENGINI+336
BCI     1,CTFL6
PZE     ENGINI+342
BCI     1,MISC1            MISCELLANE0US ENGINE DATA
PZE     ENGINI+348
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
        BCI      1,MISC2
        PZE      ENGINI+378
        BCI      1,MISC3
        PZE      ENGINI+408
        BCI      1,MISC4
        PZE      ENGINI+438
        BCI      1,MISC5
        PZE      ENGINI+468
        BCI      1,MISC6
        PZE      ENGINI+498
        BCI      1,TNKS1                TANK DATA
        PZE      ENGINI+528
        BCI      1,TNKS2
        PZE      ENGINI+558
        BCI      1,TNKS3
        PZE      ENGINI+588
        BCI      1,TNKS4
        PZE      ENGINI+618
        BCI      1,TNKS5
        PZE      ENGINI+648
        BCI      1,TNKS6
        PZE      ENGINI+678
*      DATA IN /PRINT/ C0MM0N BL0CK
        BCI      1,PRINT                C0MM0N BL0CK /PRINT/
        PZE      IPRINT,,12             L0CATI0N,,LENGTH
        BCI      1,PRT0N                0PTI0NAL TIME HIST0RY EXTRA PRINT
        PZE      IPRINT+0
        BCI      1,PRT0F                0PTI0NA BURN-0UT EXTRA PRINT
        PZE      IPRINT+6
*      DATA IN /GAIND/ C0MM0N BL0CK
        BCI      1,GAIND                C0MM0N BL0CK /GAIND/
        PZE      GAIND,,21              L0CATI0N,,LENGTH
        BCI      1,FLPAR                FLAG T0 C0MPUTE GAIN PARTIALS
        PZE      GAIND+0
        BCI      1,IVAR                 INDEPENDNENT VARIABLES
        PZE      GAIND+1
        BCI      1,DVAR                 DEPENDENT VARIABLES
        PZE      GAIND+11
*      DATA IN /ENGIN0/ C0MM0N BL0CK
        BCI      1,ENGN0                C0MM0N BL0CK /ENGIN0/
        PZE      ENGIN0,,37             L0CATI0N,,LENGTH
        BCI      1,V0                   0XIDIZER V0LUME - FT**3
        PZE      ENGIN0+9
        BCI      1,VF                   FUEL V0LUME - FT**3
        PZE      ENGIN0+10
        BCI      1,H0                   0XIDIZER HEIGHT - FEET
        PZE      ENGIN0+11
        BCI      1,HF                   FUEL HEIGHT - FEET
        PZE      ENGIN0+12
        BCI      1,PC                   CHAMBER PRESSURE
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
        PZE     ENGINØ+16
        BCI     1,FENGN                 ENGINE THRUST - LBS
        PZE     ENGINØ+23
        BCI     1,ISPE                  ENGINE ISP - SEC
        PZE     ENGINØ+28
*      DATA IN /ENGINW/ CØMMØN BLØCK
        BCI     1,ENGNW                 CØMMØN BLØCK /ENGINW/
        PZE     ENGINW,,112             LØCATIØN,,LENGTH
        BCI     1,FØPT                  ØPTIMUM THRUST - LBS
        PZE     ENGINW+48
        BCI     1,ASUBT                 THRØAT AREA
        PZE     ENGINW+62
        BCI     1,EPSI                  EXPANSIØN RATIØ
        PZE     ENGINW+63
*      DATA IN /WFLAG/ CØMMØN BLØCK
        BCI     1,WFLAG                 CØMMØN BLØCK /WFLAG/
        PZE     WFLAG,,6                LØCATIØN,,LENGTH
*      DATA IN /WØRKD/ CØMMØN BLØCK
        BCI     1,WØRKD                 CØMMØN BLØCK /WØRKD/
        PZE     WØRKD,,40               LØCATIØN,,LENGTH
*      DATA IN /SAVE/ CØMMØN BLØCK
        BCI     1,SAVE                  CØMMØN BLØCK /SAVE/
        PZE     SAVE,,35                LØCATIØN,,LENGTH
        BCI     1,WØR                   ØX RESIDUAL
        PZE     SAVE+17
        BCI     1,WFR                   FUEL RESIDUAL
        PZE     SAVE+23
        BCI     1,AVMR                  AVERAGE INFLIGHT MIXTURE RATIØ
        PZE     SAVE+29
*      DATA IN /CØMP/ CØMMØN BLØCK
        BCI     1,CØMP                  CØMMØN BLØCK /CØMP/
        PZE     CØMP,,15                LØCATIØN,,LENGTH
*      DATA IN /GILL/ CØMMØN BLØCK
        BCI     1,GILL                  CØMMØN BLØCK /GILL/
        PZE     GILL,,66                LØCATIØN,,LENGTH
        BCI     1,TIME                  TIME
        PZE     GILL+0
        BCI     1,W                     MISSILE WEIGHT
        PZE     GILL+14
        BCI     1,WØ                    ØXIDIZER WEIGHT
        PZE     GILL+16
        BCI     1,WF                    FUEL WEIGHT
        PZE     GILL+18
        BCI     1,WPRP                  MISC. PRØPELLANT WEIGHT
        PZE     GILL+20
*
        BCI     1,VGAIN                 VELØCITY GAIN
        PZE     GILL+22
        BCI     1,TBURN                 STAGE BURNING TIME -- SEC
        PZE     GILL+24
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
        BCI     1,ACCEL           (F - D)/W -- THRUST TØ WEIGHT RATIØ
        PZE     GILL+36
*      DATA IN /DATA/ CØMMØN BLØCK
        BCI     1,DATA            CØMMØN BLØCK /DATA/
        PZE     DATA,,49          LØCATIØN,,LENGTH
        BCI     1,TFLIT           FLIGHT TIME - (TIME - TI) - SEC
        PZE     DATA+0
        BCI     1,TSTAG           STAGE TIME - (TIME - TSTG) - SEC
        PZE     DATA+1
        BCI     1,TSECT           SECTIØN TIME - (TIME - TSUBS) - SEC
        PZE     DATA+2
        BCI     1,PHI             GEØCENTRIC LATITUDE  - RADIANS
        PZE     DATA+9
        BCI     1,THETA           LØCAL SIDEREAL TIME  - RADIANS
        PZE     DATA+10
        BCI     1,H               ALTITUDE - FT
        PZE     DATA+11
        BCI     1,VSUBI           INERTIAL VELØCITY - FT/SEC
        PZE     DATA+12
        BCI     1,VSUBR           RELATIVE VELØCITY - FT/SEC
        PZE     DATA+15
        BCI     1,GAMMA           GAMMA - RADIANS
        PZE     DATA+16
        BCI     1,AZ              AZIMUTH - RADIANS
        PZE     DATA+17
        BCI     1,BETA
        PZE     DATA+18
        BCI     1,PIT             PITCH ANGLE - RADIANS
        PZE     DATA+21
        BCI     1,YAW             YAW ANGLE - RADIANS
        PZE     DATA+22
        BCI     1,PRATE           PITCH RATE - RADIANS/SEC
        PZE     DATA+23
        BCI     1,YRATE           YAW RATE - RADIANS/SEC
        PZE     DATA+24
        BCI     1,RATM            ATØMØSPHERIC DENSITY - SLUGS/FT**3
        PZE     DATA+25
        BCI     1,CATM            LØCAL SPEED ØF SØUND - FT/SEC
        PZE     DATA+26
        BCI     1,TATM            ATMØSPHERIC TEMPERATURE - RANKINE
        PZE     DATA+27
        BCI     1,PATM            ATMØSPHERIC PRESSURE - LB/IN**2
        PZE     DATA+28
        BCI     1,Q               DYNAMIC PRESSURE - LB/FT**2
        PZE     DATA+29
        BCI     1,MACH            MACH NUMBER - (VSUBR/CATM)
        PZE     DATA+30
        BCI     1,AØAP            ANGLE ØF ATTACK IN PITCH PLANE - RAD
        PZE     DATA+31
        BCI     1,AØAY            ANGLE ØF ATTACK IN YAW PLANE - RADIANS
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
        PZE     DATA+32
        BCI     1,AØA              TØTAL ANGLE ØF ATTACK - RADIANS
        PZE     DATA+33
        BCI     1,AØAD             TØTAL ANGLE ØF ATTACK - DEGREES
        PZE     DATA+34
        BCI     1,CSUBN            NØRMAL FØRCE CØEFFICIENT
        PZE     DATA+35
        BCI     1,CSUBA            AXIAL FØRCE CØEFFICIENT
        PZE     DATA+36
        BCI     1,CSUBL            LIFT CØEFFICIENT
        PZE     DATA+37
        BCI     1,CSUBD            DRAG CØEFFICIENT
        PZE     DATA+38
        BCI     1,LIFT             LIFT - LB
        PZE     DATA+39
        BCI     1,DRAG             DRAG - LB
        PZE     DATA+40
        BCI     1,THRST            THRUST - LB
        PZE     DATA+41
        BCI     1,GRAV             GRAVITY
        PZE     DATA+42
        BCI     1,LAT              GEØDETIC LATITUDE - DEGREES
        PZE     DATA+43
        BCI     1,LØNG             LØNGITUDE WEST ØF GREENWICH - DEGREES
        PZE     DATA+44
        BCI     1,RANGE            RANGE - FEET
        PZE     DATA+45
        BCI     1,ESUBP            PØTENTIAL ENERGY - FT**2/SEC**2
        PZE     DATA+46
        BCI     1,ESUBK            KINETIC ENERGY - FT**2/SEC**2
        PZE     DATA+47
        BCI     1,ENRGY            ESUBK/ESUBP
        PZE     DATA+48
*      DATA IN /DUMPY/ CØMMØN BLØCK
        BCI     1,DUMPY            CØMMØN BLØCK /DUMPY/
        PZE     DUMPY,,90          LØCATIØN,,LENGTH
        BCI     1,DUMPO
        PZE     DUMPY+0
        BCI     1,DUMPØ
        PZE     DUMPY+0
        BCI     1,DUMP1
        PZE     DUMPY+10
        BCI     1,DUMPI
        PZE     DUMPY+10
        BCI     1,DUMP2
        PZE     DUMPY+20
        BCI     1,DUMP3
        PZE     DUMPY+30
        BCI     1,DUMP4
        PZE     DUMPY+40
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
       BCI     1,DUMP5
       PZE     DUMPY+50
       BCI     1,DUMP6
       PZE     DUMPY+60
       BCI     1,DUMP7
       PZE     DUMPY+70
       BCI     1,DUMP8
       PZE     DUMPY+80
       BCI     1,DUMP9
       PZE     DUMPY+90
*
       BCI     1,STUDY           CØMMØN BLØCK /STUDY/
       PZE     STUDY,,10         LØCATIØN,,LENGTH
       BCI     1,PSTDY           PARAMETER STUDY INPUT
       PZE     STUDY
*
       BCI     1,1SRCH           CØMMØN BLØCK /1PSRCH/
       PZE     SRCH1A,,12        LØCATIØN,,LENGTH
       BCI     1,1PRS1           ØNE-PARAMETER SEARCH INPUT
       PZE     SRCH1A
*
       BCI     1,2SRCH           CØMMØN BLØCK /2PSRCH/
       PZE     SRCH2A,,21        LØCATIØN,,LENGTH
       BCI     1,2PRS1           TWØ-PARAMETER SEARCH INPUT
       PZE     SRCH2A
*
       BCI     1,MNX1            CØMMØN BLØCK /MNX1/
       PZE     MNX1,,10          LØCATIØN,,LENGTH
       BCI     1,MNMX1           MIN-MAX SEARCH INPUT
       PZE     MNX1
*
       BCI     1,MNX2            CØMMØN BLØCK /MNX2/
       PZE     MNX2,,10          LØCATIØN,,LENGTH
       BCI     1,MNMX2           MIN-MAX SEARCH INPUT - BLØCK 2
       PZE     MNX2
*
       BCI     1,1SRC2           CØMMØN BLØCK /1PSRC2/
       PZE     SRCH1B,,12        LØCATIØN,,LENGTH
       BCI     1,1PRS2           ØNE-PARAMETER SEARCH INPUT BLØCK 2
       PZE     SRCH1B
*
*     PRØPELLANT LØADING INPUT
       BCI     1,PRØPL           CØMMØN BLØCK /PRØPL/
       PZE     PRØPL,,40         LØCATIØN,,LENGTH
       BCI     1,PRØPD
       PZE     PRØPL+0
       BCI     1,PØØP            FLAG TØ END SECTIØN ØN PRØPELLANT
       BCI     1,PØØP
       BCI     1,MØØN            MØØN
       PZE     0
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

```
         BCI      1,MERC            MERCURY
         PZE      1
         BCI      1,VENUS           VENUS
         PZE      2
         BCI      1,MARS            MARS
         PZE      4
         BCI      1,JUPIT           JUPITER
         PZE      5
         BCI      1,SATRN           SATURN
         PZE      6
         BCI      1,URANS           URANUS
         PZE      7
         BCI      1,NEPTN           NEPTUNE
         PZE      8
         BCI      1,COAST
         PZE      COAST
         BCI      1,INJEC
         PZE      INJEC
         BCI      1,SKIP
         PZE      SKIP
         BCI      1,ASTRO           COMMON BLOCK /ASTRO/
         PZE      ASTRO,,4          LOCATION,,LENGTH
         BCI      1,OBART           COMMON BLOCK /OBART/
         PZE      OBART,,5          LOCATION,,LENGTH
         BCI      1,PBART           COMMON BLOCK /PBART/
         PZE      PBART,,23         LOCATION,,LENGTH
         BCI      1,TOL             COMMON BLOCK /TOL/
         PZE      TOL,,7            LOCATION,,LENGTH
         BCI      1,
ENDT     PZE
         END
```

FIGURE 48. SYMBOLIC LISTING OF SUBROUTINE SYMTAB (cont.)

XXXV. SUBROUTINE BLKDTA

This subroutine presets all "trajectory" input variables to various nominal values to be used, unless they are redefined by input.

CALLING SEQUENCE

This subroutine is not executable; it has no calling sequence. It is composed only of data values to be loaded with the rest of the program before execution.

USAGE

The following data in the various COMMON blocks are generated through use of data statements:

<u>COMMON Block /TAPES/</u>

ITAPE = 5

OTAPE = 6

STAPE = 6

<u>COMMON Block /CONST/</u>

RSUBO = 2.092643 E7

FLAT = 0.3352105 E - 2

GSUBO = 32.1740485
GSUB1 = 32.174
OMEGA = 7.2921158 E-5
PSL = 14.6959243
CMAX = 922.78
HMIN = -1000.0
HMAX = 300000.0
TAGS(1) = 57.2957795 deg/rad
TAGS(2) = 0.164579163 E-3 n.m./ft
TAGS(3) = 0.3048006 E-3 Km/ft
TAGS(4) = -1.0
TAGS(5) = 0.0
TAGS(6) = 0.0
TAGS(7) = 0.0
INFIN = 10.0 E10
PI = 3.14159265
PIO2 = 1.57079633
DTRAD = 0.1745329252 E-1
DPRAD = 57.2957795

COMMON Block /INPUT/

TI = 0.0
VI = 0.0
HI = 0.0
LATI = 28.5
LONGI = 80.8333
BETAI = 0.0

AZI = 90.0
BETA0 = 0.0
WI = 0.0
WPL = 0.0
SIDETI = 0.0
DATE(1) = 0.0
DATE(2) = 0.0
DATE(3) = 0.0
DATE(4) = 0.0
DATE(5) = 0.0
DATE(6) = 0.0
DATE(7) = 5.0
TARGET = 0.0
TRANS = 0.0
FLTR = 0
FLSC = 0
FLPRT = 0
SECTS = All section data is set to zero except:

Stage indicator = 1 for sections 1 and 2
= 2 for sections 3 - 12

Thrust indicator = 1 for all sections

Integration interval = 1.0 for section 1
= 2.0 for sections 2 - 12

Section number = 2N for section N

STAGS = All stage data is set to zero except thrust, flow rate, pitch, yaw, lift and drag multipliers which are set to 1.0

HEAD = All headers are assembled as blanks

COMMON Block /GAIND/

FLPAR = 0
IVAR(1 - 10) = 0
DVAR(1 - 10) = 0

COMMON Block /TABLES/

TMACH(1 - 30, 1 - 6) = 0.0
TCSBN (1 - 30, 1 - 6) = 0.0
TCSBA (1 - 30, 1 - 6) = 0.0
TTPIT (1 - 30, 1 - 6) = 0.0
TPIT (1 - 30, 1 - 6) = 0.0
TTYAW (1 - 30, 1 - 6) = 0.0
TYAW (1 - 30, 1 - 6) = 0.0
TTTRS (1 - 30, 1 - 6) = 0.0
TTRUS (1 - 30, 1 - 6) = 0.0
TTWDT (1 - 30, 1 - 6) = 0.0
TWDOT(1 - 30, 1 - 6) = 0.0
JCSBN (1 - 6) = 1
JCSBA (1 - 6) = 1
JPIT (1 - 6) = 1
JYAW (1 - 6) = 1
JTRUS (1 - 6) = 1
JWDOT(1 - 6) = 1
IMACH(1 - 6) = 30
IPIT (1 - 6) = 30
IYAW (1 - 6) = 30
ITRUS (1 - 6) = 30
IWDOT(1 - 6) = 30

COMMON Block /ENGINI/

YBBAR (1 - 5, 1 - 6) = 0.0
YBAR (1 - 5, 1 - 6) = 0.0
XBAR (1 - 6, 1 - 6) = 0.0
KPC (1 - 6, 1 - 6) = 0.0
KWOC (1 - 6, 1 - 6) = 0.0
KWFC (1 - 6, 1 - 6) = 0.0
KWOG (1 - 6, 1 - 6) = 0.0
KWFG (1 - 6, 1 - 6) = 0.0
CTOX (1 - 6, 1 - 6) = 0.0
CTFL (1 - 6, 1 - 6) = 0.0
MISC (1 - 30, 1 - 6) = 0.0
TANKS (1 - 30, 1 - 6) = 0.0

COMMON Block /DUMPY/

DUMP0 (1 - 10) = 0.0
DUMP1 (1 - 10) = 0.0
DUMP2 (1 - 10) = 0.0
DUMP3 (1 - 10) = 0.0
DUMP4 (1 - 10) = 0.0
DUMP5 (1 - 10) = 0.0
DUMP6 (1 - 10) = 0.0
DUMP7 (1 - 10) = 0.0
DUMP8 (1 - 10) = 0.0
DUMP9 (1 - 10) = 0.0

COMMON Block /ASTRO/

KSUBS = 0.0172020989

KSUBE = 0.07436574

ECLIP = 0.409183473

KINVE = 13.44705

SUBROUTINE BLKDTA NOMENCLATURE

Symbol	Description	Units	Reference
AZI	Initial or launch azimuth	degrees	/INPUT/
BETA0	Kick at end of vertical rise	degrees	/INPUT/
BETAI	Initial angle between local vertical and velocity vector	degrees	/INPUT/
CMAX	Speed of sound above 300,000 feet	ft/sec	/CONST/
CTFL	Fuel temperature polynomial coefficients	—	/ENGINI/
CTOX	Oxidizer temperature polynomial coefficients	—	/ENGINI/
DATE	Launch data - year, month, day, hour, minutes, seconds, time zone	—	/INPUT/
DPRAD	Degrees per radian conversion factor	deg/rad	/CONST/
DTRAD	Degrees to radians conversion factor	rad/deg	/CONST/
DUMP0	Dump vector - data to dump after input	—	/DUMPY/
DUMP1	Dump vector - data to dump after trajectory	—	/DUMPY/
DUMP2	Dump vector - data to dump in case of main error	—	/DUMPY/

SUBROUTINE BLKDTA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DUMP3	Dump vector - data to dump after section set-up	—	/DUMPY/
DUMP4	Dump vector - data to dump after discontinuity	—	/DUMPY/
DUMP5	Dump vector - data to dump after integration	—	/DUMPY/
DUMP6	Dump vector - data to dump after end-of-flight	—	/DUMPY/
DUMP7	Dump vector - data to dump in case of trajectory error	—	/DUMPY/
DUMP8	Dump vector - data to dump after derivative calculations	—	/DUMPY/
DUMP9	Dump vector - data to dump before integration abort	—	/DUMPY/
DVAR	Dependent variables	—	/GAIND/
ECLIP	Obliquity of ecliptic	radians	/ASTRO/
FLAT	Earth flattening	—	/CONST/
FLPAR	Flag to compute partials	—	/GAIND/
FLPRT	Print flag	—	/INPUT/
FLSC	Flag for number of sections	—	/INPUT/
FLTR	Flag for type of run	—	/INPUT/
GSUB0	Surface gravitational force	ft/sec^2	/CONST/
GSUB1	Weight to mass conversion factor	lb/slug	/CONST/

SUBROUTINE BLKDTA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
HEAD	Headers	—	/INPUT/
HI	Initial altitude	ft	/INPUT/
HMAX	Maximum altitude of atmosphere	ft	/CONST/
HMIN	Minimum altitude before crash	ft	/CONST/
IMACH	Sizes of mach number tables	—	/GAIND/
INFIN	"Infinity"	—	/CONST/
IPIT	Sizes of pitch tables	—	/GAIND/
ITAPE	Input tape	—	/TAPES/
ITRUS	Sizes of thrust tables	--	/GAIND/
IVAR	Independent variables	—	/GAIND/
IWDOT	Sizes of flow rate tables	—	/GAIND/
IYAW	Sizes of yaw tables	—	/GAIND/
JCSBA	Interpolation degree for C_A	—	/GAIND/
JCSBN	Interpolation degree for C_N	—	/GAIND/
JPIT	Interpolation degree for pitch	—	/GAIND/
JTRUS	Interpolation degree for thrust	—	/GAIND/
JWDOT	Interpolation degree for flow rate	—	/GAIND/
JYAW	Interpolation degree for yaw	—	/GAIND/
KINVE	(1)/(KSUBE)	min/e.r.$^{1.5}$	/ASTRO/

SUBROUTINE BLKDTA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
KPC	Chamber pressure influence coefficients	—	/ENGINI/
KSUBE	Earth gravitational constant	$e.r.^{1.5}/min$	/ASTRO/
KSUBS	Solar gravitational constant	$a.u.^{1.5}/min$	/ASTRO/
KWFC	Chamber fuel flow rate influence coefficients	—	/ENGINI/
KWFG	Gaseous fuel flow rate influence coefficients	—	/ENGINI/
KWOC	Chamber oxidizer flow rate influence coefficients	—	/ENGINI/
KWOG	Gaseous oxidizer flow rate influence coefficients	—	/ENGINI/
LATI	Initial latitude	degrees	/INPUT/
LONGI	Initial longitude west of Greenwich	degrees	/INPUT/
MISC	Miscellaneous engine data	—	/ENGINI/
OMEGA	Earth rotational rate	rad/sec	/CONST/
OTAPE	Output tape	—	/TAPES/
PI	π	—	/CONST/
PIO2	$\pi/2.0$	—	/CONST/
PSL	Sea level atmospheric pressure	lb/in^2	/CONST/

SUBROUTINE BLKDTA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RSUB0	Earth equatorial radius	ft	/CONST/
SECTS	Section data	—	/INPUT/
SIDETI	Initial sidereal time of launch	degrees	/INPUT/
STAGS	Stage data	—	/INPUT/
STAPE	Scratch tape	—	/TAPES/
TAGS	Conversion factors	—	/CONST/
TANKS	Tank data	—	/ENGINI/
TARGET	Target body	—	/INPUT/
TCSBA	Tables of axial force coefficient, C_A	—	/GAIND/
TCSBN	Tables of normal force coefficient, C_N	—	/GAIND/
TI	Initial time	sec	/INPUT/
TMACH	Tables of Mach number	—	/GAIND/
TPIT	Tables of pitch angle	degrees	/GAIND/
TRANS	Transfer time to target body	days (Planets) hours (Moon)	/INPUT/
TTPIT	Tables of pitch time	sec	/GAIND/
TTRUS	Tables of thrust force	lb	/GAIND/

SUBROUTINE BLKDTA NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TTTRS	Tables of thrust time	sec	/GAIND/
TTWDT	Tables of flow rate time	sec	/GAIND/
TTYAW	Tables of yaw time	sec	/GAIND/
TWDOT	Tables of flow rate	lb/sec	/GAIND/
TYAW	Tables of yaw angle	degrees	/GAIND/
VI	Initial velocity	ft/sec	/INPUT/
WI	Initial missile weight	lb	/INPUT/
WPL	Payload weight	lb	/INPUT/
XBAR	Independent engine parameter nominal values	—	/ENGINI/
YBAR	Dependent engine parameter nominal values	—	/ENGINI/
YBBAR	Dependent engine parameter specific engine values	—	/ENGINI/

```
$IBFTC BLKDTA  FULIST,REF
       BLØCK DATA
       CØMMØN/TAPES/ITAPE,ØTAPE,STAPE
       INTEGER  ØTAPE,STAPE
       DATA  ITAPE,ØTAPE,STAPE/5,6,6/
       CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
      $              TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
       EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
       REAL  INFIN
       DATA  RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,TAGS,INFIN,
      $        PI,PIØ2,DTRAD,DPRAD/2.092643E7,0.3352105E-2,32.1740485,
      $        32.174,7.2921158E-5,14.6959243,922.78,-1000.0,300000.0,
      $        57.2957795,0.164579163E-3,0.3048006E-3,-1.0,3*0.0,10.0E10,
      $        3.14159265,1.57079633,0.1745329252E-1,57.2957795/
       CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,
      $              DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
      $              SECTS(20,12),STAGS(20,6),HEAD(11,15)
       REAL  LATI,LØNGI
       INTEGER  TARGET
       INTEGER  FLTR,FLSC,FLPRT
       DATA  TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,DATE,
      $        TARGET,TRANS,FLTR,FLSC,FLPRT,SECTS,STAGS,HEAD/
      $        0.0,0.0,0.0,28.5,80.8333,0.0,90.0,0.0,0.0,0.0,0.0,
      $        6*0.0,5.0,0,0.0,0,0,0,
      $        3*0.0,1.0,1.0,0.0,1.0,8*0.0,2.0,4*0.0,
      $        3*0.0,1.0,1.0,0.0,2.0,8*0.0,4.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,6.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,8.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,10.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,12.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,14.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,16.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,18.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,20.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,22.0,4*0.0,
      $        3*0.0,2.0,1.0,0.0,2.0,8*0.0,24.0,4*0.0,
      $        11*0.0,6*1.0,3*0.0,11*0.0,6*1.0,3*0.0,11*0.0,6*1.0,3*0.0,
      $        11*0.0,6*1.0,3*0.0,11*0.0,6*1.0,3*0.0,11*0.0,6*1.0,3*0.0,
      $        165*6H      /
       CØMMØN /IPRINT/ PRNTØN(6),PRNTØF(6)
       DATA  PRNTØN,PRNTØF/6*0.0,6*0.0/
       CØMMØN /GAIND/ FLPAR,IVAR(10),DVAR(10)
       INTEGER  FLPAR,DVAR
       DATA  FLPAR,IVAR,DVAR/0,10*0,10*0/
       CØMMØN/TABLES/TMACH(30,6),TCSBN(30,6),TCSBA(30,6),
      $             TTPIT(30,6),TPIT(30,6),TTYAW(30,6),TYAW(30,6),
      $             TTTRS(30,6),TTRUS(30,6),TTWDT(30,6),TWDØT(30,6),
      $             JCSBN(6),JCSBA(6),JPIT(6),JYAW(6),JTRUS(6),JWDØT(6),
      $             LPIT(6),LYAW(6),LTRUS(6),LWDØT(6),
      $             KMACH(6),KPIT(6),KYAW(6),KTRUS(6),KWDØT(6),
      $             IMACH(6),IPIT(6),IYAW(6),ITRUS(6),IWDØT(6)
       DATA  TMACH,TCSBN,TCSBA,TTPIT,TPIT,TTYAW,TYAW,TTTRS,TTRUS,TTWDT,
```

FIGURE 49. SYMBOLIC LISTING OF SUBROUTINE BLKDTA

```
$         TWDØT/180*0.0,180*0.0,180*0.0,180*0.0,180*0.0,180*0.0,
$         180*0.0,180*0.0,180*0.0,180*0.0,180*0.0/,
$         JCSBN,JCSBA,JPIT,JYAW,JTRUS,JWDØT/6*1,6*1,6*1,6*1,6*1,6*1/
$              ,LPIT,LYAW,LTRUS,LWDØT/6*0,6*0,6*0,6*0/,
$         IMACH,IPIT,IYAW,ITRUS,IWDØT/6*30,6*30,6*30,6*30,6*30/
 CØMMØN/ENGINI/ YBBAR(5,6),YBAR(5,6),XBAR(6,6),KPC(6,6),KWØC(6,6),
$           KWFC(6,6),KWØG(6,6),KWFG(6,6),CTØX(6,6),CTFL(6,6),
$           MISC(30,6),TANKS(30,6)
 REAL  KPC,KWØC,KWFC,KWØG,KWFG,MISC
 DATA  YBBAR,YBAR,XBAR,KPC,KWØC,KWFC,KWØG,KWFG,CTØX,CTFL,MISC,TANKS
$         /30*0.0,30*0.0,36*0.0,36*0.0,36*0.0,36*0.0,36*0.0,36*0.0,
$         36*0.0,36*0.0,180*0.0,180*0.0/
 CØMMØN /DUMPY/ DUMP0(10),DUMP1(10),DUMP2(10),DUMP3(10),DUMP4(10),
$              DUMP5(10),DUMP6(10),DUMP7(10),DUMP8(10),DUMP9(10)
 DATA DUMP0,DUMP1,DUMP2,DUMP3,DUMP4,DUMP5,DUMP6,DUMP7,DUMP8,DUMP9/
$         10*0.0,10*0.0,10*0.0,10*0.0,10*0.0,10*0.0,10*0.0,10*0.0,
$         10*0.0,10*0.0/
 CØMMØN /PRØPL/ PRØPD(28)
 DATA  PRØPD/30.0,5.0,26*0.0/
 CØMMØN/ASTRØ/KSUBS,KSUBE,ECLIP,KINVE
 REAL  KSUBS,KSUBE,KINVE
 DATA  KSUBS,KSUBE,ECLIP,KINVE/0.0172020989,0.07436574,0.409183473,
$      13.44705/
 END
```

FIGURE 49. SYMBOLIC LISTING OF SUBROUTINE BLKDTA (cont.)

XXXVI. SUBROUTINE UNPACK

This subroutine is used to "decode" and execute logical control data input by FINPUT using the "A"-prefix conversion code option. This code sets up a 4-part word — address, tag, decrement, and prefix — where the address and decrement are locations and the tag and prefix are integers from 0 to 7. There are 6 entry points.

1. Entry UNPACK

CALLING SEQUENCE

CALL UNPACK (WORD, ARRAY)

WORD — data word to be decoded
ARRAY(1) — ADD; address portion of WORD
ARRAY(2) — TAG ; tag portion of WORD
ARRAY(3) — DEC; decrement portion of WORD
ARRAY(4) — PRE ; prefix portion of WORD

USAGE

The decrement of WORD is tested. If it is non-zero, but less than PROG(PROG is defined to be equivalent to 1500_{10}, i.e., the approximate computer location where the system ends and the program begins), the decrement is assumed to be a subscript associated with the address specified in WORD, rather than an independent location; the location returned in ADD is the subscripted location and DEC is returned as zero.

2. Entry XEC

CALLING SEQUENCE

CALL XEC (FUNC)

USAGE

FUNC is tested and if it is zero, a return is made. If FUNC is non-zero, it is assumed to contain the location of a subroutine which is called before returning.

A logical word input using the "A"-prefix conversion code may specify a routine to be executed.

3. Entry CLA

CALLING SEQUENCE

CLA is a function routine and is "called" by using it as a function.

A = CLA (ADD)

USAGE

CLA is used to set A equal to the value of the location whose address is contained in ADD.

A logical word input using the "A"-prefix conversion code will usually contain the address of a parameter whose value is to be used in some way.

4. Entry STO

CALLING SEQUENCE

CALL STO (ADD, X)

USAGE

STO is used to store the value of X in the location specified in the address portion of ADD.

5. Entry LOC

CALLING SEQUENCE

LOC is a function routine and is "called" by using it as a function.

LX = LOC (X)

USAGE

LOC sets LX to be equal to the location of X.

LOC is used to compare parameter locations with locations specified in the "A"-prefix conversion code input.

6. Entry MASKA

CALLING SEQUENCE

MASKA is a function routine and is "called" by using it as a function.

MX = MASKA (X)

USAGE

MASKA sets MX equal to the Boolean Product (logical and) of X and the 12-character octal word 000000077777; this is equivalent to preserving the address portion of a logical word while setting the tag, decrement and prefix portions to zero.

MASKA is necessary when using Subroutine LOOKUP to look-up the Hollerith name associated with a parameter location.

```
$IBMAP UNPACK
PRØG   EQU      1500
*
*      CALL     UNPACK(WØRD,ARRAY)
*                 WØRD - WØRD TØ BE UNPACKED
*                 DIMENSIØN  ARRAY(4)
*                 ARRAY(1) = ADD
*                 ARRAY(2) = TAG
*                 ARRAY(3) = DEC
*                 ARRAY(4) = PRE
*                  ADD - ADDRESS (+ DECREMENT - 1 IF DECREMENT .LT. 1500)
*                  TAG - TAG = 0-7
*                  DEC - DECREMENT (IF DECREMENT .GT. 1500) ØR ZERØ
*                  PRE - PREFIX = 0-7
*
UNPACK SAVE     (4,1,2)
       CAL      4,4
       STA      ADD
       ADD      =1
       STA      TAG
       ADD      =1
       STA      DEC
       ADD      =1
       STA      PRE
       CAL*     3,4
       PAX      ,1                 ADDRESS
       PDX      ,2                 DECREMENT
       LRS      18
       ARS      15
PRE    STØ      **
       CLM
       LLS      3
TAG    STØ      **
       STZ*     DEC
       TXL      *+4,2,PRØG
       PXA      ,2
DEC    STØ      **
       AXT      0,2
       TXL      *+2,2,0
       TXI      *+1,2,-1
       SXD      *+1,2
       TXI      *+1,1,**
       PXA      ,1
ADD    STØ      **
       RETURN   UNPACK
*
*      CALL     XEC(FUNCG)
*                 IF CØNTENTS ØF FUNCG ARE NØN-ZERØ, FUNCG
*                  IS ASSUMED TØ BE A SUBRØUTINE TØ BE EXECUTED
*
```

FIGURE 50. SYMBOLIC LISTING OF SUBROUTINE UNPACK

```
XEC      SAVE
         CLA*     3,4
         TZE      1,4                    FUNCG NØT A SUBRØUTINE
         STA      *+1
         CALL     **                     EXECUTES FUNCG SUBRØUTINE
         RETURN   XEC
*
*        FUNCTIØN CLA(LØC)
*                    RETURNS WITH CØNTENTS ØF LØCATIØN GIVEN IN ADDRESS
*                     ØF LØC   --   I.E.   DØES A CLA* LØC
         ENTRY    CLA
CLA      CLA*     3,4
         STA      *+1
         CLA      **
         TRA      1,4
*
*       STØ STØRES THE CØNTENTS ØF X IN THE LØCATIØN SPECIFIED BY LØC
*
*        CALL STØ(LØC,X)
*
STØ      SAVE
         CLA*     3,4
         STA      *+2
         CLA*     4,4
         STØ      **
         RETURN   STØ
*
*
*        FUNCTIØN LØC(Z) RETURNS WITH LØCATIØN ØF Z IN ACCUMULATØR
         ENTRY    LØC
LØC      CLA      3,4
         ANA      =Ø777777
         TRA      1,4
*
*
*       FUNCTIØN MASKA
*                 MASKS ØUT ALL BUT ADDRESS
*
         ENTRY    MASKA
MASKA    CLA*     3,4
         ANA      =Ø77777
         TRA      1,4
*
*
         END
```

FIGURE 50. SYMBOLIC LISTING OF SUBROUTINE UNPACK (cont.)

XXXVII. SUBROUTINE LOOKUP

This subroutine searches the /BCIPZE/ symbol table for the Hollerith name corresponding to a parameter location.

CALLING SEQUENCE

The calling sequence is:

CALL LOOKUP (LOC, NAME)

LOC contains the parameter location.

NAME is the location to store the Hollerith name.

This subroutine uses COMMON block /BCIPZE/.

SOLUTION METHOD

A table search is done in the following manner:

```
1.  M = | TABLE(1) | + 1

2.  Do 3, I = 3, M, 2

3.  If LOC = TABLE(I), go to 4

    If LOC ≠ TABLE(I), continue
```

```
4.  I = I - 1

5.  LNAME = TABLE(I)

    Return
```

Note:

Although TABLE is defined to be of dimension 3, its actual length is specified by its first entry, TABLE(1).

SUBROUTINE LOOKUP NOMENCLATURE

Symbol	Description	Units	Reference
LOC	Location of parameter	—	CALL, 3
M	Table dimension	—	1, 2
NAME	Name of parameter	—	CALL, 5
TABLE	Symbol table	—	/BCIPZE/, INTEGER, 1, 3, 5

```
$IBFTC LØØKUP  FULIST,REF
      SUBRØUTINE LØØKUP(LØC,LNAME)
C        LØØK UP BCI NAME ØF LØC IN FINK SYMBØL TABLE
      CØMMØN /BCIPZE/TABLE(3)
      INTEGER  TABLE
      M = IABS(TABLE(1)) + 1
      DØ 10  I=3,M,2
   10 IF (LØC .EQ. TABLE(I))  GØ TØ 11
   11 I = I - 1
      LNAME = TABLE(I)
      RETURN
      END
```

FIGURE 51. SYMBOLIC LISTING OF SUBROUTINE LOOKUP

XXXVIII. SUBROUTINE ERROR

This subroutine is used for communication of error conditions. It has 6 entry points which reference the same error flag.

1. Entry SETERR

CALLING SEQUENCE

```
CALL SETERR (Z)
```

USAGE

SETERR sets the error flag equal to Z.

2. Entry CHKERR

CALLING SEQUENCE

```
CALL CHKERR (Z)
```

USAGE

CHKERR is used to check the status of the error flag. It sets Z equal to the error flag; the error flag is unchanged.

3. Entry ANSERR

CALLING SEQUENCE

CALL ANSERR (Z)

USAGE

ANSERR resets the error flag to the 36-bit Boolean Product (logical and) of Z and the error flag.

4. Entry ORSERR

CALLING SEQUENCE

CALL ORSERR (Z)

USAGE

ORSERR resets the error flag to the 36-bit Bollean Sum (logical or) of Z and the error flag.

5. Entry ANAERR

CALLING SEQUENCE

ANAERR is a function routine and is "called" by using it as a function.

A = ANAERR (Z)

USAGE

ANAERR is used to check individual bits in the 36-bit error flag. A will be set to the Boolean Product (logical and) of Z and the error flag; the error flag is unchanged.

6. Entry ORAERR

CALLING SEQUENCE

ORAERR is a function routine and is "called" by using it as a function.

A = ORAERR (Z)

USAGE

ORAERR is used to check individual bits in the 36-bit error flag. A will be set to the Boolean Sum (logical or) of Z and the error flag; the error flag is unchanged.

```
$IBMAP ERRØR
*
*     RØUTINES TØ SET, RESET, TEST, AND CHECK ERRØR FLAG
*
*      CALL    SETERR(Z)
*                WILL STØRE CØNTENTS ØF Z IN ERRØR FLAG
*      CALL    CHKERR(Z)
*                WILL STØRE CØNTENTS ØF ERRØR FLAG IN Z
*      CALL    ANSERR(Z)
*                WILL @AND@ Z TØ THE ERRØR FLAG IN STØRAGE
*      CALL    ØRSERR(Z)
*                WILL @ØR@ Z TØ ERRØR FLAG IN STØRAGE
*      FUNCTIØN ANAERR(Z)
*                WILL @AND@ Z AND ERRØR FLAG IN ACCUMULATØR
*      FUNCTIØN ØRAERR(Z)
*                WILL @ØR@ Z AND ERRØR FLAG IN ACCUMULATØR
*
       ENTRY   SETERR
       ENTRY   CHKERR
       ENTRY   ANSERR
       ENTRY   ØRSERR
       ENTRY   ANAERR
       ENTRY   ØRAERR
*
SETERR CLA*    3,4
       STØ     EFLAG
       TRA     1,4
*
CHKERR CLA     EFLAG
       STØ*    3,4
       TRA     1,4
*
ANSERR CAL*    3,4
       ANS     EFLAG
       TRA     1,4
*
ØRSERR CAL*    3,4
       ØRS     EFLAG
       TRA     1,4
*
ANAERR CAL     EFLAG
       ANA*    3,4
       TRA     1,4
*
ØRAERR CAL     EFLAG
       ØRA*    3,4
       TRA     1,4
*
EFLAG  PZE
*
       END
```

FIGURE 52. SYMBOLIC LISTING OF SUBROUTINE ERROR

XXXIX. SUBROUTINE DUMPS

This subroutine will dump any named COMMON block identified in the FINK symbol table, together with the names of the "items" if they are also in the symbol table, on input option without recompiling the program. The type of dump is also specified on input.

CALLING SEQUENCE

CALL DUMPS (LIST, ILIST, ITAPE)

SOLUTION METHOD

Each element of the LIST array, if non-zero, is assumed to have the location of the common block to be dumped in the address portion of the word. The decrement specifies the length of the block, and the tag specifies the type of dump desired. The type options are as follows:

0 = octal dump

1 = real dump

2 = integer dump

3 = Hollerith dump (BCD)

The names of the various dump options are generated in the TYPES array by a DATA statement.

1. IT = | ITAPE |

2. Do 57, I = 1, ILIST

3. LDUMP = | LIST(I) |

A zero entry terminates processing of LIST, if not, all entries are used:

4. If LDUMP = 0, return

   If LDUMP ≠ 0, go to 5

Look up the name of the common block to be dumped in the symbol table:

5. CALL UNPACK (LDUMP, ARRAY)

6. NLOC = LDUMP - (TAG)(2^{15}) - (PRE)(2^{33})

7. CALL LOOKUP (NLOC, NAME)

Division by 2^{18} is equivalent to shifting the decrement portion to the address portion where it can be used as an integer:

8. LENGTH = LDUMP/2^{18}

9. ITAG = TAG + 1

10. TYPE(1) = TYPES(1, ITAG)

11. TYPE(2) = TYPES(2, ITAG)

The location to start dumping is found by removing all portions of the word LDUMP but the address:

```
12.  LOC = NLOC - (LENGTH)$(2^{18})$
```

If for some reason the length is zero, dump 1 item only:

```
13.  If LENGTH = 0, LENGTH = 1
     If LENGTH $\neq$ 0, continue
```

Write the header identification on the output unit(s):

```
14.  If IT = 0, go to 17
     If IT $\neq$ 0, go to 15
15.  Write NAME, LENGTH, and TYPE on tape IT using
     Format Statement #18
16.  If ITAPE > 0, go to 19
     If ITAPE $\leq$ 0, go to 17
17.  Print NAME, LENGTH, and TYPE on the on-line
     printer using Format Statement #18
     Go to 19
18.  Format (1H ,/1H ,/1H ,/1H ,5X,9HDUMP OF /,
               A5,12H/, LENGTH =,I6,10H, TYPE =,2A6)
19.  J = 1
```

```
20.  Do 25,  K = 1,6
```

Find names, if any, and values of each item in the block:

```
21.  CALL LOOKUP (LOC, NAMES(K))

22.  VALUE(K) = CLA(LOC)

23.  J = J + 1

24.  If J > LENGTH,  go to 26

     If J ≤ LENGTH,  go to 25

25.  LOC = LOC + 1
```

Write the names of the items:

```
26.  If IT = 0,  go to 29

     If IT ≠ 0,  go to 27

27.  Write (NAMES(L), L = 1, K) on tape IT using

     Format Statement #30

28.  If ITAPE > 0,  go to 31

     If ITAPE ≤ 0,  go to 29

29.  Print (NAMES(L), L = 1, K) on the on-line printer

     using Format Statement #30

     Go to 31

30.  Format (1H ,/1H ,6(5X,A6,5X))
```

Write the item values in the desired format:

```
31.  If ITAG = 1, go to 32
     If ITAG = 2, go to 38
     If ITAG = 3, go to 44
     If ITAG = 4, go to 50
```

Octal dump print:

```
32.  If IT = 0, go to 35
     If IT ≠ 0, go to 33
33.  Write (VALUE(L), L = 1, K) on tape IT using
     Format Statement #37
34.  If ITAPE > 0, go to 56
     If ITAPE ≤ 0, go to 35
35.  Print (VALUE(L), L = 1, K) on the on-line printer
     using Format Statement #37
36.  Go to 56
37.  Format (1H ,6(2X,O12,2X))
```

Real dump print:

```
38.  If IT = 0, go to 41
     If IT ≠ 0, go to 39
```

39. Write (VALUE(L), L = 1, K) on tape IT using

Format Statement #43

40. If ITAPE > 0, go to 56

If ITAPE ≤ 0, go to 41

41. Print (VALUE(L), L = 1, K) on the on-line printer

using Format Statement #43

42. Go to 56

43. Format (1H ,6(E16.8))

Integer dump print:

44. If IT = 0, go to 47

If IT ≠ 0, go to 45

45. Write (VALUE(L), L = 1, K) on tape IT using

Format Statement #49

46. If ITAPE > 0, go to 56

If ITAPE ≤ 0, go to 47

47. Print (VALUE(L), L = 1, K) on the on-line printer

using Format Statement #49

48. Go to 56

49. Format (1H ,6(2X,I12,2X))

Hollerith dump print:

```
50.  If $IT = 0$, go to 53
     If $IT \neq 0$, go to 51
51.  Write (VALUE(L), $L = 1, K$) on tape IT using
     Format Statement #55
52.  If $ITAPE > 0$, go to 56
     If $ITAPE \leq 0$, go to 53
53.  Print (VALUE(L), $L = 1, K$) on the on-line printer
     using Format Statement #55
54.  Go to 56
55.  Format (1H ,6(5X,A6,5X))
56.  If $J \leq LENGTH$, go to 20
     If $J > LENGTH$, go to 57
57.  Continue
     Return
```

SUBROUTINE DUMPS NOMENCLATURE

Symbol	Description	Units	Reference
ARRAY	Array containing address, tag, decrement and prefix of LDUMP	—	DIM, EQUIV, 5
ILIST	Dimension of LIST	—	CALL, DIM, 2
IT	Tape unit	—	1, 14, 15, 27, 33, 38, 39, 44, 45, 50, 51
ITAG	Dump option indicator	—	9, 10, 11, 31
ITAPE	Output unit	—	CALL, 1, 16, 28, 34, 40, 46, 52
LDUMP	Item in LIST array being	—	3, 4, 5, 6, 8
LENGTH	Length of block being dumped	—	8, 12, 13, 15, 17, 24, 56
LIST	Input array of blocks to be dumped	—	CALL, DIM, 3
LOC	Location of item to be dumped	—	12, 21, 22, 25
NAME	Hollerith name of block being dumped	—	7, 15, 17
NAMES	Array of Hollerith names for items being dumped	—	DIM, 21, 27, 29
NLOC	Starting location and length of COMMON block to be dumped	—	6, 7, 12
PRE	Prefix of LDUMP	—	INTEGER, EQUIV, 6

SUBROUTINE DUMPS NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TAG	Tag of LDUMP	—	INTEGER, EQUIV, 6, 9
TYPE	Hollerith name for current type of dump	—	DIM, 10, 11, 15, 17
TYPES	Array of dimension (2,6) of Hollerith names for types of dump options	—	DATA, DIM, 10, 11
VALUE	Array of dimension (6) of values of items to be dumped on one line of output	—	DIM, 22, 33, 35, 39, 41, 45, 47, 51, 53

```
$IBFTC DUMPS
      SUBRØUTINE DUMPS(LIST,ILIST,ITAPE)
      DIMENSIØN  LIST(ILIST)
C         LIST IS A VECTØR ØF DIMENSIØN I ØR LESS ØF THE NAMES AND
C           LENGTHS ØF THE BLØCKS TØ BE DUMPED
C                   PZE  BLØCK,,LENGTH       MAP FØRMAT
C         DUMPS CALLS LØØKUP TØ FIND THE NAMES ØF THE ITEMS AS SPECIFIED
C           IN THE FINK INPUT TABLE
C         ITAPE IS THE TAPE TØ WRITE DUMP ØN
C              IF ZERØ -- ØN-LINE PRINTER
C              IF NEG. -- BØTH ØN-LINE AND ØN TAPE
C
      DIMENSIØN TYPES(2,4),TYPE(2)
      DATA  TYPES/6HØCTAL ,6H      ,6HREAL  ,6H      ,6HINTEGE,6HR     ,
     $            6HBCD   ,6H      /
      DIMENSIØN  VALUE(6),NAMES(6)
      DIMENSIØN  ARRAY(4)
      EQUIVALENCE  (ARRAY(1),ADD),(ARRAY(2),TAG),(ARRAY(3),DEC),
     $             (ARRAY(4),PRE)
      INTEGER  ADD,TAG,DEC,PRE
C
      IT = IABS(ITAPE)
      DØ 100  I=1,ILIST
      LDUMP = IABS(LIST(I))
      IF (LDUMP .EQ. 0)  RETURN
      CALL UNPACK(LDUMP,ARRAY)
      NLØC = LDUMP - TAG*2**15 - PRE*2**33
      CALL LØØKUP(NLØC,NAME)
      LNGTH = LDUMP/2**18
      ITAG = TAG + 1
      TYPE(1) = TYPES(1,ITAG)
      TYPE(2) = TYPES(2,ITAG)
      LØC = NLØC - LENGTH*2**18
      IF (LENGTH .EQ. 0)  LENGTH = 1
C
      IF (IT .EQ. 0)  GØ TØ 10
      WRITE(IT,1)  NAME,LENGTH,TYPE
      IF (ITAPE .GT. 0)  GØ TØ 15
   10 PRINT 1,  NAME,LENGTH,TYPE
    1 FØRMAT(1H ,/1H ,/1H ,/1H ,5X,9HDUMP ØF /,A5,12H/,  LENGTH =,I6,
     $        10H,  TYPE = ,2A6)
C
   15 J = 1
   16 DØ 20  K=1,6
      CALL LØØKUP(LØC,NAMES(K))
      VALUE(K) = CLA(LØC)
      J = J + 1
      IF (J .GT. LENGTH)  GØ TØ 21
   20 LØC = LØC + 1
   21 IF (IT .EQ. 0)  GØ TØ 25
```

FIGURE 53. SYMBOLIC LISTING OF SUBROUTINE DUMPS

```
      WRITE(IT,2)  (NAMES(L),L=1,K)
      IF (ITAPE .GT. 0)  GØ TØ 30
   25 PRINT 2,  (NAMES(L),L=1,K)
    2 FØRMAT(1H ,/1H ,6(5X,A6,5X))
C
   30 GØ TØ (40,50,60,70), ITAG
C         ØCTAL DUMP
   40 IF (IT .EQ. 0)  GØ TØ 45
      WRITE(IT,3)  (VALUE(L),L=1,K)
      IF (ITAPE .GT. 0)  GØ TØ 80
   45 PRINT 3,  (VALUE(L),L=1,K)
      GØ TØ 80
    3 FØRMAT(1H ,6(2X,Ø12,2X))
C         REAL DUMP
   50 IF (IT .EQ. 0)  GØ TØ 55
      WRITE(IT,4)  (VALUE(L),L=1,K)
      IF (ITAPE .GT. 0)  GØ TØ 80
   55 PRINT 4,  (VALUE(L),L=1,K)
      GØ TØ 80
    4 FØRMAT(1H ,6(E16.8))
C         INTEGER DUMP
   60 IF (IT .EQ. 0)  GØ TØ 65
      WRITE(IT,5)  (VALUE(L),L=1,K)
      IF (ITAPE .GT. 0)  GØ TØ 80
   65 PRINT 5,  (VALUE(L),L=1,K)
      GØ TØ 80
    5 FØRMAT(1H ,6(2X,I12,2X))
C         BCD DUMP
   70 IF (IT .EQ. 0)  GØ TØ 75
      WRITE(IT,6)  (VALUE(L),L=1,K)
      IF (ITAPE .GT. 0)  GØ TØ 80
   75 PRINT 6,  (VALUE(L),L=1,K)
      GØ TØ 80
    6 FØRMAT(1H ,6(5X,A6,5X))
C
   80 IF (J .LE. LENGTH)  GØ TØ 16
  100 CØNTINUE
      RETURN
      END
```

FIGURE 53. SYMBOLIC LISTING OF SUBROUTINE DUMPS (cont.)

XL. SUBROUTINE PRNTKY

This subroutine prints an output key of the trajectory output.

CALLING SEQUENCE

CALL PRNTKY

This subroutine uses COMMON block TAPES.

METHOD

```
1.  I = $|OTAPE|$
2.  If I = 0,  go to 5
    If I $\neq$ 0,  go to 3
3.  Write the output key on tape I
4.  If OTAPE $>$ 0, return
    If OTAPE $\leq$ 0, go to 5
5.  Print the output key on the on-line printer
    Return
```

SUBROUTINE PRNTKY NOMENCLATURE

Symbol	Description	Units	Reference
I	Tape unit to write on	—	1, 2, 3
OTAPE	Output unit specification	—	/TAPES/, INTEGER, 1, 4

```
$IBFTC PRNTKY FULIST,REF
C        PRINT ØUTPUT KEY ØN BEGINNING ØF RUN
      SUBRØUTINE PRNTKY
      CØMMØN/TAPES/ITAPE,ØTAPE,STAPE
      INTEGER  ØTAPE,STAPE
      I = IABS(ØTAPE)
      IF (I .EQ. 0)  GØ TØ 10
      WRITE (I,1)
      WRITE (I,2)
      WRITE (I,3)
      WRITE (I,4)
      IF (ØTAPE .GT. 0)  RETURN
   10 PRINT 1
      PRINT 2
      PRINT 3
      PRINT 4
      RETURN
    1 FØRMAT(102H1                              TRAJECTØRY ØUTPUT KEY
     $                                                ///
     $102H             TIME = (SECØNDS)         STAGE =        SECTIØN =
     $                                         //
     $102H      1  REL. VELØCITY          HEIGHT             BETA
     $ AZIMUTH           RANGE          WEIGHT/
     $102H              (FT/SEC)          (FEET)        (DEGREES)       (
     $DEGREES)   (NAUT. MILES)        (PØUNDS)/
     $102H      2 INER. VELØCITY          THRUST (THRUST-DRAG)/W
     $LATITUDE       LØNGITUDE           W-DØT/
     $102H              (FT/SEC)        (PØUNDS)           (GEES)       (
     $DEGREES)       (DEGREES)        (LB/SEC)/
     $102H      3       MACH NØ.               Q      LIFT CØEFF.     DRA
     $G CØEFF.            LIFT            DRAG/
     $102H                            (LB/FT**2)
     $                (PØUNDS)        (PØUNDS)/)
    2 FØRMAT(102H      4   ATTACK ANGLE       REF. AREA            C-ATM
     $        RHØ-ATM           P-ATM           T-ATM/
     $102H             (DEGREES)         (FT**2)         (FT/SEC)   (SLUG
     $S/FT**3)      (LB/IN**2)       (DEG. F.)/
     $102H      5    PITCH ANGLE      PITCH RATE        YAW ANGLE
     $YAW RATE         GRAVITY         PAYLØAD/
     $102H             (DEGREES)       (DEG/SEC)        (DEGREES)       (
     $DEG/SEC)     (FT/SEC**2)        (PØUNDS)/
     $102H      6    FLIGHT TIME      STAGE TIME     SECTIØN TIME     ESU
     $BK/ESUBP                                 /
     $102H             (SECØNDS)       (SECØNDS)        (SECØNDS)
     $                                         //)
    3 FØRMAT(102H      7   X-(INERTIAL)    Y-(INERTIAL)     Z-(INERTIAL)
     $XDØT-(INERTIAL) YDØT-(INERTIAL) ZDØT-(INERTIAL)/
     $102H                (FEET)          (FEET)           (FEET)
     $(FT/SEC)        (FT/SEC)        (FT/SEC)/
     $102H      8     THRUST-(X)      THRUST-(Y)       THRUST-(Z)     GRA
```

FIGURE 54. SYMBOLIC LISTING OF SUBROUTINE PRNTKY

```
 $VITY-(X)     GRAVITY-(Y)      GRAVITY-(Z)/
 $102H               (PØUNDS)        (PØUNDS)         (PØUNDS)      (FT
 $/SEC**2)     (FT/SEC**2)      (FT/SEC**2)/
 $102H     9             DRAG-(X)         DRAG-(Y)         DRAG-(Z)
 $LIFT-(X)        LIFT-(Y)         LIFT-(Z)/
 $102H               (PØUNDS)        (PØUNDS)         (PØUNDS)
 $(PØUNDS)        (PØUNDS)         (PØUNDS)//
 $102H                                    ØPTIØNAL EXTRA PRINT LINE
 $                                          //)
4 FØRMAT(102H     10              TØ             RHOO               TF
 $           RHØF               PTØ             PTF/
 $102H             (DEG. R.)       (LB/FT**3)        (DEG. R.)       (L
 $B/FT**3)      (LB/IN**2)       (LB/IN**2)/
 $102H     11              WØ              WF               VØ
 $     VF               PØ              PF/
 $102H              (PØUNDS)         (PØUNDS)          (FT**3)
 $ (FT**3)      (LB/IN**2)       (LB/IN**2)/
 $102H     12      WØ(C)-DØT        WF(C)-DØT               PC
 $   MR(C)           CSUBF        THRUST(C)/
 $102H              (LB/SEC)         (LB/SEC)       (LB/IN**2)
 $                                 (PØUNDS)/
 $102H     13      WØ(E)-DØT        WF(E)-DØT         W(E)-DØT
 $   MR(E)          ISP(E)        THRUST(E)/
 $102H              (LB/SEC)         (LB/SEC)         (LB/SEC)
 $               (SECØNDS)         (PØUNDS)/
 $102H     14      WØ(S)-DØT        WF(S)-DØT         W(S)-D/T
 $   MR(S)          ISP(S)        THRUST(S)/
 $102H              (LB/SEC)         (LB/SEC)         (LB/SEC)
 $               (SECØNDS)         (PØUNDS)/)
  END
```

FIGURE 54. SYMBOLIC LISTING OF SUBROUTINE PRNTKY (cont.)

XLI. SUBROUTINE OUTPUT

This subroutine is the main output routine for Blocks II and III. It is entered at specific intervals during flight (which is an input item) or at discontinuities.

CALLING SEQUENCE

CALL OUTPUT

SOLUTION METHOD

Convert from radians to degrees.

1. BETAD = (BETA)(DPRAD)
2. AZD = (AZ)(DPRAD)
3. PITD = (PIT)(DPRAD)
4. PRATED = (PRATE)(DPRAD)
5. YAWD = (YAW)(DPRAD)
6. YRATED = (YRATE)(DPRAD)

Set output tape assignment.

7. TAPE = $|$ OTAPE $|$

```
Check error flag.

 8.  CALL CHKERR(Z)

 9.  If $Z \neq 0$, go to 34
     If $Z = 0$, go to 10

Set input print flag.

10.  PFLAG = FLPRT

Check type of run.

11.  If $FLTRW \neq 0$, PFLAG = (FLPRT)/(8)
     If $FLTRW = 0$, go to 12

Check for end of flight.

12.  If FLEOF not true, go to 21
     If FLEOF true, go to 13

Check for end-of-flight print option.

13.  If $((PFLAG)/(2))(2) = PFLAG$, return
     If $((PFLAG)/(2))(2) \neq PFLAG$, go to 14

End of flight print.

14.  IT = | STAPE |

Check if printout to be off-line.

15.  If $IT = 0$, go to 18
     If $IT \neq 0$, go to 16

16.  WRITE    TIME(1), VSUBR, HATM, BETAD, RANGE, W(1),
              ERATIO   on Unit IT
```

```
Check if printout to be on-line also.

17.  If STAPE $> 0$, go to 19
     If STAPE $\leq 0$, go to 18

Print on-line.

18.  PRINT    TIME(1), VSUBR, HATM, BETAD, RANGE,
              W(1), ERATIO

Write extra print line if desired.

19.  CALL XPRINT(PRNTON, STAPE)

20.  Return

Check for discontinuity.

21.  If FLDSC $= 0$, go to 24
     If FLDSC $\neq 0$, go to 22

Check for discontinuity print option.

22.  If ((PFLAG)/(4))(2) $=$ (PFLAG)/(2),  go to 24
     If ((PFLAG)/(4))(2) $\neq$ (PFLAG)/(2),  go to 23

23.  Go to 27

Check for time to print.

24.  If FLTP not true,  return
     If FLTP true,  go to 25

Check for print-interval print option.

25.  If ((PFLAG)/(8))(2) $=$ (PFLAG)/(4),  return
     If ((PFLAG)/(8))(2) $\neq$ (PFLAG)/(4),  go to 26
```

```
26.  Go to 27

Print output.

27.  If $TIME \neq TI$, go to 34

     If $TIME = TI$, go to 28

Check if printout to be on-line only.

28.  If $TAPE = 0$, go to 32

     If $TAPE \neq 0$, go to 29

Write output off-line.

29.  WRITE headings on Unit TAPE

Check if output to be on-line also.

30.  If $OTAPE < 0$, go to 32

     If $OTAPE \geq 0$, go to 31

31.  Go to 34

Write headings on-line.

32.  PRINT headings on the on-line printer

33.  Go to 38

Time history print.

Check if output to be off-line only.

34.  If $TAPE = 0$, go to 38

     If $TAPE \neq 0$, go to 35

Write output off-line.
```

35. WRITE TIME(1), STIND, TYSC, VSUBR, HATM, BETAD, AZD, RANGE, W(1), VSUBI, THRST, ACCEL, LAT, LONG, WDOT, MACH, QATM, CSUBL, CSUBD, LIFT, DRAG, ALPHAD, SREF, CATM, RATM, PATM, TATM, PITD, PRATED, YAWD, YRATED, GRAV, WPL, TFLIT, TSTAG, TSECT, ERATIO, RXIN, RYIN, RZIN, XDIN(1) YDIN(1), ZDIN(1), THRSTX, THRSTY, THRSTZ, GRAVX, GRAVY, GRAVZ, DRAGX, DRAGY, DRAGZ, LIFTX, LIFTY, LIFTZ on Unit TAPE

Check if output to be on-line also.

36. If OTAPE < 0, go to 38

If OTAPE ≥ 0, go to 37

37. Go to 39

Print output on-line.

38. PRINT TIME(1), STIND, TYSC, VSUBR, HATM, BETAD, AZD, RANGE, W(1), VSUBI, THRST, ACCEL, LAT, LONG, WDOT, MACH, QATM, CSUBL, CSUBD, LIFT, DRAG, ALPHAD, SREF, CATM, RATM, PATM, TATM, PITD, PRATED, YAWD, YRATED, GRAV, WPL, TFLIT, TSTAG, TSECT, ERATIO, RXIN, RYIN, RZIN, XDIN(1), YDIN(1), ZDIN(1), THRSTX, THRSTY, THRSTZ, GRAVX, GRAVY, GRAVZ, DRAGX, DRAGY, DRAGZ, LIFTX, LIFTY, LIFTZ on the on-line printer

```
Write extra print line if desired.

39.  CALL XPRINT(PRNTOF, OTAPE)

Check for engine printout.

40.  If THIND < 4, go to 47

     If THIND ≥ 4, go to 41

Check if printout on-line only.

41.  If TAPE = 0, go to 45

     If TAPE ≠ 0, go to 42

Write output off-line.

42.  WRITE   TO, RHOO, TF, RHOF, PTO, PTF, WO(1),
             WF(1), VO, VF, PO, PF, WOCDT, WFCDT,
             PC, MRC, CSUBF, FCHMBR, WOEDT,
             WFEDT, WEDOT, MRE, ISPE, FENGIN,
             WOSDT, WFSDT, WSDOT, MRS, ISPS, FS
             on Unit TAPE

Check if printout on-line also.

43.  If OTAPE < 0, go to 45

     If OTAPE ≥ 0, go to 44

44.  Go to 47

Print on-line.
```

```
45.  PRINT   TO, RHOO, TF, RHOF, PTO, PTF, WO(1),
             WF(1), VO, VF, PO, PF, WOCDT, WFCDT,
             PC, MRC, CSUBF, FCHMBR, WOEDT,
             WFEDT, WEDOT, MRE, ISPE, FENGIN,
             WOSDT, WFSDT, WSDOT, MRS, ISPS, FS
             on the on-line printer

46.  Go to 47

Check end-of-section flag.

47.  If FLEOS not true, return

     If FLEOS true, go to 48

Check if section heading on-line only.

48.  If TAPE = 0, go to 52

     If TAPE ≠ 0, go to 49

Write off-line.

49.  WRITE  section heading on Unit TAPE

Check if on-line also.

50.  If OTAPE < 0, go to 52

     If OTAPE ≥ 0, go to 51

51.  Return

52.  PRINT  section heading on the on-line printer.

53.  Return
```

SUBROUTINE OUTPUT NOMENCLATURE

Symbol	Description	Units	Reference
ACCEL	Acceleration	—	/GILL/, 35, 38
ALPHAD	Angle of attack	degrees	/DATA/, 35, 38
AZD	Azimuth	degrees	2, 35, 38
AZ	Azimuth	radians	/DATA/, 2
BETAD	Angle between local vertical and velocity vector	degrees	1, 16, 18, 35, 38
BETA	Angle between local vertical and velocity vector	radians	/DATA/, 1
CATM	Speed of sound	ft/sec	/DATA/, 35, 38
CSUBD	Drag coefficient	—	/DATA/, 35, 38
CSUBF	Thrust coefficient	—	/ENGINO/, 42, 45
CSUBL	Lift coefficient	—	/DATA/, 35, 38
DPRAD	Conversion factor	$\frac{\text{degrees}}{\text{radians}}$	/CONST/, 1, 2, 3, 4, 5, 6
DRAG	Drag	lbs	/DATA/, 35, 38
DRAGX	X-component of drag	lbs	/COMP/, 35, 38
DRAGY	Y-component of drag	lbs	/COMP/, 35, 38
DRAGZ	Z-component of drag	lbs	/COMP/, 35, 38
ERATIO	Ratio of kinetic to potential energy	—	/DATA/, 16, 18, 35, 38
FCHMBR	Thrust chamber thrust	lbs	/ENGINO/, 42, 45

SUBROUTINE OUTPUT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
FENGIN	Engine thrust	lbs	/ENGINO/, 42, 45
FLDSC	Discontinuity flag	—	INTEGER, /WFLAG/, 21
FLEOF	End-of-flight flag	—	LOGICAL, /WFLAG/, 12
FLEOS	End-of-section flag	—	LOGICAL, /WFLAG/, 47
FLPRT	Print flag	—	INTEGER, /INPUT/, 10, 11
FLTP	Time-to-print flag	—	LOGICAL, /WFLAG/, 24
FLTRW	Type-of-run working flag	—	INTEGER, /WFLAG/, 11
FS	Total stage thrust	lbs	/ENGINO/, 42, 45
GRAV	Gravity	ft/sec^2	/DATA/, 35, 38
GRAVX	X-component of gravity	ft/sec^2	/COMP/, 35, 38
GRAVY	Y-component of gravity	ft/sec^2	/COMP/, 35, 38
GRAVZ	Z-component of gravity	ft/sec^2	/COMP/, 35, 38
HATM	Height above reference spheroid	ft	/DATA/, 16, 18, 35, 38
ISPE	Engine specific impulse	sec	REAL, /ENGINO/, 42, 45
ISPS	Stage specific impulse	sec	REAL, /ENGINO/, 42, 45
IT	Output scratch tape unit	—	14, 15, 16
LAT	Geodetic latitude	degrees	REAL, /DATA/, 35, 38

SUBROUTINE OUTPUT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
LIFT	Lift	lbs	REAL, /DATA/, 35, 38
LIFTX	X-component of lift	lbs	REAL, /COMP/, 35, 38
LIFTY	Y-component of lift	lbs	REAL, /COMP/, 35, 38
LIFTZ	Z-component of lift	lbs	REAL, /COMP/, 35, 38
LONG	West longitude	degrees	REAL, /DATA/, 35, 38
MACH	Mach number	—	REAL, /DATA/, 35, 38
MRC	Thrust chamber mixture ratio	—	REAL, /ENGINO/, 42, 45
MRE	Engine mixture ratio	—	REAL, /ENGINO/, 42, 45
MRS	Stage mixture ratio	—	REAL, /ENGINO/, 42, 45
OTAPE	Output tape assignment	—	INTEGER, /TAPES/, 7, 30, 36, 39, 43, 50
PATM	Atmospheric pressure	psia	/DATA/, 35, 38
PC	Thrust chamber nozzle stagnation pressure	psia	/ENGINO/, 42, 45
PFLAG	Print flag	—	INTEGER, 10, 11, 13, 22, 25
PF	Fuel pump inlet total pressure	psia	/ENGINO/, 42, 45
PITD	Pitch angle	degrees	3, 35, 38

SUBROUTINE OUTPUT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PIT	Pitch angle	radians	/DATA/, 3
PO	Oxidizer pump inlet total pressure	psia	/ENGINO/, 42, 45
PRATED	Pitch rate	$\frac{\text{degrees}}{\text{sec}}$	4, 35, 38
PRATE	Pitch rate	$\frac{\text{radians}}{\text{sec}}$	/DATA/, 4
PRNTOF	Array of parameters to be output on optional print line	—	/IPRINT/, 39
PRNTON	Array of parameters to be output on optional print line	—	/IPRINT/, 19
PTF	Fuel tank top pressure	psia	/ENGINO/, 42, 45
PTO	Oxidizer tank top pressure	psia	/ENGINO/, 42, 45
QATM	Dynamic pressure	lb/ft^2	/DATA/, 35, 38
RANGE	Instantaneous surface range	n.m.	/DATA/, 16, 18, 35, 38
RATM	Atmospheric density	$\frac{\text{slugs}}{ft^3}$	/DATA/, 35, 38
RHOF	Fuel tank propellant density	lb/ft^3	/ENGINO/, 42, 45
RHOO	Oxidizer tank propellant density	lb/ft^3	/ENGINO/, 42, 45
RXIN	X-component of inertial position vector	ft	/GILL/, 35, 38
RYIN	Y-component of inertial position vector	ft	/GILL/, 35, 38

SUBROUTINE OUTPUT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RZIN	Z-component of inertial position vector	ft	/GILL/, 35, 38
SREF	Stage reference area	ft^2	/WORKD/, 35, 38
STAPE	Output scratch tape assignment	—	INTEGER, /TAPES/, 14, 17, 19
STIND	Stage indicator	—	INTEGER, /WORKD/, 35, 38
TAPE	Output tape unit	—	INTEGER, 7, 28, 34, 35, 41, 42, 48, 49
TATM	Atmospheric temperature	^oF	/DATA/, 35, 38
TFLIT	Flight time (instantaneous time minus initial time)	sec	/DATA/, 35, 38
TF	Fuel tank propellant temperature	^oR	/ENGINO/, 42, 45
THIND	Thrust option indicator	—	INTEGER, /WORKD/, 40
THRST	Vehicle thrust	lb	/DATA/, 35, 38
THRSTX	X-component of vehicle thrust	lb	/COMP/, 35, 38
THRSTY	Y-component of vehicle thrust	lb	/COMP/, 35, 38
THRSTZ	Z-component of vehicle thrust	lb	/COMP/, 35, 38
TIME	Range time	sec	/GILL/, 16, 18, 27, 35, 38

SUBROUTINE OUTPUT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TI	Initial time	sec	/INPUT/, 27
TO	Oxidizer tank propellant temperature	$^{\circ}R$	/ENGINO/, 42, 45
TSECT	Section time	sec	/DATA/, 35, 38
TSTAG	Stage time	sec	/DATA/, 35, 38
TYSC	Section tally	—	INTEGER, /WFLAG/, 35, 38
VF	Volume of fuel propellant in tank	ft^3	/ENGINO/, 42, 45
VO	Volume of oxidizer propellant in tank	ft^3	/ENGINO/, 42, 45
VSUBI	Inertial velocity	ft/sec	/DATA/, 35, 38
VSUBR	Relative velocity	ft/sec	/DATA/, 16, 18, 35, 38
WDOT	Missile flow rate	lb/sec	/GILL/, 35, 38
WEDOT	Engine oxidizer flow rate	lb/sec	/ENGINO/, 42, 45
WFCDT	Thrust chamber fuel flow rate	lb/sec	/ENGINO/, 42, 45
WFEDT	Engine fuel flow rate	lb/sec	/ENGINO/, 42, 45
WF	Fuel propellant weight	lbs	/GILL/, 42, 45
WFSDT	Stage fuel flow rate	lb/sec	/ENGINO/, 42, 45

SUBROUTINE OUTPUT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WOCDT	Thrust chamber oxidizer flow rate	lb/sec	/ENGINO/, 42, 45
WOEDT	Engine oxidizer flow rate	lb/sec	/ENGINO/, 42, 45
WO	Oxidizer propellant weight	lb	/GILL/, 42, 45
WOSDT	Stage oxidizer flow rate	lb/sec	/ENGINO/, 42, 45
WPL	Payload weight	lb	/INPUT/, 35, 38
W	Missile weight	lb	/GILL/, 16, 18, 35, 38
WSDOT	Stage total flow rate	lb/sec	/ENGINO/, 42, 45
XDIN	X-component of inertial velocity vector	ft/sec	/GILL/, 35, 38
YAWD	Yaw angle	degrees	5, 35, 38
YAW	Yaw angle	radians	/DATA/, 5
YDIN	Y-component of inertial velocity vector	ft/sec	/GILL/, 35, 38
YRATED	Yaw rate	$\frac{\text{degrees}}{\text{sec}}$	6, 35, 38
YRATE	Yaw rate	$\frac{\text{radians}}{\text{sec}}$	/DATA/, 6
ZDIN	Z-component of inertial velocity vector	ft/sec	/GILL/, 35, 38
Z	Error communication flag	—	8, 9

```
$IBFTC ØUTPUT  FULIST,REF
      SUBRØUTINE ØUTPUT
      CØMMØN/TAPES/ITAPE,ØTAPE,STAPE
      INTEGER  ØTAPE,STAPE
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $              TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL  INFIN
      CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETM,
     $              DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $              SECTS(20,12),STAGS(20,6),HEAD(11,15)
      REAL  LATI,LØNGI
      INTEGER  TARGET
      INTEGER  FLTR,FLSC,FLPRT
      CØMMØN /IPRINT/ PRNTØN(6),PRNTØF(6)
      CØMMØN/WFLAG/FLEØF,FLEØS,FLTP,FLDSC,TYSC,FLTRW
      LØGICAL FLEØF,FLEØS,FLTP
      INTEGER  FLDSC,TYSC,FLTRW
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSES,DTSBP,
     $              PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $              PADSK,VESK,PADRT1,PADRT2,
     $            SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $             ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL  ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $            W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $            DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $            WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $              PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $              GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $              QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $              CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $              ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN/CØMP/THRSTX,THRSTY,THRSTZ,GRAVX,GRAVY,GRAVZ,
     $              DRAGX,DRAGY,DRAGZ,LIFTX,LIFTY,LIFTZ,
     $              VRX,VRY,VRZ
      REAL  LIFTX,LIFTY,LIFTZ
      CØMMØN /ENGINØ/ PØ,PF,TØ,TF,CSTR,RHØØ,RHØF,PTØ,PTF,VØ,VF,HØ,HF,
     $            WØCDT,WFCDT,MRC,PC,WØGDT,WFGDT,CVTH,CFTH,CSUBF,FCHMBR,
     $            FENGIN,WØEDT,WFEDT,MRE,WEDØT,ISPE,WØPDT,WFPDT,
     $            FS,WØSDT,WFSDT,MRS,WSDØT,ISPS
      REAL  MRC,MRE,MRS,ISPE,ISPS
C          ØUTPUT RØUTINE
      INTEGER  TAPE
      INTEGER  PFLAG
      BETAD = BETA * DPRAD
      AZD = AZ * DPRAD
      PITD = PIT * DPRAD
```

FIGURE 55. SYMBOLIC LISTING OF SUBROUTINE OUTPUT

```
      PRATED = PRATE * DPRAD
      YAWD = YAW * DPRAD
      YRATED = YRATE * DPRAD
      TAPE = IABS(ØTAPE)
      CALL CHKERR(Z)
      IF (Z .NE. 0.0)  GØ TØ 510
      PFLAG = FLPRT
      IF (FLTRW .NE. 0)  PFLAG = FLPRT/8
C             CHECK FØR END ØF FLIGHT
      IF (.NØT. FLEØF)  GØ TØ 100
      IF ((PFLAG/2)*2 .EQ. PFLAG)  RETURN
C             END ØF FLIGHT PRINT
      IT = IABS(STAPE)
      IF (IT .EQ. 0)  GØ TØ 50
      WRITE (IT,1) TIME(1),VSUBR,HATM,BETAD,RANGE,W(1),ERATIØ
      IF (STAPE .GT. 0)  GØ TØ 51
   50 PRINT 1, TIME(1),VSUBR,HATM,BETAD,RANGE,W(1),ERATIØ
   51 CALL XPRINT(PRNTØN,STAPE)
      RETURN
    1 FØRMAT (1H ,F15.2,2F15.0,2F15.2,2F15.3)
C             CHECK FØR DISCØNTINUITY
  100 IF (FLDSC .EQ. 0)  GØ TØ 200
      IF ((PFLAG/4)*2 .EQ. PFLAG/2)  GØ TØ 200
      GØ TØ 500
C             CHECK FØR TIME TØ PRINT
  200 IF (.NØT. FLTP)  RETURN
      IF ((PFLAG/8)*2 .EQ. PFLAG/4)  RETURN
      GØ TØ 500
C             PRINT ØUTPUT
  500 IF (TIME .NE. TI)  GØ TØ 510
      IF (TAPE .EQ. 0)  GØ TØ 505
      WRITE (TAPE,2)  (HEAD(I,13),I=1,11)
      WRITE (TAPE,3)  (HEAD(I,14),I=1,11)
      WRITE (TAPE,3)  (HEAD(I,15),I=1,11)
      IF (ØTAPE .LT. 0)  GØ TØ 505
      GØ TØ 510
  505 PRINT 2,  (HEAD(I,13),I=1,11)
      PRINT 3,  (HEAD(I,14),I=1,11)
      PRINT 3,  (HEAD(I,15),I=1,11)
      GØ TØ 515
    2 FØRMAT(1H1,5X,11A6)
    3 FØRMAT(1H ,5X,11A6)
C             TIME HISTØRY PRINT
  510 IF (TAPE .EQ. 0)  GØ TØ 515
      WRITE (TAPE,5)  TIME(1),STIND,TYSC,
     $            VSUBR,HATM,BETAD,AZD,RANGE,W(1),
     $            VSUBI,THRST,ACCEL,LAT,LØNG,WDØT,
     $            MACH,QATM,CSUBL,CSUBD,LIFT,DRAG,
     $            ALPHAD,SREF,CATM,RATM,PATM,TATM,
     $            PITD,PRATED,YAWD,YRATED,GRAV,WPL,
```

FIGURE 55. SYMBOLIC LISTING OF SUBROUTINE OUTPUT (cont.)

```
     $             TFLIT,TSTAG,TSECT,ERATIØ,
     $             RXIN,RYIN,RZIN,XDIN(1),YDIN(1),ZDIN(1),
     $             THRSTX,THRSTY,THRSTZ,GRAVX,GRAVY,GRAVZ,
     $             DRAGX,DRAGY,DRAGZ,LIFTX,LIFTY,LIFTZ
      IF (ØTAPE .LT. 0)  GØ TØ 515
      GØ TØ 520
  515 PRINT 5,  TIME(1),STIND,TYSC,
     $             VSUBR,HATM,BETAD,AZD,RANGE,W(1),
     $             VSUBI,THRST,ACCEL,LAT,LØNG,WDØT,
     $             MACH,QATM,CSUBL,CSUBD,LIFT,DRAG,
     $             ALPHAD,SREF,CATM,RATM,PATM,TATM,
     $             PITD,PRATED,YAWD,YRATED,GRAV,WPL,
     $             TFLIT,TSTAG,TSECT,ERATIØ,
     $             RXIN,RYIN,RZIN,XDIN(1),YDIN(1),ZDIN(1),
     $             THRSTX,THRSTY,THRSTZ,GRAVX,GRAVY,GRAVZ,
     $             DRAGX,DRAGY,DRAGZ,LIFTX,LIFTY,LIFTZ
    5 FØRMAT(1H ,/1H ,12X,6HTIME =,F12.6,6X,7HSTAGE =,I2,6X,9HSECTIØN =,
     $        I2/1H ,/6H     1,6F16.3/6H     2,6F16.3/
     $        6H     3,4F16.6,2F16.3/6H     4,F16.5,F16.3,4F16.8/
     $        6H     5,4F16.6,2F16.3/6H     6,4F16.6//
     $        6H     7,6E16.8/6H     8,6E16.8/6H     9,6E16.8)
  520 CALL XPRINT(PRNTØF,ØTAPE)
  530 IF (THIND .LT. 4)  GØ TØ 900
      IF (TAPE .EQ. 0)  GØ TØ 535
      WRITE (TAPE,6)  TØ,RHØØ,TF,RHØF,PTØ,PTF,
     $             WØ(1),WF(1),VØ,VF,PØ,PF,
     $             WØCDT,WFCDT,PC,MRC,CSUBF,FCHMBR,
     $             WØEDT,WFEDT,WEDØT,MRE,ISPE,FENGIN,
     $             WØSDT,WFSDT,WSDØT,MRS,ISPS,FS
      IF (ØTAPE .LT. 0)  GØ TØ 535
      GØ TØ 900
  535 PRINT 6,  TØ,RHØØ,TF,RHØF,PTØ,PTF,
     $             WØ(1),WF(1),VØ,VF,PØ,PF,
     $             WØCDT,WFCDT,PC,MRC,CSUBF,FCHMBR,
     $             WØEDT,WFEDT,WEDØT,MRE,ISPE,FENGIN,
     $             WØSDT,WFSDT,WSDØT,MRS,ISPS,FS
      GØ TØ 900
    6 FØRMAT (1H ,/6H    10,6F16.6/6H    11,6F16.3/6H    12,3F16.3,
     $        F16.4,F16.5,F16.3/6H    13,3F16.3,F16.4,2F16.3/
     $        6H    14,3F16.3,F16.4,2F16.3//)
C
  900 IF (.NØT. FLEØS)  RETURN
      IF (TAPE .EQ. 0)  GØ TØ 905
      WRITE (TAPE,4)  (HEAD(I,TYSC),I=1,11)
      IF (ØTAPE .LT. 0)  GØ TØ 905
      RETURN
  905 PRINT 4,  (HEAD(I,TYSC), I=1,11)
      RETURN
    4 FØRMAT (1H ,/1H ,5X,11A6)
      END
```

FIGURE 55. SYMBOLIC LISTING OF SUBROUTINE OUTPUT (cont.)

XLII. SUBROUTINE XPRINT

This subroutine writes a line of output, on any tape, consisting of up to 6 parameters specified by program input, together with their input names.

CALLING SEQUENCE

The calling sequence is:

CALL XPRINT (LIST, TAPE)

LIST is an array of dimension 6 containing data read in via the FINPUT A-prefix option.

SOLUTION METHOD

```
1.  If LIST(1) = 0,  return

    If LIST(1) ≠ 0,  go to 2

Look up items to be printed:

2.  Do 12, J = 1, 6

3.  If LIST(J) = 0,  go to 13

    If LIST(J) ≠ 0,  go to 4
```

```
 4.  K = J

 5.  CALL UNPACK (LIST(J), ARRAY)

 6.  NLOC = MASKA (LIST(J))

 Get Hollerith name of parameter:

 7.  CALL LOOKUP (NLOC, NAMES(1, J))

 Calculate parameter array subscript:

 8.  NAMES(2, J) = ADD - NLOC + 1

 Execute routine to calculate parameter, if any:

 9.  CALL XEC(DEC)

 Find conversion factor for parameter:

10.  CONV = TAGS(TAG)

11.  If TAG = 0, CONV = 1.0

     If TAG ≠ 0, CONV = CONV

12.  VAR(J) = (CLA(ADD))(CONV)

 Print items:

13.  IT = | TAPE |

14.  If IT = 0, go to 17

     If IT ≠ 0, go to 15
```

```
15.  Write K names and K variables on unit IT:
16.  If TAPE $>$ 0,  go to 18
     If TAPE $\leq$ 0,  go to 17
17.  Print K names and K variables on the on-line printer
18.  Return
```

SUBROUTINE XPRINT NOMENCLATURE

Symbol	Description	Units	Reference
ADD	Address portion of LIST(J)	—	INTEGER, EQUIV, 8, 12
ARRAY	Array containing address, tag, decrement and prefix of LIST(J)	—	DIM, EQUIV, 5
CONV	Conversion factor for variable	—	10, 11, 12
DEC	Decrement portion of LIST(J)	—	EQUIV, 9
IT	Tape unit to write on	—	13, 14, 15
J	Do loop counter	—	2, 3, 4, 5, 6, 7, 8, 12
K	Number of items to print	—	4, 15, 17
LIST	Array specifying parameters to be output	—	CALL, DIM, 1, 3, 5, 6
NAMES	Hollelrith names and subscripts of parameters to be output	—	DIM, 7, 8, 15, 17
NLOC	Base location of parameter for table lookup	—	6, 7, 8
TAG	Tag portion of LIST(J)	—	INTEGER, EQUIV, 10, 11
TAGS	Array of conversion factors	—	/CONST/, 10
TAPE	Output unit(s) to use	—	CALL, INTEGER, 13, 16
VAR	Array of parameter values for output	—	DIM, 12, 15, 17

```
$IBFTC XPRINT
      SUBRØUTINE XPRINT(LIST,TAPE)
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $             TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL  INFIN
C        ØPTIØNAL PRINT LINE ØF UP TØ 6 PARAMETERS
      DIMENSIØN  LIST(6),NAMES(2,6),VAR(6)
      DIMENSIØN  ARRAY(4)
      EQUIVALENCE  (ARRAY(1),ADD),(ARRAY(2),TAG),(ARRAY(3),DEC),
     $             (ARRAY(4),PRE)
      INTEGER  ADD,TAG,TAPE
C
      IF (LIST(1) .EQ. 0)  RETURN
      DØ 10  J=1,6
      IF (LIST(J) .EQ. 0)  GØ TØ 20
      K = J
      CALL UNPACK(LIST(J),ARRAY)
      NLØC = MASKA(LIST(J))
      CALL LØØKUP(NLØC,NAMES(1,J))
      NAMES(2,J) = ADD - NLØC + 1
      CALL XEC(DEC)
      CØNV = TAGS(TAG)
      IF (TAG .EQ. 0)  CØNV = 1.0
      VAR(J) = CLA(ADD) * CØNV
   10 CØNTINUE
C
   20 IT = IABS(TAPE)
      IF (IT .EQ. 0)  GØ TØ 25
      WRITE (IT,1)  ((NAMES(1,I),NAMES(2,I)),I=1,K)
      WRITE (IT,2)  (VAR(I),I=1,K)
      IF (TAPE .GT. 0)  GØ TØ 30
   25 PRINT 1,   ((NAMES(1,I),NAMES(2,I)),I=1,K)
      PRINT 2,   (VAR(I),I=1,K)
    1 FØRMAT(1H ,/1H ,5X,6(4X,A6,1H(,I4,1H)))
    2 FØRMAT(1H ,5X,6(E16.8))
   30 RETURN
      END
```

FIGURE 56. SYMBOLIC LISTING OF SUBROUTINE XPRINT

XLIII. SUBROUTINE PLOAD

This subroutine provides the capability to optimize the propellant loading for each of as many stages as desired. Optimization of propellant loading implies minimization of residual propellants. Optimum propellant loading may be achieved with constant or variable missile weight. In either case, minimum ullage requirements are always maintained.

CALLING SEQUENCE

The calling sequence is:

CALL PLOAD (FUNCG)

This subroutine uses COMMON blocks TAPES, CONST, INPUT, SAVE, ENGINI and PROPL.

SOLUTION METHOD

Set maximum iteration limit.

1. MXIT1 = IPROPD(1)

Set maximum consecutive error limit.

2. MXIT2 = IPROPD(2)

Set option indicator. If OPTION = 0, vehicle weight will vary. If OPTION $\neq$ 0, payload weight will vary and vehicle weight will remain constant.

3. OPTION = PROPD(3)

Set number of stages for weight buildup.

4. NSTAGE = $|$IPROPD(4)$|$

Set output tape unit.

5. IT = $|$STAPE$|$

Clear Storage Block

6. Do 7 I = 1, 78

7. WARRAY(I) = 0

Set the constant $\sqrt{2}$.

8. SQRT2 = $(2)^{.5}$

Maximum Propellant Weight Computations

9. Do 58 I = 1, NSTAGE

Determine tank pressures.

10. If TANKS(4, I) $\neq$ 0, go to 14

If TANKS(4, I) = 0, go to 11

```
11.  PTO = TANKS(5, I) + PSL
12.  PTF = TANKS(6, I) + PSL
13.  Go to 16
14.  PTO = TANKS(8, I)
15.  PTF = TANKS(13, I)
```

Determine propellant densities.

```
16.  CALL PRPDEN (1, PTO, MISC(21, I) - 459.688,
                  Z, Z, PVO, RHOO)
17.  CALL PRPDEN (ITANKS(1, I) + 1, PTF, MISC(23, I)
     -459.688, TANKS(2, I), TANKS(3, I), PVF, RHOF)
```

Set minimum ullage volumes.

```
18.  VOULL = PROPD((4)(I) + 3)
19.  VFULL = PROPD((4)(I) + 4)
```

Set oxidizer tank geometry.

```
20.  X = TANKS(19, I)
21.  R = TANKS(20, I)
22.  B = R
```

Compute volume of oxidizer tank upper header.

23. If MOD(IFIX(TANKS(21, I) + .5), 2) = 0, B = (R)/SQRT2)

Otherwise, B = B

MOD and IFIX are FORTRAN functions used for remaindering and converting from real to integer, respectively.

24. VUH = $(2)(PI)(B)(R^2)/(3)$

25. If TANKS(21, I) $<$ 2.5, go to 28

If TANKS(21, I) $\geq$ 2.5, go to 26

26. VUH = $(PI)(B)(R^2)$ - VUH

27. X = X - B

Compute volume of oxidizer tank lower header.

28. B = R

29. If MOD(IFIX(TANKS(22, I) + .5), 2) = 0, B = (R)/(SQRT2)

Otherwise, B = B

30. VLH = $(2)(PI)(B)(R^2)/(3)$

31. If TANKS(22, I) $<$ 2.5, go to 34

If TANKS(22, I) $\geq$ 2.5, go to 32

32. VLH = $(PI)(B)(R^2)$ - VLH

Set length of cylindrical portion of oxidizer tank.

33. X = X - B

34. Continue

Compute maximum weight of propellant in oxidizer tank.

35. WOTNK(I) = ((X)(PI)(R^2) + VUH + VLH - VOULL)(RHOO)

Compute weight of oxidizer below bulkhead.

36. WOBB(I) = (MISC(25, I))(RHOO)

Compute total stage oxidizer weight.

37. WOTOT(I) = WOTNK(I) + WOBB(I)

Compute oxidizer residual weight convergence tolerance.

38. WOEPSI(I) = (MISC(8, I))(RHOO)

Set fuel tank geometry.

39. X = TANKS(23, I)

40. R = TANKS(24, I)

41. B = R

Compute volume of fuel tank upper header.

42. If MOD(IFIX(TANKS(25, I) + .5), 2) = 0, B = (R)/(SQRT2)

Otherwise, B = B

43. VUH = (2)(PI)(B)(R^2)/(3)

44. If TANKS(25, I) $<$ 2.5, go to 47

If TANKS(25, I) $\geq$ 2.5, go to 45

45. VUH = (PI)(B)(R^2) - VUH

46. X = X - B

Compute volume of fuel tank lower header.

47. B = R

48. If MOD (IFIX(TANKS(26, I) + .5), 2) = 0, B = (R)/(SQRT2)

Otherwise, B = B

49. VLH = (2)(PI)(B)(R^2)/(3)

50. If TANKS(26, I) $<$ 2.5, go to 53

If TANKS(26, I) $\geq$ 2.5, go to 51

51. VLH = (PI)(B)(R^2) - VLH

Set length of cylindrical portion of fuel tank.

52. X = X - B

53. Continue

Compute maximum weight of propellant in fuel tank.

54. WFTNK(I) = ((X)(PI)(R^2) + VUH + VLH - VFULL)(RHOF)

Compute weight of fuel below bulkhead.

```
55.  WFBB(I) = (MISC(26, I))(RHOF)
```

Compute total stage fuel weight.

```
56.  WFTOT(I) = WFTNK(I) + WFBB(I)
```

Compute fuel residual weight convergence tolerance.

```
57.  WFEPSI(I) = (MISC(8, I))(RHOF)
58.  Continue
```

Save Nominal Case Values

```
59.  Do 67 I = 1, NSTAGE
60.  WOIN(I) = STAGS(3, I)
61.  WFIN(I) = STAGS(4, I)
62.  If PROPD((4)(I) + 1) = 0, or PROPD(4) > 0, go to 65
     Otherwise, go to 63
63.  STAGS(3, I) = WOTOT(I)
64.  STAGS(4, I) = WFTOT(I)
65.  Continue
66.  WPRPN(I) = STAGS(3, I) + STAGS(4, I)
67.  Continue
68.  WPLN = WPL
```

```
69.  NN = 40

70.  Go to 164
```

Maximum Propellant Weights Print Out

```
71.  If IT = 0, go to 74
     If IT ≠ 0, go to 72

72.  Write K, WOTNK(K), WOBB(K), WOTOT(K), WFTNK(K),
          WFBB(K), WFTOT(K) on Unit IT

73.  If STAPE > 0, go to 75
     If STAPE ≤ 0, go to 74

74.  Write K, WOTNK(K), WOBB(K), WOTOT(K), WFTNK(K),
          WFBB(K), WFTOT(K) on the on-line-printer

75.  Continue
```

Minimize Residual Propellants

```
76.  Do 129 ITER1 = 1, MXIT1
```

Compute residual propellants.

```
77.  CALL FUNCG
```

Check if an error occurred during the generation of propellant residuals.

```
78.  CALL CHKERR(Z)
```

79. If Z $\neq$ 0, go to 136

If Z = 0, go to 80

Set error counter to zero.

80. ITER2 = 0

Write out propellant weight data.

81. If IT = 0, go to 84

If IT $\neq$ 0, go to 82

82. Write STAGS(3, K), WOR(K), STAGS(4, K), WFR(K), K on Unit IT

83. If STAPE $>$ 0, go to 85

If STAPE $\leq$ 0, go to 84

84. Write STAGS(3, K), WOR(K), STAGS(4, K), WFR(K), K on the on-line-printer

85. Continue

86. Do 127 I = 1, NSTAGE

87. J = NSTAGE - I + 1

Starting from top of vehicle, check if each stage is to be optimally loaded.

88. If PROPD((4)(J) + 1) = 0, go to 127

If PROPD((4)(J) + 1) $\neq$ 0, go to 89

Set loaded propellant weights.

89. WOIL(J) = STAGS(3, J)

90. WFIL(J) = STAGS(4, J)

Compare oxidizer residual with convergence tolerance.

91. If WOR(J) ≤ WOEPSI(J), go to 112

If WOR(J) > WOEPSI(J), go to 92

Compare fuel residual with convergence tolerance.

92. If WFR(J) ≤ WFEPSI(J), go to 103

If WFR(J) > WFEPSI(J), go to 93

Both Oxidizer and Fuel Residuals

Check if stage burnout criterion is propellant depletion.

93. If PROPD((4)(J) + 1) < 0, go to 96

If PROPD((4)(J) + 1) ≥ 0, go to 94

Set error indicator.

94. CALL SETERR(6HRESID)

95. Go to 142

Reduce loaded propellant weights by amount of residuals.

96. STAGS(3, J) = WOIL(J) - WOR(J)

97. STAGS(4, J) = WFIL(J) - WFR(J)

98. Go to 122

99. STAGS(3, J) = WOIL(J) - WOR(J)

100. Go to 122

101. STAGS(4, J) = WFIL(J) - WFR(J)

102. Go to 122

Only Oxidizer Residual

Check if stage burnout criterion is propellant depletion.

103. If PROPD((4)(J) + 1) $<$ 0, go to 99

If PROPD((4)(J) + 1) $\geq$ 0, go to 104

Compute required fuel propellant weight.

104. WF = WFIL(J) + (WOR(J))/(1.0 + AVMR(J))

Test if required fuel propellant weight is greater than allowable.

105. If WF $>$ WFTOT(J), go to 109

If WF $\leq$ WFTOT(J), go to 106

Set new propellant weights.

106. STAGS(3, J) = WPRPN(J) - WF

107. STAGS(4, J) = WF

108. Go to 122

Maximize fuel tank load and calculate oxidizer load.

109. STAGS(4, J) = WFTOT(J)

110. STAGS(3, J) = WOIL(J) - WOR(J) +

(WFTOT(J) - WFIL(J))(AVMR(J))

111. Go to 122

Compare fuel residual weight with convergence tolerance.

112. If WFR(J) ≤ WFEPSI(J), go to 127

If WFR(J) > WFEPSI(J), go to 113

Only Fuel Residual

Check if stage burnout criterion is propellant depletion.

113. If PROPD((4)(J) + 1) < 0, go to 101

If PROPD((4)(J) + 1) ≥ 0, go to 114

Compute required oxidizer propellant weight.

114. WO = WOIL(J) + (WFR(J))(AVMR(J))/(1 + AVMR(J))

Test if required oxidizer propellant weight is greater than allowable.

115. If WO > WOTOT(J), go to 119

If WO ≤ WOTOT(J), go to 116

Set new propellant weights.

116. STAGS(3, J) = WO

117. STAGS(4, J) = WPRPN(J) - WO

118. Go to 122

Maximize oxidizer tank load and calculate fuel load.

119. STAGS(3, J) = WOTOT(J)

120. STAGS(4, J) = WFIL(J) - WFR(J) + (WOTOT(J) - WOIL(J))/(AVMR(J))

121. Go to 122

Payload Weight Computation

122. If OPTION $\neq$ 0, WPL = WPL - (STAGS(3, J) - WOIL(J)) - (STAGS(4, J) - WFIL(J))

Otherwise, WPL = WPL

Test for negative payload.

123. If WPL $<$ 0, go to 159

If WPL $\geq$ 0, go to 124

124. NN = 410

125. Go to 164

126. Go to 129

127. Continue

128. Return

129. Continue

Maximum Iterations Error Print

130. If IT = 0, go to 133

If IT ≠ 0, go to 131

131. Write error message on Unit IT

132. If STAPE > 0, go to 134

If STAPE ≤ 0, go to 133

133. Write error message on the on-line-printer

Set error indicator.

134. CALL SETERR(5HPLOAD)

135. Go to 142

Error Control

136. If ITER1 > 1, go to 151

If ITER1 ≤ 1, go to 137

Write error message.

137. If IT = 0, go to 140

If IT ≠ 0, go to 138

138. Write error message on Unit IT

139. If STAPE $>$ 0, go to 141

If STAPE $\leq$ 0, go to 140

140. Write error message on the on-line-printer

Set error indicator.

141. CALL SETERR(5HPROPL)

142. Do 145 I = 1, NSTAGE

Reset nominal propellant weights.

143. STAGS (3, I) = WOIN(I)

144. STAGS (4, I) = WFIN(I)

145. Continue

Reset nominal payload.

146. WPL = WPLN

147. NN = 820

148. Go to 164

149. Continue

150. Return

Increase iteration counter.

151. ITER2 = ITER2 + 1

Test if maximum number of consecutive errors has been exceeded.

152. If ITER2 $\geq$ MXIT2, go to 130

If ITER2 $<$ MXIT2, go to 153

Re-evaluate propellant loads.

153. WOTRY = (WOIL(J) + STAGS(3, J))/(2)

154. WFTRY = (WFIL(J) + STAGS(4, J))/(2)

Calculate payload weight.

155. If OPTION $\neq$ 0, WPL = WPL - (WOTRY - STAGS(3, J))

- (WFTRY - STAGS(4, J))

Otherwise, WPL = WPL

Set loaded propellant weights.

156. STAGS(3, J) = WOTRY

157. STAGS(4, J) = WFTRY

158. Go to 122

Negative Payload Error Print

159. If IT = 0, go to 162

If IT $\neq$ 0, go to 160

160. Write error message on Unit IT

```
161.  If STAPE > 0, go to 141
      If STAPE ≤ 0, go to 162
162.  Write error message on the on-line-printer
163.  Go to 141
```

Weight Buildup Computations

Compute initial weight of each stage starting from top of vehicle.

```
164.  STAGS(2, NSTAGE) = STAGS(3, NSTAGE) +
      STAGS(4, NSTAGE) + PROPD((4)(NSTAGE) + 2)
165.  NSTOP = NSTAGE - 1
166.  Do 169 N = 1, NSTOP
167.  M1 = NSTAGE - N
168.  M2 = M1 + 1
169.  STAGS(2, M1) = STAGS(2, M2) + STAGS(3, M1) +
      STAGS(4, M1) + PROPD((4)(M1) + 2)
170.  If NN = 40, go to 71
      If NN = 410, go to 126
      If NN = 820, go to 149
```

SUBROUTINE PLOAD NOMENCLATURE

Symbol	Description	Units	Reference
AVMR	Average in-flight mixture ratio	—	/SAVE/, 104, 110, 114, 120
B	Height of elliptical or spherical portion of the tank	ft	22, 23, 24, 26, 27, 28, 29, 30, 32, 33, 41, 42, 43, 45, 46, 47, 48, 49, 51, 52
FUNCG	Function generator	—	CALL, 77
I	Counter	—	86, 87
IPROPD	Alternate name for PROPD	—	DIM, EQUIV, 1, 2, 4
ITANKS	Input array	—	DIM, EQUIV, 17
ITER1	Iteration counter	—	76, 136
ITER2	Consecutive error counter	—	80, 151, 152
IT	Tape unit for output	—	5, 71, 72, 81, 82, 130, 137, 159
J	Counter	—	87
M1	Counter for weight buildup	—	167, 168, 169
M2	Counter for weight buildup	—	168, 169
MISC	Input array of miscellaneous engine data	—	/ENGINI/, REAL, 16, 17, 36, 38, 55, 57
MXIT1	Maximum iterations	—	1, 76
MXIT2	Maximum consecurive errors	—	2, 152
NN	—	—	69, 124, 147, 170

SUBROUTINE PLOAD NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
N	Counter	—	166, 167
NSTAGE	Number of stages	—	4, 9, 59, 86, 87, 142, 164, 165, 167
NSTOP	—	—	165, 166
OPTION	Indicator for loading option 0 = vehicle weight varies ≠0 = vehicle weight constant	—	3, 122, 155
PI	3.14159265	—	/CONST/, 24, 26, 30, 32, 35, 43, 45, 49, 51, 54
PROPD	Propellant loading input array	—	/PROPL/, EQUIV, 3, 18, 19, 62, 88, 93, 103, 113, 164, 169
PSL	Sea level atmospheric pressure	psia	/CONST/, 11, 12
PTF	Fuel tank pressure	psia	12, 15, 17
PTO	Oxidizer tank pressure	psia	11, 14, 16
PVF	Fuel vapor pressure	psia	17
PVO	Oxidizer vapor pressure	psia	16
RHOF	Fuel density	lb/ft^3	17, 54, 55, 57
RHOO	Oxidizer density	lb/ft^3	16, 35, 36, 38
R	Radius of tank	ft	21, 22, 23, 24, 26, 28, 29, 30, 32, 35, 40, 41, 42, 43, 45, 47, 48, 49 51, 54

SUBROUTINE PLOAD NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
SQRT2	$(2)^{.5}$	—	8, 23, 29, 42, 48
STAGS	Stage data input array	—	/INPUT/, 60, 61, 63, 64, 66, 82, 84, 89, 90, 96, 97, 99, 101, 106, 107, 109, 110, 116, 117, 119, 120, 122, 143, 144, 153, 154, 155, 156, 157, 164, 169
STAPE	Summary tape for propellant loading data	—	/TAPES/, INTEGER, 5, 73, 83, 132, 139, 161
TANKS	Tank data input array	—	/ENGINI/, EQUIV, 10, 11, 12, 14, 15, 17, 20, 21, 23, 25, 29, 31, 39, 40, 42, 44, 48, 50
VFULL	Fuel ullage volume	ft^3	19, 54
VLH	Volume of lower header	ft^3	30, 32, 35, 49, 51, 54
VOULL	Oxidizer ullage volume	ft^3	18, 35
VUH	Volume of upper header	ft^3	24, 26, 35, 43, 45, 54
WARRAY	Dummy array name for clearing storage	—	DIM, EQUIV, 7
WFIL	Array of saved initial fuel weights	lb	DIM, EQUIV, 90, 97, 101, 104, 110, 120, 122, 154
WFIN	Array of nominal fuel weights	lb	DIM, EQUIV, 61, 144
WFBB	Weight of fuel below bulkhead	lb	DIM, EQUIV, 55, 56, 72, 74

SUBROUTINE PLOAD NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WFEPSI	Fuel weight convergence tolerance	lb	DIM, EQUIV, 57, 92, 112
WF	Fuel weight	lb	104, 105, 106, 107
WFR	Array of residual fuel weights	lb	/SAVE/, 82, 84, 92, 97, 101, 112, 114, 120
WFTNK	Maximum weight of fuel in tank	lb	DIM, EQUIV, 54, 56, 72, 74
WFTOT	Total maximum fuel weight	lb	DIM, EQUIV, 56, 64, 72, 74, 105, 109, 110
WFTRY	Trial fuel weight	lb	154, 155, 157
WOBB	Weight of oxidizer below bulkhead	lb	DIM, EQUIV, 36, 37, 72, 74
WOEPSI	Oxidizer weight convergence tolerance	lb	DIM, EQUIV, 38, 91
WOIL	Array of saved initial oxidizer weights	lb	DIM, EQUIV, 89, 96, 99, 110, 114, 120, 122, 153
WOIN	Array of nominal oxidizer weights	lb	DIM, EQUIV, 60, 143
WO	Oxidizer weight	lb	114, 115, 116, 117
WOR	Array of oxidizer residuals	lb	/SAVE/, 82, 84, 91, 96, 99, 104, 110
WOTNK	Maximum weight of oxidizer in tank	lb	DIM, EQUIV, 35, 37, 72, 74
WOTOT	Total maximum oxidizer weight	lb	DIM, EQUIV, 37, 63, 72, 74, 115, 119, 120

SUBROUTINE PLOAD NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WOTRY	Trial oxidizer weight	lb	153, 155, 156
WPL	Payload weight	lb	/INPUT/, 68, 122, 123, 146, 155
WPLN	Nominal payload weight	lb	68, 146
WPRPN	Nominal total propellant weight	lb	DIM, EQUIV, 66, 106, 117
X	Length of tank	ft	20, 27, 33, 35, 39, 46, 52, 54
Z	Dummy variable	—	16, 78, 79

```
$IBFTC PLØAD    FULIST,REF
      SUBRØUTINE PLØAD(FUNCG)
C        PRØPELLANT LØADING ØPTIMIZATIØN
      CØMMØN/TAPES/ITAPE,ØTAPE,STAPE
      INTEGER  ØTAPE,STAPE
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $             TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL  INFIN
      CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,
     $             DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $             SECTS(20,12),STAGS(20,6),HEAD(11,15)
      REAL  LATI,LØNGI
      INTEGER  TARGET
      INTEGER  FLTR,FLSC,FLPRT
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $             AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $             WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/ENGINI/ YBBAR(5,6),YBAR(5,6),XBAR(6,6),KPC(6,6),KWØC(6,6),
     $          KWFC(6,6),KWØG(6,6),KWFG(6,6),CTØX(6,6),CTFL(6,6),
     $          MISC(30,6),TANKS(30,6)
      REAL  KPC,KWØC,KWFC,KWØG,KWFG,MISC
      CØMMØN /PRØPL/ PRØPD(28)
      DIMENSIØN  ITANKS(30,6)
      EQUIVALENCE  (ITANKS(1,1),TANKS(1,1))
      DIMENSIØN  IPRØPD(28)
      EQUIVALENCE  (PRØPD(1),IPRØPD(1))
      DIMENSIØN  WØTNK(6),WFTNK(6),WØBB(6),WFBB(6),WØTØT(6),WFTØT(6),
     $             WØEPSI(6),WFEPSI(6),WØIN(6),WFIN(6),WPRPN(6),
     $             WØIL(6),WFIL(6)
      DIMENSIØN  WARRAY(78)
      EQUIVALENCE  (WARRAY(1),WØTNK(1)),(WARRAY(7),WFTNK(1)),
     $             (WARRAY(13),WØBB(1)),(WARRAY(19),WFBB(1)),
     $             (WARRAY(25),WØTØT(1)),(WARRAY(31),WFTØT(1)),
     $             (WARRAY(37),WØEPSI(1)),(WARRAY(43),WFEPSI(1)),
     $             (WARRAY(49),WØIN(1)),(WARRAY(55),WFIN(1)),
     $             (WARRAY(61),WPRPN(1)),(WARRAY(67),WØIL(1)),
     $             (WARRAY(73),WFIL(1))
C
      MXIT1 = IPRØPD(1)
      MXIT2 = IPRØPD(2)
      ØPTIØN = PRØPD(3)
      NSTAGE = IABS(IPRØPD(4))
      IT = IABS(STAPE)
C        CLEAR STØRAGE BLØCKS
      DØ 10  I=1,78
   10 WARRAY(I) = 0.0
      SQRT2 = SQRT(2.0)
C        CALCULATE MAX PRØPELLANT WEIGHTS
```

FIGURE 57.   SYMBOLIC LISTING OF SUBROUTINE PLOAD

```
      DØ 25  I=1,NSTAGE
      IF (TANKS(4,I) .NE. 0.0)  GØ TØ 12
      PTØ = TANKS(5,I) + PSL
      PTF = TANKS(6,I) + PSL
      GØ TØ 13
   12 PTØ = TANKS(8,I)
      PTF = TANKS(14,I)
   13 CALL PRPDEN(1,PTØ,MISC(21,I)-459.688,Z,Z,PVØ,RHØØ)
      CALL PRPDEN(ITANKS(1,I)+1,PTF,MISC(23,I)-459.688,TANKS(2,I),
     $         TANKS(3,I),PVF,RHØF)
      VØULL = PRØPD(4*I+3)
      VFULL = PRØPD(4*I+4)
C
      X = TANKS(19,I)
      R = TANKS(20,I)
      B = R
      IF (MØD(IFIX(TANKS(21,I)+.5),2) .EQ. 0)  B = R/SQRT2
      VUH = 2.0*PI * B * R**2/3.0
      IF (TANKS(21,I) .LT. 2.5)  GØ TØ 14
      VUH = PI * B * R**2 - VUH
      X = X - B
   14 B = R
      IF (MØD(IFIX(TANKS(22,I)+.5),2) .EQ. 0)  B = R/SQRT2
      VLH = 2.0*PI * B * R**2/3.0
      IF (TANKS(22,I) .LT. 2.5)  GØ TØ 15
      VLH = PI * B * R**2 - VLH
      X = X - B
   15 CØNTINUE
      WØTNK(I) = (X * PI * R**2 + VUH + VLH - VØULL) * RHØØ
      WØBB(I) = MISC(25,I) * RHØØ
      WØTØT(I) = WØTNK(I) + WØBB(I)
      WØEPSI(I) = MISC(8,I) * RHØØ
C
      X = TANKS(23,I)
      R = TANKS(24,I)
      B = R
      IF (MØD(IFIX(TANKS(25,I)+.5),2) .EQ. 0)  B = R/SQRT2
      VUH = 2.0*PI * B * R**2/3.0
      IF (TANKS(25,I) .LT. 2.5)  GØ TØ 19
      VUH = PI * B * R**2 - VUH
      X = X - B
   19 B = R
      IF (MØD(IFIX(TANKS(26,I)+.5),2) .EQ. 0)  B = R/SQRT2
      VLH = 2.0*PI * B * R**2/3.0
      IF (TANKS(26,I) .LT. 2.5)  GØ TØ 20
      VLH = PI * B * R**2 - VLH
      X = X - B
   20 CØNTINUE
      WFTNK(I) = (X * PI * R**2 + VUH + VLH - VFULL) * RHØF
      WFBB(I) = MISC(26,I) * RHØF
```

FIGURE 57. SYMBOLIC LISTING OF SUBROUTINE PLOAD (cont.)

```
      WFTØT(I) = WFTNK(I) + WFBB(I)
      WFEPSI(I) = MISC(8,I) * RHØF
C
   25 CØNTINUE
C        SET UP AND SAVE NØMINAL CASE VALUES
      DØ 30  I=1,NSTAGE
      WØIN(I) = STAGS(3,I)
      WFIN(I) = STAGS(4,I)
      IF (PRØPD(4*I+1) .EQ. 0.0 .ØR. PRØPD(4) .GT. 0.0)  GØ TØ 29
      STAGS(3,I) = WØTØT(I)
      STAGS(4,I) = WFTØT(I)
   29 CØNTINUE
      WPRPN(I) = STAGS(3,I) + STAGS(4,I)
   30 CØNTINUE
      WPLN = WPL
C        DØ WEIGHT BUILD-UP FØR NØMINAL VALUES
      ASSIGN 40 TØ NN
      GØ TØ 900
C          PRINT ØUT MAXIMUM PRØPELLANT WEIGHTS
   40 IF (IT .EQ. 0)  GØ TØ 45
      WRITE (IT,2)   ((K,WØTNK(K),WØBB(K),WØTØT(K),WFTNK(K),WFBB(K),
     $                  WFTØT(K)),K=1,NSTAGE)
      IF (STAPE .GT. 0)  GØ TØ 50
   45 PRINT 2,   ((K,WØTNK(K),WØBB(K),WØTØT(K),WFTNK(K),WFBB(K),
     $             WFTØT(K)),K=1,NSTAGE)
    2 FØRMAT((1H ,5X,5HSTAGE,I2,5X,7HWØTNK =,E16.8,5X,6HWØBB =,E16.8,
     $        5X,7HWØTØT =,E16.8/18X,7HWFTNK =,E16.8,5X,6HWFBB =,E16.8,
     $        5X,7HWFTØT =,E16.8))
   50 CØNTINUE
C
  100 DØ 600  ITER1 = 1,MXIT1
      CALL FUNCG
      CALL CHKERR(Z)
      IF (Z .NE. 0.0)  GØ TØ 800
      ITER2 = 0
      IF (IT .EQ. 0)  GØ TØ 105
      WRITE (IT,5)  ((STAGS(3,K),WØR(K),STAGS(4,K),WFR(K),K),K=1,NSTAGE)
      IF (STAPE .GT. 0)  GØ TØ 110
  105 PRINT 5,   ((STAGS(3,K),WØR(K),STAGS(4,K),WFR(K),K),K=1,NSTAGE)
    5 FØRMAT((1H ,5X,5HWØI =,E16.8,3X,5HWØR =,E16.8,
     $            5X,5HWFI =,E16.8,3X,5HWFR =,E16.8,5X,5HSTAGE,I2))
  110 CØNTINUE
C
      DØ 500  I=1,NSTAGE
      J = NSTAGE - I + 1
      IF (PRØPD(4*J+1) .EQ. 0.0)  GØ TØ 500
      WØIL(J) = STAGS(3,J)
      WFIL(J) = STAGS(4,J)
      IF (WØR(J) .LE. WØEPSI(J))  GØ TØ 300
      IF (WFR(J) .LE. WFEPSI(J))  GØ TØ 200
```

FIGURE 57. SYMBOLIC LISTING OF SUBROUTINE PLOAD (cont.)

```
C         BØTH ØX AND FUEL RESIDUALS
      IF (PRØPD(4*J+1) .LT. 0.0)  GØ TØ 150
      CALL SETERR(6HRESID.)
      GØ TØ 811
  150 STAGS(3,J) = WØIL(J) - WØR(J)
      STAGS(4,J) = WFIL(J) - WFR(J)
      GØ TØ 400
  160 STAGS(3,J) = WØIL(J) - WØR(J)
      GØ TØ 400
  170 STAGS(4,J) = WFIL(J) - WFR(J)
      GØ TØ 400
C         ØNLY ØXIDIZER RESIDUAL
  200 IF (PRØPD(4*J+1) .LT. 0.0)  GØ TØ 160
      WF = WFIL(J) + WØR(J)/(1.0 + AVMR(J))
      IF (WF .GT. WFTØT(J))  GØ TØ 250
      STAGS(3,J) = WPRPN(J) - WF
      STAGS(4,J) = WF
      GØ TØ 400
C                TØØ MUCH ØXIDIZER FØR FULL FUEL TANK
  250 STAGS(4,J) = WFTØT(J)
      STAGS(3,J) = WØIL(J) - WØR(J) + (WFTØT(J) - WFIL(J)) * AVMR(J)
      GØ TØ 400
C
C
  300 IF (WFR(J) .LE. WFEPSI(J))  GØ TØ 500
C     ØNLY FUEL RESIDUAL
      IF (PRØPD(4*J+1) .LT. 0.0)  GØ TØ 170
      WØ = WØIL(J) + WFR(J) * AVMR(J)/(1.0 +AVMR(J))
      IF (WØ .GT. WØTØT(J))  GØ TØ 350
      STAGS(3,J) = WØ
      STAGS(4,J) = WPRPN(J) - WØ
      GØ TØ 400
C                TØØ MUCH FUEL FØR FULL ØXIDIZER TANK
  350 STAGS(3,J) = WØTØT(J)
      STAGS(4,J) = WFIL(J) - WFR(J) + (WØTØT(J) - WØIL(J))/AVMR(J)
      GØ TØ 400
C         DØ WEIGHT BUILD-UP
  400 IF (ØPTIØN .NE. 0.0)  WPL = WPL - (STAGS(3,J) - WØIL(J))
     $                                - (STAGS(4,J) - WFIL(J))
      IF (WPL .LT. 0.0)  GØ TØ 890
      ASSIGN 410 TØ NN
      GØ TØ 900
  410 GØ TØ 600
  500 CØNTINUE
C         CØNVERGED
      RETURN
C
  600 CØNTINUE
C         MAX ITER
  601 IF (IT .EQ. 0)  GØ TØ 605
```

FIGURE 57. SYMBOLIC LISTING OF SUBROUTINE PLOAD (cont.)

```
      WRITE (IT,1)
      IF (STAPE .GT. 0)  GØ TØ 610
  605 PRINT 1
    1 FØRMAT(41H       MAX ITERATIØNS - PRØPELLANT LØADING)
  610 CALL SETERR(5HPLØAD)
      GØ TØ 811
C
C         ERRØR CØNTRØL
  800 IF (ITER1 .GT. 1)  GØ TØ 850
      IF (IT .EQ. 0)  GØ TØ 805
      WRITE (IT,3)
      IF (STAPE .GT. 0)  GØ TØ 810
  805 PRINT 3
    3 FØRMAT(31H       NØMINAL TRAJECTØRY FAILED)
  810 CALL SETERR(5HPRØPL)
  811 DØ 815  I=1,NSTAGE
      STAGS(3,I) = WØIN(I)
      STAGS(4,I) = WFIN(I)
  815 CØNTINUE
      WPL = WPLN
      ASSIGN 820 TØ NN
      GØ TØ 900
  820 CØNTINUE
      RETURN
C
  850 ITER2 = ITER2 + 1
      IF (ITER2 .GE. MXIT2)  GØ TØ 601
      WØTRY = (WØIL(J) + STAGS(3,J))/2.0
      WFTRY = (WFIL(J) + STAGS(4,J))/2.0
      IF (ØPTIØN .NE. 0.0)  WPL = WPL - (WØTRY - STAGS(3,J))
     $                                - (WFTRY - STAGS(4,J))
      STAGS(3,J) = WØTRY
      STAGS(4,J) = WFTRY
      GØ TØ 400
C
C         NEGATIVE PAYLØAD
  890 IF (IT .EQ. 0)  GØ TØ 895
      WRITE (IT,4)
      IF (STAPE .GT. 0)  GØ TØ 810
  895 PRINT 4
    4 FØRMAT(22H       NEGATIVE PAYLØAD)
      GØ TØ 810
C
C                    WEIGHT BUILD-UP
  900 STAGS(2,NSTAGE) = STAGS(3,NSTAGE) + STAGS(4,NSTAGE)
     $                        + PRØPD(4*NSTAGE+2)
      NSTØP = NSTAGE-1
      DØ 910  N = 1,NSTØP
      M1 = NSTAGE - N
      M2 = M1 + 1
```

FIGURE 57. SYMBOLIC LISTING OF SUBROUTINE PLOAD (cont.)

```
910 STAGS(2,M1) = STAGS(2,M2) + STAGS(3,M1) + STAGS(4,M1)
     $                             + PROPD(4*M1+2)
      GO TO NN, (40,410,820)
C
      END
```

FIGURE 57. SYMBOLIC LISTING OF SUBROUTINE PLOAD (cont.)

XLIV. SUBROUTINE SKIP

This subroutine executes Subroutine TRNSFR and skips to the start of the re-ignition section. It therefore bypasses trajectory integration during coast.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL SKIP

SOLUTION METHOD

Execute Subroutine TRNSFR

1. CALL TRNSFR

Compute re-ignition time.

2. TIME = TIME + (TCST)(60.0)

Compute components of the integrated position vector at re-ignition.

3. XIN = (RXR)(RSUB0) - ROX

4. YIN = (RYR)(RSUB0) - ROY

5. ZIN = (RZR)(RSUB0) - ROZ

Compute velocity conversion factor.

6. CON = (RSUBO)/((13.447052)(60.0))

Compute components of velocity vector at re-ignition.

7. XDIN = (RDXR)(CON)

8. YDIN = (RDYR)(CON)

9. ZDIN = (RDZR)(CON)

Inject vehicle into transfer orbit.

10. CALL INJEC

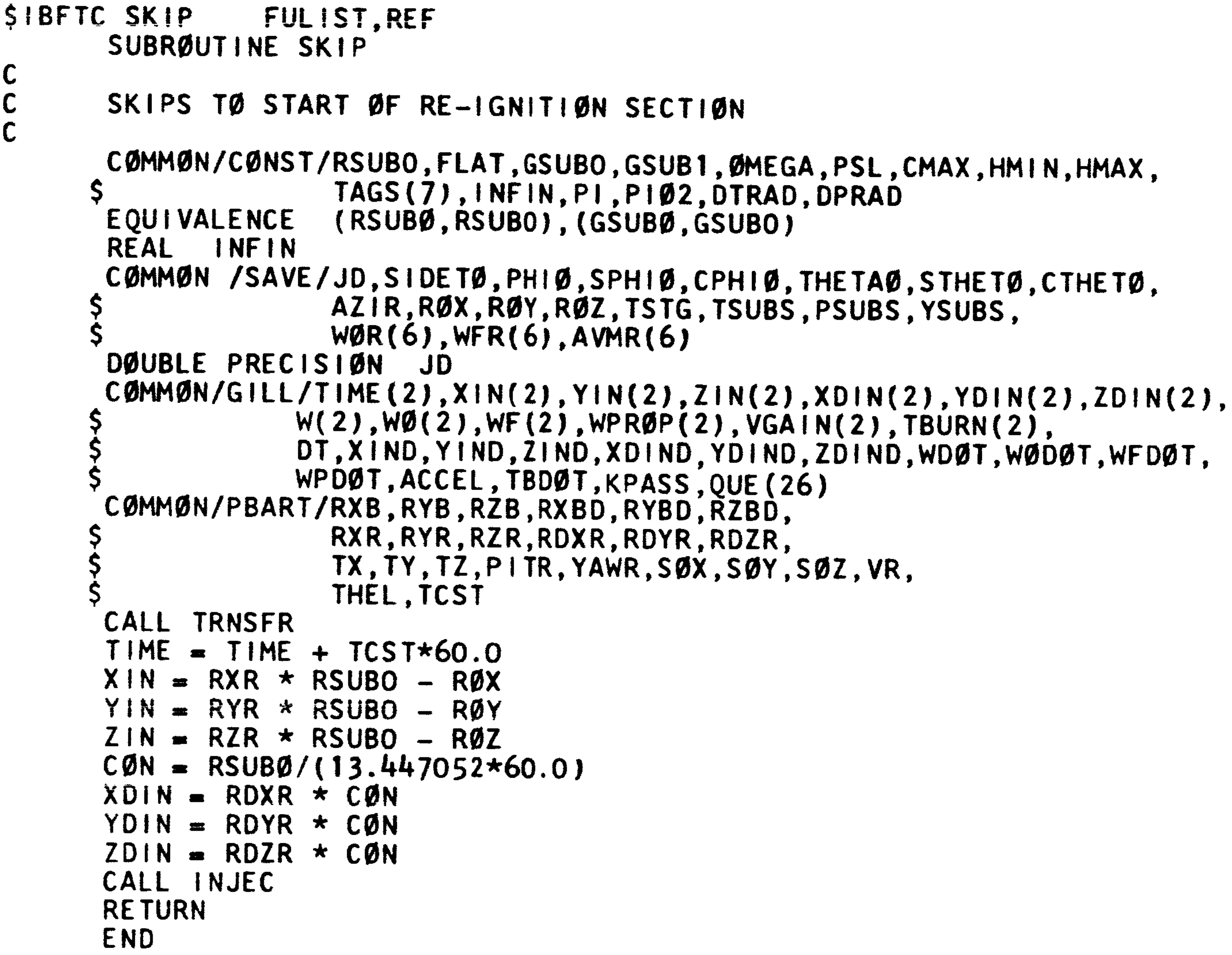
Return.

SUBROUTINE SKIP NOMENCLATURE

Symbol	Description	Units	Reference
CON	Conversion factor	$\frac{K_e \text{ min ft}}{\text{e.r. sec}}$	6, 7, 8, 9
RDXR	X-component of inertial velocity vector at re-ignition	$\text{e.r.}/K_e^{-1}$ min	/PBART/, 7
RDYR	Y-component of inertial velocity vector at re-ignition	$\text{e.r.}/K_e^{-1}$ min	/PBART/, 8
RDZR	Z-component of inertial velocity vector at re-ignition	$\text{e.r.}/K_e^{-1}$ min	/PBART/, 9
ROX	X-component of initial position vector	ft	/SAVE/, 3
ROY	Y-component of initial position vector	ft	/SAVE/, 4
ROZ	Z-component of initial position vector	ft	/SAVE/, 5
RSUB0	Earth equatorial radius	ft	EQUIV, /CONST/, 3, 4, 5
RSUBO	Earth equatorial radius	ft	EQUIV, 6
RXR	X-component of position vector at re-ignition	e.r.	/PBART/, 3
RYR	Y-component of position vector at re-ignition	e.r.	/PBART/, 4
RZR	Z-component of position vector at re-ignition	e.r.	/PBART/, 5

SUBROUTINE SKIP NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TCST	Coast time	min	/PBART/, 2
TIME	Instantaneous time	sec	/GILL/, 2
XDIN	X-component of inertial velocity vector	ft/sec	/GILL/, 7
XIN	X-component of integrated position vector	ft	/GILL/, 3
YDIN	Y-component of inertial velocity vector	ft/sec	/GILL/, 8
YIN	Y-component of integrated position vector	ft	/GILL/, 4
ZDIN	Z-component of inertial velocity vector	ft/sec	/GILL/, 9
ZIN	Z-component of integrated position vector	ft	/GILL/, 5



```
$IBFTC SKIP    FULIST,REF
      SUBRØUTINE SKIP
C
C     SKIPS TØ START ØF RE-IGNITIØN SECTIØN
C
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $             TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL   INFIN
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $             AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $             WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $           W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $           DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $           WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN/PBART/RXB,RYB,RZB,RXBD,RYBD,RZBD,
     $             RXR,RYR,RZR,RDXR,RDYR,RDZR,
     $             TX,TY,TZ,PITR,YAWR,SØX,SØY,SØZ,VR,
     $             THEL,TCST
      CALL TRNSFR
      TIME = TIME + TCST*60.0
      XIN = RXR * RSUBO - RØX
      YIN = RYR * RSUBO - RØY
      ZIN = RZR * RSUBO - RØZ
      CØN = RSUBØ/(13.447052*60.0)
      XDIN = RDXR * CØN
      YDIN = RDYR * CØN
      ZDIN = RDZR * CØN
      CALL INJEC
      RETURN
      END
```

FIGURE 58. SYMBOLIC LISTING OF SUBROUTINE SKIP

XLV. SUBROUTINE COAST

This subroutine executes Subroutine TRNSFR and stores the coast ending time in the section working data to provide a criterion for terminating integration at the correct reignition time.

CALLING SEQUENCE

CALL COAST

SOLUTION METHOD

Execute Subroutine TRNSFR.

1. CALL TRNSFR

Specify time as the section ending criterion.

2. PARAM = 0

Convert coast ending time to seconds.

3. VE = (TCST)(60.0)

Set thrust indicator for coast option.

4. THIND = 0

Return

SUBROUTINE COAST NOMENCLATURE

Symbol	Description	Units	Reference
PARAM	Section ending parameter	—	/WORKD/, 2
TCST	Coast time	min	/PBART/, 3
THIND	Thrust indicator	—	INTEGER, /WORKD/, 4
VE	Section ending time	sec	/WORKD/, 3

```
$IBFTC CØAST   FULIST,REF
      SUBRØUTINE CØAST
C
C     EXECUTES TRNSFR AND STØRES CØAST ENDING TIME IN SECTIØN WØRKING
C             DATA IN ØRDER TØ INTEGRATE TØ RE-IGNITIØN
C
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $               PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $               PADSK,VESK,PADRT1,PADRT2,
     $             SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $              ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL   ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN/PBART/RXB,RYB,RZB,RXBD,RYBD,RZBD,
     $               RXR,RYR,RZR,RDXR,RDYR,RDZR,
     $               TX,TY,TZ,PITR,YAWR,SØX,SØY,SØZ,VR,
     $               THEL,TCST
C
      CALL TRNSFR
      PARAM = 0
      VE = TCST * 60.0
      THIND = 0
      RETURN
      END
```

FIGURE 59. SYMBOLIC LISTING OF SUBROUTINE COAST

XLVI. SUBROUTINE TRNSFR

This subroutine is the main control routine for Block III. It executes Subroutine PATCH or LUNAR, and BART. It is called by Subroutines COAST or SKIP.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL TRNSFR

SOLUTION METHOD

Determine target body.

1. If TARGET $\neq$ 0, go to 4

   If TARGET = 0, go to 2

Lunar Transfer

2. CALL LUNAR (JD, TRANS, (RVCTR)/(RSUB0))
3. Go to 5

Planetary Transfer

4. CALL PATCH (TARGET, JD, TRANS, (RVCTR)/(RSUB0), 0)

5. Continue

Convert booster burn-out azimuth to degrees.

6. AZB = (AZIN)(DPRAD)

Convert booster burn-out latitude to degrees.

7. DLATB = (PHI) (DPRAD)

Compute booster burn-out east longitude.

8. DLONB = 360 - LONG

Compute east longitude of launch point.

9. DLONL = 360 - LONGI

Set average vehicle thrust and mass during re-ignition.

10. RAT = FVAC

11. RMAS = (W)/(GSUB1)

Convert booster burn time to minutes.

12. TIMB = (TFLIT)/(60)

Compute velocity conversion factor.

13. CON = (RSUBO)/((13.447052)(60))

Convert velocities at end of re-ignition and at end of boost to ft/sec.

14. VMF = (VMF) (CON)

15. VMB = (VMB) (CON)

Compute time of booster lift-off, time of re-ignition, and the re-ignition guidance parameters.

16. CALL BART (RVCTR, RSUBO, (OMEGA)(60))

Reset initial conditions for corrected launch time.

17. DSID = AMOD ((THEL)/(DPRAD) - THETAO, (2)(PI))

Compute launch sidereal time.

18. THETAO = THETAO + DSID

Compute initial Greenwich sidereal time.

19. SIDETO = SIDETO + DSID

Compute launch Cartesian coordinates.

20. CALL G1OB (RSUB0, FLAT, PHIO, THETAO, HI, 0, 0, 0,
ROX, ROY, ROZ, U, U, U, 1)

Compute sine and cosine of THETAO.

21. STHETO = Sin(THETAO)

22. CTHETO = Cos(THETAO)

Compute components of the integrated position vector at end of boost.

23. XIN = (RXB)(RSUBO) - ROX

24. YIN = (RYB)(RSUBO) - ROY

25. ZIN = (RZB)(RSUBO) - ROZ

Set components of inertial velocity vector at end of boost.

26. XDIN = (RXBD)(CON)

27. YDIN = (RYBD)(CON)

28. ZDIN = (RZBD)(CON)

Compute pitch and yaw angles for re-ignition.

```
29.  CALL G2OB(ROX, ROY, ROZ, SOX - (RDXR)(CON),
                SOY - (RDYR)(CON), SOZ - (RDZR)(CON),
                RSUBO, 0, U, U, U, U, PITR, YAWR, 1)

30.  PITR = PIO2 - PITR

31.  YAWR = YAWR - AZIR
```

Calculate sidereal time correction in hours, and pitch and yaw angles in degrees.

```
32.  DSIDH = (DSID)(DPRAD)/(15.0)

33.  PITD = (PITR)(DPRAD)

34.  YAWD = (YAWR)(DPRAD)
```

Print out variables of interest.

```
35.  IT = |OTAPE|

36.  If IT = 0, go to 39
     If IT = 0, go to 37

37.  WRITE DISDH, TCST, VMF, PITD, YAWD, SOX,
           SOY, SOZ on IT

38.  If OTAPE > 0, go to 40
     If OTAPE ≤ 0, go to 39

39.  PRINT DSIDH, TCST, VMF, PITD, YAWD,
           SOX, SOY, SOZ

40.  Continue
     Return
```

SUBROUTINE TRNSFR NOMENCLATURE

Symbol	Description	Units	Reference
AZB	Booster burn-out azimuth	deg	/TOL/, 6
AZIN	Booster burn-out inertial azimuth	rad	/DATA/, 6
CON	Velocity conversion factor	$\frac{K_e \text{ min ft}}{\text{e.r. sec}}$	13, 14, 15, 26, 27, 28, 29
CTHETO	Cosine of THETAO	—	/SAVE/, 22
DLATB	Latitude at end of boost	deg	/TOL/, 7
DLONB	East longitude at end of boost	deg	/TOL/, 8
DLONL	East longitude of launch point	deg	/TOL/, 9
DPRAD	Conversion factor	deg/rad	/CONST/, 6, 7, 17
DSID	Launch time correction factor	rad	17, 18, 19, 32
DSIDH	Launch time correction factor	hours	32, 37, 39
FLAT	Earth flattening	—	/CONST/, 20
FVAC	Vacuum thrust	lb	/WORKD/, 10
GSUB1	Weight-to-mass conversion factor	lb/slug	/CONST/, 11
HI	Initial altitude	ft	/INPUT/, 20
IT	Tape unit to write output	—	35, 36, 37

SUBROUTINE TRNSFR NOMENCLATURE

Symbol	Description	Units	Reference
JD	Julian date	days	/SAVE/, DP, 2, 4
LONGI	Initial west longitude	deg	/INPUT/, REAL, 9
LONG	Instantaneous west longitude	deg	/DATA/, REAL, 8
OMEGA	Earth rotation rate	rad/sec	/CONST/, 16
OTAPE	Standard output unit	—	/TAPES/, INTEGER, 35, 38
PHIO	Initial latitude	rad	/SAVE/, 20
PHI	Instantaneous latitude	rad	/DATA/, 7
PITR	Pitch angle	rad	/PBART/, 29, 30, 33
PITD	Pitch angle	degrees	33, 37, 39
RAT	Average thrust during re-ignition	lb	/TOL/, 10
RDXR	Velocity component in the X direction at the beginning of re-ignition	e.r./K_e^{-1} min	/PBART/, 29
RDYR	Velocity component in the Y direction at the beginning of re-ignition	e.r./K_e^{-1} min	/PBART/, 29
RDZR	Velocity component in the Z direction at the beginning of re-ignition	e.r./K_e^{-1} min	/PBART/, 29

SUBROUTINE TRNSFR NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RMAS	Average mass during re-ignition	slugs	/TOL/, 11
ROX	X-component of launch position vector	ft	/SAVE/, 20, 23, 29
ROY	Y-component of launch position vector	ft	/SAVE/, 20, 24, 29
ROZ	Z-component of launch position vector	ft	/SAVE/, 20, 25, 29
RSUB0	Earth equatorial radius	ft	/CONST/, EQUIV, 2, 4, 20
RSUBO	Earth equatorial radius	ft	EQUIV, 13, 16, 23, 24, 25, 29
RVCTR	Magnitude of position vector	ft	/DATA/, 2, 4, 16
RXBD	X-component of inertial velocity vector at end of boost	e.r./K_e^{-1} min	/PBART/, 26
RXB	X-component of position vector at end of boost	e.r.	/PBART/, 23
RYBD	Y-component of inertial velocity vector at end of boost	e.r./K_e^{-1} min	/PBART/, 27
RYB	Y-component of position vector at end of boost	e.r.	/PBART/, 24
RZBD	Z-component of inertial velocity vector at end of boost	e.r./K_e^{-1} min	/PBART/, 28

SUBROUTINE TRNSFR NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RZB	Z-component of position vector at end of boost	e.r.	/PBART/, 25
SIDETO	Initial Greenwich sidereal time	rad	/SAVE/, 19
SOX	X-component of transfer velocity	ft/sec	/PBART/, 29, 37, 39
SOY	Y-component of transfer velocity	ft/sec	/PBART/, 29, 37, 39
SOZ	Z-component of transfer velocity	ft/sec	/PBART/, 29, 37, 39
STHETO	Sine of THETAO	—	/SAVE/, 21
TARGET	Target body	—	/INPUT/, INTEGER, 1, 4
TCST	Coast time	min	/PBART/, 37, 39
TFLIT	Flight time	sec	/DATA/, 12
THEL	Sidereal time at launch	deg	/PBART/, 17
THETAO	Initial sidereal time	rad	/SAVE/, 17, 18, 20, 21, 22
TIMB	Flight time	min	/TOL/, 12
TRANS	Transfer time	days or hours	/INPUT/, 2, 4
U	Dummy variable	—	20, 29

SUBROUTINE TRNSFR NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
VMB	Magnitude of inertial velocity vector at end of boost	e.r./K_e^{-1} min of ft/sec	/OBART/, 15
VMF	Magnitude of required inertial velocity vector at end of re-ignition	e.r./K_e^{-1} min of ft/sec	/OBART/, 14, 37, 39
W	Vehicle weight	lb	/GILL/, 11
XDIN	X-component of inertial velocity vector	ft/sec	/GILL/, 26
XIN	X-component of the integrated position vector	ft	/GILL/, 23
YAWR	Yaw angle	rad	/PBART/, 29, 31, 34
YAWD	Yaw angle	degrees	34, 37, 39
YDIN	Y-component of inertial velocity vector	ft/sec	/GILL/, 27
YIN	Y-component of the integrated position vector	ft	/GILL/, 24
ZDIN	Z-component of inertial velocity vector	ft/sec	/GILL/, 28
ZIN	Z-component of the integrated position vector	ft	/GILL/, 25

```
$IBFTC TRNSFR  FULIST,REF
      SUBRØUTINE TRNSFR
C
C        DRIVER TØ EXECUTE PATCH, LUNAR AND BART
C              EXECUTED FRØ- SOBA AN+ INPUT BY CØAST ØR SKIP
C
      CØMMØN/TAPES/ITAPE,ØTAPE,STAPE
      INTEGER  ØTAPE,STAPE
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $             TAGS(7),INFIN,PI,PI02,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL  INFIN
      CØMMØN /INPUT/TI,VI,HI,LATI,LØNGI,BETAI,AZI,BETAO,WI,WPL,SIDETI,
     $             DATE(7),TARGET,TRANS,FLTR,FLSC,FLPRT,DUM(7),
     $             SECTS(20,12),STAGS(20,6),HEAD(11,15)
      REAL  LATI,LØNGI
      INTEGER  TARGET
      INTEGER  FLTR,FLSC,FLPRT
      CØMMØN/WØRKD/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $             PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $             PADSK,VESK,PADRT1,PADRT2,
     $           SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $            ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL  ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $             AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $             WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN/GILL/TIME(2),XIN(2),YIN(2),ZIN(2),XDIN(2),YDIN(2),ZDIN(2),
     $           W(2),WØ(2),WF(2),WPRØP(2),VGAIN(2),TBURN(2),
     $           DT,XIND,YIND,ZIND,XDIND,YDIND,ZDIND,WDØT,WØDØT,WFDØT,
     $           WPDØT,ACCEL,TBDØT,KPASS,QUE(26)
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $             PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $             GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $             QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $             CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $             ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN /ØBART/ PLAT,PSIDT,VMF,TRANG,VMB
      CØMMØN/PBART/RXB,RYB,RZB,RXBD,RYBD,RZBD,
     $             RXR,RYR,RZR,RDXR,RDYR,RDZR,
     $             TX,TY,TZ,PITR,YAWR,SØX,SØY,SØZ,VR,
     $             THEL,TCST
      CØMMØN /TØL/ AZB,DLATB,DLØNB,DLØNL,RAT,RMAS,TIMB
      IF (TARGET .NE. 0)  GØ TØ 100
C        LUNAR TRANSFER
      CALL LUNAR(JD,TRANS,RVCTR/RSUBO)
      GØ TØ 200
```

FIGURE 60. SYMBOLIC LISTING OF SUBROUTINE TRNSFR

```
C          PLANETARY TRANSFER
  100 CALL PATCH(TARGET,JD,TRANS,RVCTR/RSUBO,0.0)
C
  200 CØNTINUE
      AZB = AZIN * DPRAD
      DLATB = PHI * DPRAD
      DLØNB = 360.0 - LØNG
      DLØNL = 360.0 - LØNGI
      RAT = FVAC
      RMAS = W/GSUB1
      TIMB = TFLIT/60.0
      CØN = RSUBØ/(13.447052*60.0)
      VMF = VMF * CØN
      VMB = VMB * CØN
      CALL BART(RVCTR,RSUBØ,ØMEGA*60.0)
C        RESET INITIAL CØNDITIØNS FØR CØRRECTED LAUNCH
      DSID = AMØD(THEL/DPRAD-THETAØ,2.0QPI)
      THETAØ = THETAØ + DSID
      SIDETØ = SIDETØ + DSID
      CALL G1ØB(RSUBO,FLAT,PHIØ,THETAØ,HI,0.0,0.0,0.0,RØX,RØY,RØZ,U,U,U,
     $   1)
      STHETØ = SIN(THETAØ)
      CTHETØ = CØS(THETAØ)
      XIN = RXB * RSUBØ - RØX
      YIN = RYB * RSUBØ - RØY
      ZIN = RZB * RSUBØ - RØZ
      XDIN = RXBD * CØN
      YDIN = RYBD * CØN
      ZDIN = RZBD * CØN
C        CALCULATE PITCH AND YAW ANGLES FØR RE-IGNITIØN
      CALL G2ØB(RØX,RØY,RØZ,SØX-R+XRQCØN,SØY-RDYR*CØN,SØZ-RDZR*CØN,
     $          RSUBØ,0.0,U,U,U,U,PITR,YAWR,1)
      PITR = PIØ2 - PITR
      YAWR = YAWR - AZIR
C
C          PRINT-ØUT
C
  900 DSIDH = DSID * DPRAD/15.0
      PITD = PITR * DPRAD
      YAWD = YAWR * DPRAD
      IT = IABS(ØTAPE)
      IF (IT .EQ. 0)  GØ TØ 905
      WRITE (IT,1)  DSIDH,TCST,VMF,PITD,YAWD,SØX,SØY,SØZ
      IF (ØTAPE .GT. 0)  GØ TØ 910
  905 PRINT 1,  DSIDH,TCST,VMF,PITD,YAWD,SØX,SØY,SØZ
C
    1 FØRMAT(1H ,1H ,32HSIDEREAL TIME CØRRECTIØN (HØURS),E16.8,5X,
     $               23HTIME ØF CØAST (MINUTES),E16.8/1X,
     $               26HTRANSFER VELØCITY (FT/SEC),E16.8,2X,
     $               21HPITCH ANGLE (DEGREES),E16.8,2X,
```

FIGURE 60. SYMBOLIC LISTING OF SUBROUTINE TRNSFR (cont.)

```
     $                   19HYAW ANGLE (DEGREES),E16.8/1X,
     $                   28HGUIDANCE VELØCITIES (FT/SEC),5X,
     $                    4HVX =,E16.8,5X,4HVY =,E16.8,5X,4HVZ =,E16.8//)
 910 CØNTINUE
     RETURN
     END
```

FIGURE 60. SYMBOLIC LISTING OF SUBROUTINE TRNSFR (cont.)

XLVII. SUBROUTINE PATCH

This subroutine determines three-dimensional geocentric launch coordinates for direct, single-maneuver interplanetary transfers.

CALLING SEQUENCE

The calling sequence is:

CALL PATCH (TARGET, LDATE, DELT, RSUB0, GAM0)

SOLUTION METHOD

Compute arrival date.

1. TDATE = LDATE + DELT

EFEM provides orbital data for the planets, Find orbital elements, mean anomaly, mass, and radius of earth at launch and at impact.

2. CALL EFEM (LDATE, 3, VI1, MUI, RRI)
3. CALL EFEM (TDATE, 3, VI2, MUI, RRI)

Find similar information for target planet.

4. CALL EFEM (LDATE, TARGET, VJ1, MUJ, RRJ)

5. CALL EFEM (TDATE, TARGET, VJ2, MUJ, RRJ)

Compute times since perifocal passage. Find mean motion of earth.

6. NI1 = (KSUBS)(MUI)$^{.5}$(VI1(1))$^{-1.5}$

Divide mean anomaly by mean motion to find time.

7. TI1 = (VI1(3))/(NI1)

Refer to epoch of pericenter passage.

8. VI1(3) = 0.0

Repeat calculations for earth at time of impact.

9. TI2 = (VI2(3))/(NI1)

10. VI2(3) = 0.0

Repeat calculations for target planet at launch time . . .

11. NJ1 = (KSUBS)(MUJ)$^{.5}$(VJ1(1))$^{-1.5}$

12. TJ1 = (VJ1(3))/(NJ1)

13. VJ1(3) = 0.0

. . . and at impact time.

14. TJ2 = (VJ2(3))/(NJ1)

15. VJ2(3) = 0.0

TORB computes heliocentric position and velocity from the elements and time.

16. CALL TORB(VI1, VOI1, 0, MUI, TI1, KSUBS, ICON)

17. CALL TORB(VI2, VOI2, 0, MUI, TI2, KSUBS, ICON)

18. CALL TORB (VJ1, VOJ1, 0, MUJ, TJ1, KSUBS, ICON)

19. CALL TORB (VJ2, VOJ2, 0, MUJ, TJ2, KSUBS, ICON)

DTPRD computes the scalar product of two vectors.

20. CALL DTPRD (VOI1, VOI1, RI1)

Compute magnitude of earth's radius vector at launch . . .

21. RI1 = $(RI1)^{.5}$

22. CALL DTPRD (VOI2, VOI2, RI2)

. . . and at impact.

23. RI2 = $(RI2)^{.5}$

Compute magnitude of target planet radius vector at launch . . .

24. CALL DTPRD (VOJ1, VOJ1, RJ1)

25. RJ1 = $(RJ1)^{.5}$

26. CALL DTPRD (VOJ2, VOJ2, RJ2)

. . . and at impact.

```
27.  RJ2 = $(RJ2)^{.5}$
```

Use magnitudes to compute unit position vectors.

```
28.  Do 32  J = 1, 3
29.  UI1(J) = (VOI1(J))/(RI1)
30.  UI2(J) = (VOI2(J))/(RI2)
31.  UJ1(J) = (VOJ1(J))/(RJ1)
32.  UJ2(J) = (VOJ2(J))/(RJ2)
```

Find angle between earth at launch and target planet at impact.

```
33.  CALL DTPRD (UI1, UJ2, CPHI1)
```

Compute normal to transfer plane.

```
34.  CALL CRPRD (UI1, UJ2, W14)
```

CRPRD computes the vector product of two vectors.

```
35.  If W14(3) $\neq$ 0,  go to 39
     If W14(3) = 0,  go to 36
```

If transfer angle is zero, or a straight angle, slightly perturb final position . . .

```
36.  Do 38  J = 1, 3
37.  VOJ2(J) = VOJ2(J) - (.0001)(VOJ2(J + 3))
```

. . . and repeat calculation for new values.

```
38.  Go to 28
```

Compute unit vector normal to orbital plane.

```
39.  CALL DTPRD (W14, W14, W14P)
40.  W14P = $(W14P)^{.5}$
41.  Do 42  J = 1, 3
42.  W14(J) = (W14(J))/(W14P)
```

Compute central transfer angle.

```
43.  PHI1 = ACOS(CPHI1)
```

Determine proper quadrant of transfer angle.

```
44.  If W14(3) < 0, PHI1 = TWOPI - PHI1
     If W14(3) ≥ 0, go to 45
```

Compute inclination of orbital plane.

```
45.  I = ACOS(W14(3))
```

Calculate launch and target azimuths.

```
46.  CAZ1 = (UI1(2))(W14(1)) - (UI1(1))(W14(2))
47.  AZ1 = ACOS(CAZ1)
48.  CAZ4 = (UJ2(2))(W14(1)) - (UJ2(1))(W14(2))
```

49. AZ4 = ACOS(CAZ4)

Compute angle between transfer orbit and target orbit; find normal to target-planet orbit.

50. CALL CRPRD (UJ1, UJ2, W24)

51. CALL DTPRD (W24, W24, W24P)

52. $W24P = (W24P)^{.5}$

Compute unit vector in direction of normal.

53. Do 54 J = 1, 3

54. W24(J) = (W24(J))/(W24P)

Compute cosine of angle

55. CALL DTPRD (W24, W14, CETA)

and angle between transfer orbit and target-planet orbit.

56. ETA = ACOS (CETA)

Calculate longitude of ascending node.

57. $CNODE = (-W14(2))/((W14(1)^2 + W14(2)^2)^{.5})$

58. NODE = ACOS(CNODE)

59. If VOI1(2) < 0, NODE = PI + NODE

If VOI1(2) ≥ 0, go to 60

60. Do 62 J = 1,3

Store variables for input to VITER.

61. RI(J) = VOI1(J)

62. RI(J + 3) = VOJ2(J)

Set times for VITER.

63. TI(1) = LDATE

64. TI(2) = TDATE

Set constants for VITER.

Solar gravitational constant:

65. CON(1) = KSUBS

Sum of masses:

66. CON(2) = 1.0

Tolerance in anomaly:

67.. CON(3) = .00001

VITER performs an iteration on the true anomaly to solve Kepler's equation and develops the velocities corresponding to two positions in an orbit, given the time between observations.

68. CALL VITER (RI, RO, TI, CON, 1)

Calculate velocity in orbit relative to earth at launch.

69. Do 70 J = 1,3

70. RDOT(J) = VEL1(J) - VOI1(J + 3)

Transform to earth equatorial coordinates.

71. CECLIP = Cos(ECLIP)

72. SECLIP = Sin(ECLIP)

73. RPDOT(1) = RDOT(1)

74. RPDOT(2) = (RDOT(2))(CECLIP) - (RDOT(3))(SECLIP)

75. RPDOT(3) = (RDOT(2))(SECLIP) + (RDOT(3))(CECLIP)

Compute right ascension and declination of pierce point.

76. DENOM = $(\mathrm{RPDOT}(1)^2 + \mathrm{RPDOT}(2)^2)^{.5}$

77. SALFA = (RPDOT(2))/(DENOM)

78. CALFA = (RPDOT(1))/(DENOM)

Find right ascension.

79. CALL QUAD(CALFA, SALFA, ALFA)

Find declination.

80. DELTA = (ATAN(RPDOT(3)))/(DENOM)

Calculate residual velocity.

81. CALL DTPRD (RPDOT, RPDOT, SDOT2)

Convert heliocentric units to geocentric units.

82. $VP = (SDOT2)((KSUBS)/(KSUBE))^{2}/(((RRI)(1440))^{2})$

Calculate semi-major axis of escape hyperbola.

83. $AH = (-1.0)/(VP)$

Calculate launch velocity.

84. $V0 = ((2.0)/(RSUB0) + VP)^{.5}$

Calculate circular orbit velocity.

85. $VCO = ((1.0)/(RSUB0))^{.5}$

Calculate eccentricity of escape hyperbola.

86. $EH = (((VO)/(VCO))^{2}(((VO)/(VCO))^{2} - 2.0)$

$(\mathrm{Cos}\,(GAM0))^{2} + 1.0)^{.5}$

Calculate semi-parameter of escape hyperbola.

87. $PH = (-AH)(EH^{2} - 1.0)$

Calculate true anomaly of launch position.

88. $VSUB0 = ACOS((PH - RSUB0)/((EH)(RSUB0)))$

Calculate true anomaly of asymptote.

89. VINFIN = ACOS((-1.0)/(EH))

Calculate planetocentric transfer angle.

90. VBAR = VINFIN - VSUB0

Determine allowable range of declinations.

91. DMIN = DELTA - VBAR

92. DMAX = DELTA + VBAR

Return

SUBROUTINE PATCH NOMENCLATURE

Symbol	Description	Units	Reference
AH	Semi-major axis of escape hyperbola	c.u.	83, 87
ALFA	Right ascension of pierce point	radians	/OBART/, 79
AZ1	Azimuth at launch	radians	47
AZ4	Azimuth at impact	radians	49
CALFA	Cosine of right ascension of pierce point	—	78, 79
CAZ1	Cosine of launch azimuth	—	46, 47
CAZ4	Cosine of impact azimuth	—	48, 49
CECLIP	Cosine of obliquity of ecliptic	—	71, 74, 75
CETA	Cosine of angle between launch and target orbit planes	—	55, 56
CNODE	Cosine of longitude of ascending node	—	57, 58
CON	Constant array for input to VITER (see VITER documentation)	—	DIM, 65, 66, 67, 68
CPHI1	Cosine of PHI1	—	33, 43
DELT	Transfer time	days	CALL, 1
DELTA	Declination of pierce point	radians	/OBART/, 80, 91, 92

SUBROUTINE PATCH NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DENOM	Temporary storage for denominator in computing right ascension	—	76, 77, 78, 80
DMAX	Maximum . . .	radians	92
DMIN	. . . and minimum of possible pierce-point declinations	radians	91
ECLIP	Obliquity of ecliptic	radians	/ASTRO/, 71, 72
EH	Eccentricity of escape hyperbola	—	86, 87, 88, 89
ETA	Angle between transfer plane and target orbit plane	—	56
GAM0	Flight-path angle of vehicle at injection	radians	CALL, 86
ICON	Control variable in TORB (see TORB documentation)	—	16, 17, 18, 19
I	Inclination of orbital plane	radians	REAL, 45
KSUBE	Earth gravitational constant	e.r.$^{1.5}$/min	REAL, /ASTRO/, 82
KSUBS	Solar gravitational constant	a.u.$^{1.5}$/day	REAL, /ASTRO/, 6, 11, 16, 17, 18, 19, 65, 82
LDATE	Launch date	Julian days	CALL, DP, 1, 2, 4, 63
MUI	Sum of masses of initial planet and sun (from EFEM)	—	REAL, 2, 3, 6, 16, 17
MUJ	Sum of masses of target planet and sun (from EFEM)	—	REAL, 4, 5, 11, 18, 19

SUBROUTINE PATCH NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
NI1	Mean motion of earth	radians/day	REAL, 6, 7, 9
NJ1	Mean motion of target planet	radians/day	REAL, 11, 12, 14
NODE	Longitude of ascending node	radians	REAL, 58, 59
PHI1	Transfer angle from earth to target	radians	43, 44
PH	Semiparameter of escape hyperbola	c.u.	87, 88
PI	3.14159265	—	DATA, 59
RDOT	Velocity relative to earth at launch	c.u.	DIM, 70, 73, 74, 75
RI1	Magnitude of radius vector to earth at launch	c.u.	20, 21, 29
RI2	Magnitude of radius vector to earth at impact	c.u.	22, 23, 30
RI	Input array to VITER (see VITER documentation)	—	DIM, 61, 62, 68
RJ1	Magnitude of radius vector to target at launch	c.u.	24, 25, 31
RJ2	Magnitude of radius vector to target at impact	c.u.	26, 27, 32
RO	Output array from VITER (see VITER documentation)	—	EQUIV, DIM, 68
RPDOT	Relative velocity in earth equatorial coordinates	c.u.	DIM, 73, 74, 75, 76 77, 78, 80, 81

SUBROUTINE PATCH NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RRI	Radius of earth from EFEM	a.u.	2, 3, 82
RRJ	Radius of target planet	a.u.	4, 5
RSUB0	Injection altitude	c.u.	CALL, 84, 85, 88
SALFA	Sine of right ascension of pierce point	—	77, 79
SDOT2	Residual velocity	c.u.	81, 82
SECLIP	Sine of obliquity of ecliptic	—	72, 74, 75
TARGET	Identification index for target planet for input to EFEM	—	CALL, INTEGER, 4, 5
TDATE	Target date	Julian days	DP, 1, 3, 5, 64
TI	Input array to VITER (see VITER documentation)	—	DIM, 63, 64, 68
TI1	Time since perifocal passage of earth at launch . . .	c.u.	7, 16
TI2	. . . and at impact	c.u.	9, 17
TJ1	Time since perifocal passage of target planet at launch . . .	c.u.	12, 18
TJ2	. . . and at impact	c.u.	14, 19
TWOPI	2π	—	DATA, 44
UI1	Unit vector to position of earth at launch	c.u.	DIM, 29, 33, 34, 46
UI2	Unit vector to position of earth at impact	c.u.	DIM, 30

SUBROUTINE PATCH NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
UJ1	Unit vector to position of target planet at launch	c.u.	DIM, 31, 50
UJ2	Unit vector to position of target planet at impact	c.u.	DIM, 32, 33, 34, 48, 50
VBAR	Angle from launch position to asymptote of escape hyperbola	radians	/OBART/, 90, 91, 92
VCO	Circular parking orbit velocity	c.u.	/OBART/, 85, 86
VEL1	Inertial velocity in transfer orbit at injection	c.u.	EQUIV, DIM, 70
VINFIN	True anomaly of asymptote of escape hyperbola	c.u.	89, 90
VI1	Output array from EFEM (see EFEM documentation) (earth at launch)	—	DIM, 2, 6, 7, 8, 16
VI2	Output from EFEM (earth at impact)	—	DIM, 3, 9, 10, 17
VJ1	Output from EFEM (target at launch)	—	DIM, 4, 11, 12, 13, 18
VJ2	Output from EFEM (target at impact)	—	DIM, 5, 19
V0	Launch velocity	c.u.	DIM, 84, 86
VOI1	Output from TORB (see TORB documentation) (earth at launch)	—	DIM, 16, 20, 29, 59, 61, 70
VOI2	Output from TORB (earth at impact)	—	DIM, 17, 22, 30

SUBROUTINE PATCH NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
VOJ1	Output from TORB (target at launch)	—	DIM, 18, 24, 31
VOJ2	Output from TORB (target at impact)	—	DIM, 19, 26, 32, 37, 62
VP	Geocentric velocity	c.u.	82, 83, 84
VSUB0	True anomaly of launch position	radians	88, 90
W14	Vector normal to transfer plane	c.u.	DIM, 34, 35, 39, 42, 44, 45, 46, 48, 55, 57
W14P	Magnitude of W14	c.u.	39, 40, 42
W24	Vector normal to target planet orbit	c.u.	DIM, 50, 51, 52, 54, 55
W24P	Magnitude of W24	c.u.	51, 54

```
$IBFTC PATCH   FULIST,REF
C     SUBRØUTINE TØ DETERMINE 3-DIMENSIØNAL GEØCENTRIC LAUNCH
C          CØØRDINATES FØR DIRECT, SINGLE MANEUVER INTER-PLANETARY
C          TRANSFERS
      SUBRØUTINE PATCH(TARGET,LDATE,DELT,RSUBO,GAMO)
      INTEGER  TARGET
C          TARGET = TARGET PLANET   (1-8)
C          LDATE = JULIAN DATE ØF LAUNCH TIME
C         DELT = DESIRED TRANSFER TIME IN DAYS
C         RSUBO = LAUNCH RADIUS FRØM CENTER ØF EARTH (EARTH RADII)
C         GAMO = VEHICLE INITIAL     GEØCENTRIC FLIGHT PATH ANGLE (RAD)
C      USES SUBRØUTINES ACØS, CRPRD, DTPRD, EFEM,ELEM,QUAD,TØRB,VITER
      DIMENSIØN VI1(6),VI2(6),VJ1(6),VJ2(6),VØI1(6),VØI2(6),VØJ1(6),
     $        VØJ2(6),UI1(3),UI2(3),UJ1(3),UJ2(3),W14(3),W24(3),RI(6),
     $         TI(2),CØN(3),VØ(6),VEL1(3),VEL4(3),RDØT(3),RPDØT(3),RØ(6)
      REAL MUI,MUJ,NI1,NI2,NJ1,NJ2,I,NØDE,N,M1,M2,MO,NUM,LAM1,LAM4
      CØMMØN/ASTRØ/KSUBS,KSUBE,ECLIP,KINVE
      REAL  KSUBS,KSUBE,KINVE
      CØMMØN/ØBART/DELTA,ALFA,VO,VBAR,VCØ
      DØUBLE PRECISIØN LDATE,TDATE
      EQUIVALENCE(VEL1(1),RØ(1)),(VEL4(1),RØ(4))
      DATA  PI,TWØPI/3.14159265,6.28318531/
C         CALCULATE IMPACT DATE
      TDATE = LDATE + DELT
C         GET CLASSICAL ELEMENTS ØF PLANETARY ØRBITS
      CALL EFEM(LDATE,  3   ,VI1,MUI,RRI)
      CALL EFEM(TDATE,   3  ,VI2,MUI,RRI)
      CALL EFEM(LDATE,TARGET,VJ1,MUJ,RRJ)
      CALL EFEM(TDATE,TARGET,VJ2,MUJ,RRJ)
C         CØNVERT MEAN ANØMALY TØ TIME SINCE PERIFØCAL PASSAGE
      NI1 = KSUBS * SQRT(MUI) * VI1(1)**(-1.5)
      TI1 = VI1(3)/NI1
      VI1(3) = 0.0
      TI2 = VI2(3)/NI1
      VI2(3) = 0.0
      NJ1 = KSUBS * SQRT(MUJ) * VJ1(1)**(-1.5)
      TJ1 = VJ1(3)/NJ1
      VJ1(3) = 0.0
      TJ2 = VJ2(3)/NJ1
      VJ2(3) = 0.0
C         GET HELIØCENTRIC PØSITIØN AND VELØCITY FRØM ELEMENTS
      CALL TØRB(VI1,VØI1,0,MUI,TI1,KSUBS,ICØN)
      CALL TØRB(VI2,VØI2,0,MUI,TI2,KSUBS,ICØN)
      CALL TØRB(VJ1,VØJ1,0,MUJ,TJ1,KSUBS,ICØN)
      CALL TØRB(VJ2,VØJ2,0,MUJ,TJ2,KSUBS,ICØN)
C         CALCULATE MAGNITUDE ØF HELIØCENTRIC PØSITIØN VECTØRS
      CALL DTPRD(VØI1,VØI1,RI1)
      RI1 = SQRT(RI1)
      CALL DTPRD(VØI2,VØI2,RI2)
      RI2 = SQRT(RI2)
```

FIGURE 61. SYMBOLIC LISTING OF SUBROUTINE PATCH

```
      CALL DTPRD(VØJ1,VØJ1,RJ1)
      RJ1 = SQRT(RJ1)
      CALL DTPRD(VØJ2,VØJ2,RJ2)
      RJ2 = SQRT(RJ2)
C          CALCULATE UNIT VECTØRS ALØNG PØSITIØN VECTØRS
   10 DØ 11  J=1,3
      UI1(J) = VØI1(J)/RI1
      UI2(J) = VØI2(J)/RI2
      UJ1(J) = VØJ1(J)/RJ1
   11 UJ2(J) = VØJ2(J)/RJ2
C          CALCULATE CENTRAL TRANSFER ANGLE
      CALL DTPRD(UI1,UJ2,CPHI1)
      CALL CRPRD(UI1,UJ2,W14)
      IF (W14(3) .NE. 0.0)  GØ TØ 21
      DØ 20  J=1,3
   20 VØJ2(J) = VØJ2(J) - .0001*VØJ2(J+3)
      GØ TØ 10
   21 CALL DTPRD(W14,W14,W14P)
      W14P = SQRT(W14P)
      DØ 22  J=1,3
   22 W14(J) = W14(J)/W14P
      PHI1 = ACØS(CPHI1)
      IF (W14(3) .LT. 0.0)  PHI1 =  TWØPI - PHI1
C          CALCULATE INCLINATIØN ØF TRANSFER ØRBIT
      I = ACØS(W14(3))
C          CALCULATE LAUNCH AND TARGET AZIMUTHS
      CAZ1 = UI1(2)*W14(1) - UI1(1)*W14(2)
      AZ1 = ACØS(CAZ1)
      CAZ4 = UJ2(2)*W14(1) - UJ2(1)*W14(2)
      AZ4 = ACØS(CAZ4)
C          CALCULATE ANGLE BETWEEN TRANSFER AND TARGET PLANE ØRBITS
      CALL CRPRD(UJ1,UJ2,W24)
      CALL DTPRD(W24,W24,W24P)
      W24P = SQRT(W24P)
      DØ 30  J=1,3
   30 W24(J) = W24(J)/W24P
      CALL DTPRD(W24,W14,CETA)
      ETA = ACØS(CETA)
C          CALCULATE LØNGITUDE ØF ASCENDING NØDE
      CNØDE = -W14(2)/SQRT(W14(1)**2 + W14(2)**2)
      NØDE = ACØS(CNØDE)
      IF (VØI1(2) .LT. 0.0)  NØDE =    PI + NØDE
C          GET ELEMENTS AND TRUE ANØMALIES ØF TRANSFER ØRBIT
      DØ 40  J=1,3
      RI(J) = VØI1(J)
   40 RI(J+3) = VØJ2(J)
      TI(1) = LDATE
      TI(2) = TDATE
      CØN(1) = KSUBS
      CØN(2) = 1.0
```

FIGURE 61. SYMBOLIC LISTING OF SUBROUTINE PATCH (cont.)

```
      CØN(3) = .00001
      CALL VITER(RI,RØ,TI,CØN,1)
C         CALCULATE RELATIVE VELØCITY
      DØ 60  J=1,3
   60 RDØT(J) = VEL1(J) - VØI1(J+3)
C         TRANSFØRM RELATIVE VELØCITY TØ EARTH EQUATØRIAL CØØRDINATES
      CECLIP = CØS(ECLIP)
      SECLIP = SIN(ECLIP)
      RPDØT(1) = RDØT(1)
      RPDØT(2) = RDØT(2) * CECLIP - RDØT(3) * SECLIP
      RPDØT(3) = RDØT(2) * SECLIP + RDØT(3) * CECLIP
C         CALCULATE RIGHT ASCENSIØN AND DECLINATIØN ØF PIERCE PØINT
      DENØM = SQRT(RPDØT(1)**2 + RPDØT(2)**2)
      SALFA = RPDØT(2)/DENØM
      CALFA = RPDØT(1)/DENØM
      CALL QUAD(CALFA,SALFA,ALFA)
      DELTA = ATAN(RPDØT(3)/DENØM)
C         CALCULATE RESIDUAL VELØCITY
      CALL DTPRD(RPDØT, RPDØT,SDØT2)
C        CØNVERT HELIØCENTRIC VELØCITY TØ GEØCENTRIC VELØCITY
      VP = SDØT2 * (KSUBS/KSUBE)**2 / (RRI*1440.0)**2
C         CALCULATE SEMI-MAJØR AXIS ØF ESCAPE HYPERBØLA
      AH = -1.0/VP
C         CALCULATE LAUNCH VELØCITY
      VO = SQRT(2.0/RSUBO + VP)
C         CALCULATE SURFACE CIRCULAR VELØCITY
      VCØ = SQRT(1.0/RSUBO)
C         CALCULATE ECCENTRICITY ØF ESCAPE HYPERBØLA
      EH = SQRT((VO/VCØ)**2 * ((VO/VCØ)**2 - 2.0) * CØS(GAMO)**2 + 1.0)
C         CALCULATE SEMI-PARAMETER FØR ESCAPE HYPERBØLA
      PH = -AH * (EH**2 - 1.0)
      VSUBO = ACØS((PH - RSUBO)/(EH * RSUBO))
      VINFIN = ACØS(-1.0/EH)
      VBAR = VINFIN - VSUBO
C         EVALUATE LIMITS ØF LAUNCH AZIMUTH AND RIGHT ASCENSIØN
      DMIN = DELTA - VBAR
      DMAX = DELTA + VBAR
      RETURN
      END
```

FIGURE 61. SYMBOLIC LISTING OF SUBROUTINE PATCH (cont.)

XLVIII. SUBROUTINE EFEM

This subroutine provides the classical set of orbital elements for the planets Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus and Neptune at a given Julian date. A set of mean planetary elements for epoch 1960 January 1.5 (ephemeris time) are stored within this routine. The classical set of orbital elements are computed for the given date. The orbital elements are referenced to a heliocentric coordinate system with the ecliptic plane as the fundamental plane. The stored elements are from Ref. [3].

CALLING SEQUENCE

The subroutine calling sequence is:

```
CALL EFEM (DATE, PLANET, V, MU, RR)
```

PLANET is the planetary designator such that:

1 is for Mercury
2 is for Venus
3 is for Earth
4 is for Mars
5 is for Jupiter
6 is for Saturn
7 is for Uranus
8 is for Neptune

The array V contains the classical set of orbital elements at the given Julian date with:

V(1) = semi-major axis, a.u.

V(2) = eccentricity

V(3) = mean anomaly, radians

V(4) = inclination, radians

V(5) = longitude of ascending node, radians

V(6) = argument of perifocus, radians

This subroutine uses COMMON blocks MATCON and ASTCON.

SOLUTION METHOD

Set value of planetary radius:

1. RR = PR(PLANET)

Compute argument of perifocus:

2. QMEGA = PMEGA(PLANET) - NODE(PLANET)

Compute mean anomaly at epoch:

3. M = L(PLANET) - PMEGA(PLANET)

Compute sum of masses:

4. MU = 1 + MASS(PLANET)

Set value of semi-major axis:

5. V(1) = A(PLANET)

Set value of eccentricity:

6. V(2) = E(PLANET)

Compute mean motion:

7. $N = (KSUBS)(MU^{.5})(V(1)^{-1.5})$

Compute mean anomaly at given date:

8. V(3) = (M)(DTR) + (N)(DATE - 2436935)

Compute mean anomaly at given date between $0 - 2\pi$:

9. V(3) = AMOD(V(3), TWOPI)

AMOD is a FORTRAN function used for remaindering.

Compute orbital inclination in radians:

10. V(4) = (I(PLANET))(DTR)

Compute longitude of ascending node in radians:

11. V(5) = (NODE(PLANET))(DTR)

Compute argument of perifocus in radians:

12. V(6) = (QMEGA)(DTR)

Return

SUBROUTINE EFEM NOMENCLATURE

Symbol	Description	Units	Reference
A	Semi-major axis	a.u.	DIM, DATA, 5
DATE	Julian Date	days	CALL, 8
DTR	Constant	rad/deg	DATA, 8, 10, 11, 12
E	Eccentricity	—	DIM, DATA, 6
I	Orbital inclination	degrees or radians	DIM, REAL, DATA, 10
KSUBS	Gravitational constant	$\text{a.u.}^{1.5}/\text{day}$	/ASTRO/, REAL, 7
L	Mean longitude	degrees	DIM, REAL, DATA, 3
MASS	Planetary mass based upon the Sun as a reference	—	DIM, REAL, DATA, 4
M	Mean anomaly	degrees	REAL, 3, 8
MU	Sum of masses	—	CALL, REAL, 4, 7
NODE	Longitude of ascending	degrees or radians	DIM, REAL, DATA, 2, 11
N	Mean motion	rad/day	7, 8
PLANET	Planetary designator	—	CALL, INTEGER, 1, 2, 3, 4, 5, 6, 10, 11
PMEGA	Mean longitude of perihelion	degrees	DIM, DATA, 2, 3
PR	Planetary radius	a.u.	DIM, DATA, 1
QMEGA	Argument of perifocus	radians	2, 12

SUBROUTINE EFEM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RR	Planetary radius	a.u.	CALL, 1
TWOPI	Constant	—	DATA, 9
V	Output array	—	CALL, DIM, 5, 6, 7, 8, 9, 10, 11, 12

```
$IBFTC EFEM    FULIST,REF
      SUBRØUTINE EFEM(DATE,PLANET,V,MU,RR)
C         SUBRØUTINE TØ DETERMINE MEAN ELEMENTS ØF PLANETARY ØRBITS
C         FØR EPØCH 1960 JANUARY 1.5 EPHEMERIS TIME
C              RR IS THE PLANETARY RADIUS IN AU
      INTEGER  PLANET
      REAL  MU,MASS,I,NØDE,M,L
      DØUBLE PRECISIØN  N,DATE
      CØMMØN/ASTRØ/KSUBS,KSUBE,ECLIP,KINVE
      REAL  KSUBS,KSUBE,KINVE
      DATA  TWØPI,DTR/6.28318531,0.0174532925/
      DIMENSIØN  V(6),A(8),E(8),I(8),NØDE(8),PMEGA(8),L(8),MASS(8)
      DIMENSIØN PR(8)
      DATA  (PR(J),J=1,8)/0.000016172,0.000041075,0.000042569,
     $         0.000022258,0.000466989,0.000384408,0.000158602,
     $         0.000144086/
      DATA  (MASS(J),J=1,8)/16.3E-8,2.4472E-6,3.00763E-6,3.238E-7,
     $    9.5379E-4,2.85907E-4,4.3684E-5,5.1776E-5/,
     $    (A(J),J=1,8)/0.387099,0.723322,1.0,1.523691,5.202803,
     $    9.538843,19.181951,30.057779/,
     $    (E(J),J=1,8)/0.205627,0.006793,.016726 ,0.093368,0.048435,
     $    0.055682,0.047209,0.008575/,
     $    (I(J),J=1,8)/7.00399,3.39423,0.0,1.84991,1.30536,2.48991,
     $    0.77306,1.77375/,
     $    (NØDE(J),J=1,8)/47.85714,76.31972,0.0,49.24903,100.04444,
     $    113.30747,73.79630,131.33980/,
     $    (PMEGA(J),J=1,8)/76.83309,131.00831,102.25253,335.32269,
     $    13.67823,92.26447,170.01083,44.27395/,
     $    (L(J),J=1,8)/222.62165,174.29431,100.15815,258.76729,
     $    259.83112,280.67135,141.30496,216.94090/
      RR = PR(PLANET)
      QMEGA = PMEGA(PLANET) - NØDE(PLANET)
      M = L(PLANET) - PMEGA(PLANET)
      MU = 1.0 + MASS(PLANET)
      V(1)= A(PLANET)
      V(2)= E(PLANET)
      N = KSUBS * SQRT(MU) * V(1)**(-1.5)
      V(3) = M * DTR + N * (DATE - 2436935.0)
      V(3)=AMØD(V(3),TWØPI)
      V(4) = I(PLANET) * DTR
      V(5) = NØDE(PLANET) * DTR
      V(6) = QMEGA * DTR
      RETURN
      END
```

FIGURE 62. SYMBOLIC LISTING OF SUBROUTINE EFEM

XLIX. SUBROUTINE JULDT

This subroutine computes a double precision Julian date and corresponding Greenwich sidereal time, given the local time and time zone.

CALLING SEQUENCE

The calling sequence is:

CALL JULDT (TIME, OMEGA, TJD, THETG)

The local time and time zone are brought into the subroutine in the array TIME with:

TIME(1) = Year (1948-2100)

TIME(2) = Month (1-12)

TIME(3) = Day (1-31)

TIME(4) = Hour (0-24)

TIME(5) = Minute (0-60)

TIME(6) = Second (0-60)

TIME(7) = Time zone (0-23)

SOLUTION METHOD

Determine if year is a leap year.

1. X2 = AMOD(TIME(1), 4.0)

AMOD is a FORTRAN function used for remaindering.

2. If X2 = 0, go to 3

   If X2 $\neq$ 0, go to 4

3. X(2) = 29.0

Convert local time to universal time.

4. UTIME(4) = TIME(4) + TIME(7)
5. If UTIME(4) $<$ 24.0, go to 16

   If UTIME(4) $\geq$ 24.0, go to 6

6. UTIME(3) = TIME(3) + 1.0
7. UTIME(4) = UTIME(4) - 24.0
8. K = TIME(2)
9. If UTIME(3) $\leq$ X(K), go to 17

   If UTIME(3) $>$ X(K), go to 10

10. UTIME(2) = TIME(2) + 1.0
11. UTIME(3) = 1.0
12. If UTIME(2) $\leq$ 12.0, go to 18

    If UTIME(2) $>$ 12.0, go to 13

13. UTIME(1) = TIME(1) + 1.0

```
14.  UTIME(2) = 1.0

15.  Go to 19

16.  UTIME(3) = TIME(3)

17.  UTIME(2) = TIME(2)

18.  UTIME(1) = TIME(1)
```

Determine number of days in transpired years since January 0 1948 U.T.

```
19.  YR = UTIME(1)

20.  Y(2) = (365.0)(YR - 1948.0)
```

Correct for number of leap years since 1948.

```
21.  Y(3) = AINT((YR - 1945.0)(.25))
```

AINT is a FORTRAN function used for truncation.

Determine number of days in transpired months since beginning of year.

```
22.  J = IFIX(UTIME(2) - 1.0)
```

IFIX is a FORTRAN function which converts from real to integer.

```
23.  Y(4) = 0

24.  If J ≤ 0, go to 26

     If J > 0, go to 25

25.  Y(4) = Y(4) + X(I),  I = 1, J

26.  D = Y(1) + Y(2) + Y(3) + Y(4) + UTIME(3)
```

```
27.  Z = (UTIME(4))/(24) + (TIME(5))/(1440) + (TIME(6))/(86400)

 Compute double precision Julian date.

28.  YD    = Z

29.  TJDD  = D

30.  TJD   = YD + TJDD

 Compute Greenwich sidereal time at Julian date TJD.

31.  T  =  (D - 2436935.0)/(36525.0)

32.  ALPHA1 = 100.152475 + (36000)(T) +

                              (.7694 + (.0003871)(T))(T)

33.  ALPHA2 = AMOD(ALPHA1 , 360.0)

34.  THETG = (ALPHA2)/(57.2957795) + (Z)(1440)(OMEGA)

 Reset X(2) to original value.

35.  X(2) = 28.0

     Return
```

SUBROUTINE JULDT NOMENCLATURE

Symbol	Description	Units	Reference
ALPHA1	Right ascension of Greenwich at zero hours U.T. of computed Julian date	degrees	32, 33
ALPHA2	Right ascension ($0 - 2\pi$) of Greenwich at zero hour U.T. of computed Julian date	degrees	33, 34
D	Julian date at zero hours U.T. of given date	days	26, 29, 31
I	—	—	25
J	—	—	22, 24, 25
K	—	—	8, 9
OMEGA	Earth's rotation rate	radians/minute	CALL, 34
T	Number of Julian centuries since Jan 0.5 1960	Julian centuries	31, 32
THETG	Greenwich sidereal time at computed Julian date	radians	CALL, 34
TIME	Input array	—	CALL, DIM, 1, 4, 6, 8, 10, 13, 16, 17, 18, 27
TJD	Computed double precision Julian date	days	CALL, DP, 30
TJDD	D in double precision	days	DP, 29, 30
UTIME	Array containing elements of universal time	—	DIM, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 16, 17, 18, 19, 22, 26, 27

SUBROUTINE JULDT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
X	Array containing number of days in each month	days	DIM, DATA, 3, 9, 25, 35
X2	—	—	1, 2
Y	—	—	DIM, DATA, 20, 21, 23, 25, 26
YD	Z in double precision	days	DP, 28, 30
YR	—	—	19, 20, 21
Z	Portion of Julian date	days	27, 28, 34

```
$IBFTC JULDT   REF,FULIST
      SUBROUTINE JULDT (TIME,OMEGA,TJD,THETG)
C     CALCULATES JULIAN DATE AND CORRESPONDING THETAG .
C     TIME IS A VECTOR CONTAINING YEAR, MONTH, DAY, HOUR, MIN., SEC.,ZONE
C     IN THAT ORDER .
C     OMEGA IS INPUT IN RADIANS/MINUTE
C        ASSUMES YEAR GREATER THAN 1947. AND LESS THAN 2100.
C                MONTH LESS THAN OR EQUAL TO 12.
C                DAY NOT GREATER THAN NO. OF DAYS IN YR. AND MONTH ABOVE
      DOUBLE PRECISION TJD,YD,TJDD
      DIMENSION TIME(7),X(12),Y(4),UTIME(4)
      DATA (X(I),I=1,12)/31.,28.,31.,30.,31.,30.,31.,31.,30.,31.,30.,31.
     $/,Y(1)/2432550.5/
      X2 = AMOD(TIME(1),4.0)
      IF (X2) 6,5,6
    5 X(2)=29.0
C          CONVERT LOCAL TIME TO UNIVERSAL TIME
    6 UTIME(4)=TIME(4)+TIME(7)
      IF (UTIME(4)-24.0)37,10,10
   10 UTIME(3)=TIME(3)+1.0
      UTIME(4)=UTIME(4)-24.0
      K=TIME(2)
      IF (UTIME(3)-X(K))38,38,20
   20 UTIME(2)=TIME(2)+1.0
      UTIME(3)=1.0
      IF (UTIME(2)-12.0)39,39,30
   30 UTIME(1)=TIME(1)+1.0
      UTIME(2)=1.0
      GO TO 40
   37 UTIME(3)= TIME(3)
   38 UTIME(2)= TIME(2)
   39 UTIME(1)= TIME(1)
C        DAYS IN TRANSPIRED YEARS SINCE JANUARY 0 1948
   40 YR=UTIME(1)
      Y(2)=365.0*(YR-1948.0)
C        LEAP YEAR CORRECTION
      Y(3)=AINT((YR-1945.0)*.25)
C        DAYS IN TRANSPIRED MONTHS
      J=IFIX(UTIME(2)-1.0)
      Y(4)=0.0
      IF(J)60,60,45
   45 DO 50 I=1,J
   50 Y(4)=Y(4)+X(I)
   60 D    =Y(1)+Y(2)+Y(3)+Y(4)+UTIME(3)
      Z=UTIME(4)/24.+ TIME(5)/1440.+ TIME(6)/86400.
      YD=Z
      TJDD=D
      TJD=YD+TJDD
C          COMPUTE R.A. OF GREENWICH AT 0 HOUR (GMT)
C          OF GIVEN JULIAN DATE (DEGREES).
```

FIGURE 63. SYMBOLIC LISTING OF SUBROUTINE JULDT

```
      T=(D-2436935.0)/36525.0
      ALPHA1=100.152475+36000.*T + (.7694+.0003871*T)*T
      ALPHA2=AMOD(ALPHA1,360.0)
C          COMPUTE GREENWICH SIDEREAL TIME
      THETG=ALPHA2/57.2957795+Z*1440.*OMEGA
      X(2)=28.0
      RETURN
      END
```

FIGURE 63. SYMBOLIC LISTING OF SUBROUTINE JULDT (cont.)

L. SUBROUTINE TORB

This subroutine computes the position and velocity vectors of an orbiting body at a particular time, given the classical set of orbital elements.

CALLING SEQUENCE

The calling sequence is:

```
CALL TORB (V, VO, L, AMU, TI, FK, ICON)
```

The orbital elements are brought into the subroutine in the array V with:

V(1) = semi-major axis for elliptical and hyperbolic orbits
= perifocal distance for parabolic orbits

V(2) = eccentricity

V(3) = time of perifocal passage

V(4) = orbital inclination

V(5) = longitude of ascending node

V(6) = argument of perifocus

The semi-major axis and perifocal distance are in canonical units (c.u.) while the orientation angles are in radians. The time of perifocal passage must possess units which are consistent with those for the gravitational constant FK.

The position and velocity vectors of interest are in the array VO with VO(1) through VO(3) containing the position components, and VO(4) through VO(6) containing the velocity components. Both the position and velocity components are in canonical units.

The parameter L must be set to a non-zero quantity if the conic is a parabola. Otherwise, it must be set to zero.

SOLUTION METHOD

Store input array V into array VI:

1. VI(I) = V(I) , I = 1 to 6

Determine sines and cosines of orientation angles:

2. Q000FL = Cos (VI(6))

3. CAND = Cos (VI(5))

4. SOM = Sin (VI(6))

5. SAND = Sin (VI(5))

6. COSI = Cos (VI(4))

7. SINI = Sin (VI(4))

Determine a unit vector $\underline{P}$ which points toward perifocus:

8. PX = (Q000FL)(CAND) - (SOM)(SAND)(COSI)

9. PY = (Q000FL)(SAND) + (SOM)(CAND)(COSI)

10. PZ = (SOM)(SINI)

Determine a unit vector $\underline{Q}$ which lies in the orbit plane and is advanced from $\underline{P}$ by 90 degrees:

11. QX = - (SOM)(CAND) - (Q000FL)(SAND)(COSI)

12. QY = - (SOM)(SAND) + (Q000FL)(CAND)(COSI)

13. QZ = (Q000FL)(SINI)

Determine if conic is a parabola:

14. If L = 0, go to 15

If L $\neq$ 0, go to 67

Determine if conic is a hyperbola:

15. If VI(1) < 0, go to 44

If VI(1) > 0, go to 16

Computations for Ellipse or Circle

Compute mean anomaly:

16. $EN = (FK)((AMU)/(VI(1))^3)^{.5}$

17. AM = (EN)(TI - VI(3))

Determine if conic is an ellipse or a circle:

18. If VI(2) = 0, go to 19

If VI(2) $\neq$ 0, go to 21

19. ICON = 1

20. Go to 23

21. ICON = 2

Determine mean anomaly between 0 and 2π:

22. If $|AM| < 2\pi$, go to 25

If $|AM| \geq 2\pi$, go to 23

23. $JIX = (AM)/(6.28318530)$

24. $AM = AM - (6.28318530)(FLOAT(JIX))$

FLOAT is a FORTRAN function which converts the argument from integer to real.

Approximate eccentric anomaly:

25. $CEJ = AM + (VI(2))(Sin(AM))$

$+ (.5)(VI(2))^2(Sin(2AM))$

$+ (.125)(VI(2))^3((3)(Sin((3)(AM))) - Sin(AM))$

Initialize iteration counter:

26. $N = 0$

Compute eccentric anomaly:

27. $AMV = CEJ - (VI(2))(Sin(CEJ))$

28. $DEJ = (AM - AMV)/(1 - (VI(2))(Cos(CEJ)))$

29. $CEJ = CEJ + DEJ$

30. $N = N + 1$

31. If $N \leq 15$, go to 32

If $N > 15$, go to 81

32. If $\left| AM - AMV \right| \leq 10^{-7}$, go to 33

If $\left| AM - AMV \right| > 10^{-7}$, go to 27

Compute position and velocity vectors:

33. $XO = (VI(1))(Cos(CEJ) - VI(2))$

34. $YO = (VI(1))(Sin(CEJ))(1 - VI(2)^2)^{.5}$

35. $VO(1) = (PX)(XO) + (QX)(YO)$

36. $VO(2) = (PY)(XO) + (QY)(YO)$

37. $VO(3) = (PZ)(XO) + (QZ)(YO)$

38. $RI = (VO(1)^2 + VO(2)^2 + VO(3)^2)^{.5}$

39. $XO = -(Sin(CEJ))((AMU)(VI(1)))^{.5}/(RI)$

40. $YO = ((AMU)(VI(1))(1 - VI(2)^2))^{.5}(Cos(CEJ))/(RI)$

41. $VO(4) = (XO)(PX) + (YO)(QX)$

42. $VO(5) = (XO)(PY) + (YO)(QY)$

43. $VO(6) = (XO)(PZ) + (YO)(QZ)$

Return

Computations for Hyperbola

Compute mean anomaly:

44. $EN = (FK)((AMU)/(-VI(1)^3))^{.5}$

45. $ICON = 4$

46. $AM = (EN)(TI - VI(3))$

```
47.  $RATIO = (VI(2) - 1)/(VI(2))$
48.  $Q = (VI(1))(1 - VI(2))$
49.  $RAD3 = ((3)(AM) + ((9)(AM)^2 + (8)(Q)^3)^{.5})^{1/3}$
50.  $FJ = RAD3 - (2)(Q)/(RAD3)$
51.  $N = 0$
52.  $AMO = (VI(2))(\exp(FJ) - \exp(-FJ))/(2) - FJ$
53.  $DFJ = (AM - AMO)/((VI(2))(\exp(FJ) + \exp(-FJ))/(2) -1)$
54.  $FJ = DFJ + FJ$
55.  $N = N + 1$
56.  If $N < 15$, go to 57
     If $N \geq 15$, go to 81
57.  If $(RATIO)(\left| DFJ/FJ \right|) \leq (.5)(10^{-6})$, go to 58
     If $(RATIO)(\left| DFJ/FJ \right|) > (.5)(10^{-6})$, go to 52
Compute position vector:
58.  $XO = (VI(1))((\exp(FJ) + \exp(-FJ))/(2) - VI(2))$
59.  $YO = -(VI(1))(VI(2)^2 - 1)^{.5}(\exp(FJ) - \exp(-FJ))/(2)$
60.  $VO(1) = (XO)(PX) + (YO)(QX)$
61.  $VO(2) = (XO)(PY) + (YO)(QY)$
62.  $VO(3) = (XO)(PZ) + (YO)(QZ)$
Compute magnitude of position vector:
63.  $RI = (VO(1)^2 + VO(2)^2 + VO(3)^2)^{.5}$
```

64. $XO = -(-(AMU)(VI(1)))^{.5}$

$(\exp(FJ) - \exp(-FJ))/((2)(RI))$

65. $YO = ((AMU)(VI(1))(1 - VI(2)^{2}))^{.5}$

$(\exp(FJ) + \exp(-FJ))/((2)(RI))$

Compute velocity vector:

66. Go to 41

Computations for Parabola

Compute mean anomaly:

67. $EN = (FK)(AMU)^{.5}$

68. $ICON = 3$

69. $AM = (EN)(TI - VI(3))$

Compute position vector:

70. $RAD3 = ((3)(AM) + ((9)(AM)^{2} + (8)(VI(1))^{3})^{.5})^{1/3}$

71. $DJ = RAD3 - (2)(VI(1))/(RAD3)$

72. $XO = VI(1) - (.5)(DJ)^{2}$

73. $YO = (DJ)((2)(VI(1)))^{.5}$

74. $VO(1) = (XO)(PX) + (YO)(QX)$

75. $VO(2) = (XO)(PY) + (YO)(QY)$

76. $VO(3) = (XO)(PZ) + (YO)(QZ)$

Compute magnitude of position vector:

77. $RI = (VO(1)^{2} + VO(2)^{2} + VO(3)^{2})^{.5}$

78. XO = -(DJ)/(RI)

79. YO = $((2)(VI(1))^{.5}/(RI)$

Compute velocity vector:

80. Go to 41

Error return

81. CALL SETERR (4HTORB)

82. Return

SUBROUTINE TORB NOMENCLATURE

Symbol	Description	Units	Reference
AM	Mean anomaly	radians	17, 22, 23, 24, 25, 28, 32, 46, 49, 53, 69, 70
AMO	Mean anomaly for hyperbolic orbit	radians	52, 53
AMU	Sum of masses	—	CALL, 16, 39, 40, 44, 64, 65, 67
AMV	Mean anomaly	radians	27, 28, 32
CAND	Cosine of the longitude of ascending node	—	3, 8, 9, 11, 12
CEJ	Eccentric anomaly for elliptical orbit	radians	25, 27, 28, 29, 33, 34, 39, 40
COSI	Cosine of the inclination angle	—	6, 8, 9, 11, 12
DEJ	Eccentric anomaly correction for elliptical orbit	radians	28, 29
DFJ	Eccentric anomaly correction for hyperbolic orbit	radians	53, 54, 57
DJ	"Eccentric anomaly" for parabolic orbit	radians	71, 72, 73, 78
EN	Mean motion	radians/unit time	16, 17, 44, 46, 67, 69
FJ	"Eccentric anomaly" for hyperbolic orbit	radians	50, 52, 53, 54, 57, 58, 59, 64, 65
FK	Gravitational constant	consistent with units of VI(1) and TI	CALL, 16, 44, 67

SUBROUTINE TORB NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
ICON	Indicator which determines type of conic	—	CALL, 19, 21, 45, 68
JIX	—	—	23, 24
L	Indicator to determine if conic is a parabola L = 0, conic is not a parabola L ≠ 0, conic is a parabola	—	CALL, 14
N	Iteration counter	—	26, 30, 31, 51, 55, 56
PX	X component of unit vector P which points toward perifocus	—	8, 35, 41, 60, 74
PY	Y component of unit vector P	—	9, 36, 42, 61, 75
PZ	Z component of unit vector P	—	10, 37, 43, 62, 76
Q000FL	Cosine of the argument of of perifocus	—	2, 8, 9, 11, 12, 13
Q	Perifocus radius	c.u.	48, 49, 50
QX	X component of unit vector Q which lies in orbit plane and is advanced from P by 90°	—	11, 35, 41, 60, 74
QY	Y component of unit vector Q	—	12, 36, 42, 61, 75
QZ	Z component of unit vector Q	—	13, 37, 43, 62, 76
RAD3	—	—	49, 50, 70, 71
RATIO	(eccentricity - 1) / eccentricity	—	47, 57

SUBROUTINE TORB NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RI	Magnitude of position vector	c.u.	38, 39, 40, 63, 64, 65, 77, 78, 79
SAND	Sine of the longitude of the ascending node	—	5, 8, 9, 11, 12
SINI	Sine of the inclination angle	—	7, 10, 13
SOM	Sine of the argument of perifocus	—	4, 8, 9, 10, 11, 12
TI	Time at which to calculate position and velocity	consistent	CALL, 17, 46, 69
V	Input array containing orbital elements	—	CALL, DIM, 1
VI	Array containing orbital elements	—	DIM, 1, 2, 3, 4, 5, 6, 7, 15, 16, 17, 18, 25, 27, 28, 33, 34, 39, 40, 44, 46, 47, 48, 52, 53, 58, 59, 64, 65, 69, 70, 71, 72, 73, 79
VO	Output array containing components of position and velocity vectors	c.u.	CALL, DIM, 35, 36, 37, 38, 63, 77
XO	X component of position or velocity vector in a two-dimensional coordinate frame with the x-axis pointing along $\underline{P}$ and the y-axis pointing along $\underline{Q}$	c.u.	33, 35, 36, 37, 39, 41, 42, 43, 58, 60, 61, 62, 64, 72, 74, 75, 76, 78
YO	Y component of position or velocity vector in the two-dimensional coordinate frame	c.u.	34, 35, 36, 37, 40, 41, 42, 43, 59, 60, 61, 62, 65, 73, 74, 75, 76, 79

```
$IBFTC TORB    REF,LIST
C       COMPUTION OF POSITION AND VELOCITY VECTORS FROM A,E,T,I,NODE,OMEGA
        SUBROUTINE TORB (V ,VO,L,AMU,TI,FK,ICON)
C       INPUT
C         V(1)=A (Q FOR PARABOLA)
C         V(2)=E
C         V(3)=T
C         V(4)=I       IN RADIANS
C         V(5)=NODE    IN RADIANS
C         V(6)=OMEGA   IN RADIANS
C        L=0 CONIC IS NOT A PARABOLA
C        L=1 CONIC IS A PARABOLA--INPUT Q IN  V(1) AND SET  V(2)=1.
C        AMU = SUM OF MASSES
C        TI= TIME AT WHICH TO CALCULATE POSITION AND VELOCITY
C       OUTPUT
C        POSITION VECTOR              VELOCITY VECTOR
C        VO(1)=X                      VO(4)=X DOT
C        VO(2)=Y                      VO(5)=Y DOT
C        VO(3)=Z                      VO(6)=Z DOT
C       FK= GRAVITATIONAL CONSTANT
C       ICON=1     CIRCLE
C       ICON=2   ELLIPSE
C       ICON=3     PARABOLA
C       ICON=4    HYPERBOLA
C
        DIMENSION VI(6),VO(6)        ,V(6)
C       DIMENSION VI(6),VO(6)        ,V(6)
        DO 5 I=1,6
        VI(I)=V(I)
 5      CONTINUE
        QOOOFL=COS(VI(6))
        CAND=COS(VI(5))
        SOM =SIN(VI(6))
        SAND=SIN(VI(5))
        COSI=COS(VI(4))
        SINI=SIN(VI(4))
        PX=QOOOFL*CAND-SOM*SAND*COSI
        PY=QOOOFL*SAND+SOM*CAND*COSI
        PZ=SOM*SINI
        QX=-SOM*CAND-QOOOFL*SAND*COSI
        QY=-SOM*SAND+QOOOFL*CAND*COSI
        QZ=QOOOFL*SINI
        IF (L) 40,10,40
 10     IF (VI(1))   30,12,12
C       ELLIPSE OR CIRCLE
 12     EN=SQRT(AMU/VI(1)**3) *FK
        AM=EN*(TI-VI(3))
        IF(VI(2)       ) 122,121,122
 121    ICON=1
        GO TO 13
```

FIGURE 64. SYMBOLIC LISTING OF SUBROUTINE TORB

```
  122 ICON=2
      IF(ABS(AM)-6.28318530) 14, 13,13
  13  JIX=AM/6.28318530
      AM=AM-FLOAT(JIX)*6.28318530
  14  CEJ=AM+VI(2)*SIN(AM)+VI(2)**2*SIN(AM+AM)/2.0+VI(2)**3*.125* (3.0*S
     1IN(3.0*AM)-SIN(AM))
      N=0
  16  AMV=CEJ-VI(2)*SIN(CEJ)
      DEJ=(AM-AMV)/(1.0-VI(2)*COS(CEJ))
      CEJ=CEJ+DEJ
      N=N+1
      IF (N-15)  18, 18,50
  18  IF (ABS(AM-AMV)-1.E-7) 20,20,16
  20  XO= VI(1)*(COS(CEJ)-VI(2))
      YO=VI(1)*SQRT(1.0-VI(2)**2)*SIN(CEJ)
      VO(1)=PX*XO+QX*YO
      VO(2)=PY*XO+QY*YO
      VO(3)=PZ*XO+QZ*YO
      RI=SQRT(VO(1)**2+VO(2)**2+VO(3)**2)
      XO=-SQRT(AMU*VI(1))*SIN(CEJ)/RI
      YO= SQRT(AMU*VI(1)*(1.0-VI(2)**2))/RI*COS(CEJ)
  25  VO(4)=XO*PX+YO*QX
      VO(5)=XO*PY+YO*QY
      VO(6)=XO*PZ+YO*QZ
  27  RETURN
C     HYPERBOLA
  30  EN=SQRT(AMU/(-VI(1))**3) *FK
      ICON=4
      AM=EN*(TI-VI(3))
      RATIO=(VI(2)-1.0)/VI(2)
      Q=VI(1)*(1.0-VI(2))
      RAD3=(3.0*AM+SQRT(9.0*AM**2+8.0*Q**3)) **.333333333
      FJ=RAD3-2.0*Q/RAD3
      N=0
  32  AMO=VI(2)*(EXP(FJ)-EXP(-FJ))/2.0-FJ
      DFJ=(AM-AMO)/(VI(2)*(EXP(FJ)+EXP(-FJ))/2.0-1.0)
      FJ=DFJ+FJ
      N=N+1
      IF (N-15)    34,34,50
  34  IF (ABS(DFJ/FJ)*RATIO-.5E-6) 36, 36,32
  36  XO =VI(1)*((EXP(FJ)+EXP(-FJ))/2.0-VI(2))
      YO =-VI(1)*SQRT(VI(2)**2-1.0)*(EXP(FJ)-EXP(-FJ))/2.0
      VO(1)=XO*PX +YO*QX
      VO(2)=XO*PY +YO*QY
      VO(3)=XO*PZ +YO*QZ
      RI=SQRT(VO(1)**2+VO(2)**2+VO(3)**2)
      XO = -SQRT(-AMU*VI(1))/RI* (EXP(FJ)-EXP(-FJ))/2.0
      YO = SQRT(AMU*VI(1)*(1.0-VI(2)**2))/RI *(EXP(FJ)+EXP(-FJ))/2.0
      GO TO 25
C     PARABOLA
```

FIGURE 64. SYMBOLIC LISTING OF SUBROUTINE TORB (cont.)

```
   40  EN=SQRT(AMU) *FK
       ICON=3
       AM=EN*(TI-VI(3))
       RAD3= (3.0*AM+SQRT(9.0*AM**2+8.0*VI(1)**3))**.333333333
       DJ=RAD3-2.0*VI(1)/RAD3
       XO = VI(1)-.5*DJ**2
       YO = DJ*SQRT(2.0*VI(1))
       VO(1)=XO*PX+YO*QX
       VO(2)=XO*PY+YO*QY
       VO(3)=XO*PZ+YO*QZ
       RI =SQRT(VO(1)**2+VO(2)**2+VO(3)**2)
       XO = -DJ/RI
       YO = SQRT(2.0*VI(1))/RI
       GO TO 25
C         MAX ITERATIONS
   50  CALL SETERR(4HTORB)
       RETURN
       END
```

FIGURE 64. SYMBOLIC LISTING OF SUBROUTINE TORB (cont.)

LI. SUBROUTINE VITER

This subroutine computes an orbit from two observations of position and time, and computes the velocity in orbit at the first position.

CALLING SEQUENCE

CALL VITER (RI, RO, TI, CON, L)

RI(1) = X1
RI(2) = Y1
RI(3) = Z1
RI(4) = X2
RI(5) = Y2
RI(6) = Z2

RO(1) = $\dot{X}1$
RO(2) = $\dot{Y}1$
RO(3) = $\dot{Z}1$
RO(4) = $\dot{X}2$
RO(5) = $\dot{Y}2$
RO(6) = $\dot{Z}2$

TI(1) = T1

TI(2) = T2

CON(1) = k (gravitational constant)

CON(2) = μ (sum of masses)

CON(3) = tolerance on iteration

L = 1 for direct motion

L = 2 for retrograde motion

VITER requires Subroutines QUAD and DTPRD

SOLUTION METHOD

Define arithmetic functions:

COSH(x) = 0.5 (Exp(x) + Exp(-x))

SINH(x) = 0.5 (Exp(x) - Exp(-x))

Set iteration counter:

1. LZ = 0

Define constant equal to square root of mass sum:

2. SQMU = SQRT(CON(2))

Find time difference:

3. DT = TI(2) - TI(1)

Scale time with gravitational constant:

4. T = (CON(1))(DT)

```
5.  Do 6, K = 1, 3
```

Get separate array for second position:

```
6.  RF(K) = RI(K + 3)
```

Find magnitudes of position vectors:

```
7.  CALL DTPRD(RI, RI, R1)
```

DTPRD finds the dot product of two vectors:

```
8.  R1 = SQRT(R1)

9.  CALL DTPRD(RF, RF, R2)

10. R2 = SQRT(R2)
```

Find cosine of angle between positions:

```
11. CALL DTPRD (RI, RF, CV2V1)
```

Find sine of angle between positions:

```
12. CV2V1 = (CV2V1)/((R1)(R2))

13. SV2V1 = SIGN (SQRT(1 - ((CV2V1)(CV2V1))),
                  ((RI(1))(RF(2)) - ((RI(2))(RF(1)))
```

SIGN is a FORTRAN function. The use of SIGN(A, B) assigns the algebraic sign of B to the magnitude A.

Reverse sign of angle for retrograde motion:

14. If $L \neq 1$, $SV2V1 = -SV2V1$

If $L = 1$, go to 15

Find angle between positions:

15. CALL QUAD (CV2V1, SV2V1, V2V1)

QUAD is a subroutine for computing an angle given its cosine and sine.

Test for straight angle between positions:

16. If $| V2V1 - 3.1415927 | > .00005$, go to 19

If $| V2V1 - 3.1415927 | \leq .00005$, go to 17

Indeterminate solution.

17. CALL SETERR (6HVITER1)

18. Go to 100

Start iteration to find anomalies, arbitrarily setting:

19. $V1 = 0$

Set indicator at 1 until physically feasible anomaly is found:

20. $NV = 1$

Compute anomaly at second position:

21. $V2 = V1 + V2V1$

Define cosine of V1:

```
22.  COSV1 = Cos(V1)
```

Define cosine of V2:

```
23.  COSV2 = Cos(V2)
```

Compute eccentricity of orbit:

```
24.  E = (R2 - R1)/(((R1)(COSV1)) - ((R2)(COSV2)))
```

Check tolerances:

```
25.  If |E| < .000001, E = 0
     If |E| ≥ .000001, go to 26
26.  If E ≥ 0, go to 29
     If E < 0, go to 27
```

Make new guess to find feasible anomaly:

```
27.  V1 = V1 + 1
```

Loop back:

```
28.  Go to 21
```

Compute semi-major axis: First, define

```
29.  ECV11 = 1 + ((E)(COSV1))
30.  E12 = 1 - ((E)(E))
```

31. ECV21 = 1 + ((E)(COSV2))

Get semi-major axis:

32. A = (R1)(ECV11)/(E12)

Check for physical compatibility of A and E:

33. If $E < 1$ and $A < 0$, or if $E > 1$ and $A > 0$, go to 27.

If the motion is not feasible, loop back for new estimate of V1.

Otherwise, go to 34

34. If $A < 0$, go to 56 (hyperbola)

If $A \geq 0$, go to 35 (nonhyperbolic)

Check tolerances:

35. If $|E - 1| < .000001$, $E = 1$

If $|E - 1| \geq .000001$, go to 36

36. If $E = 1$, go to 49 (parabolic)

If $E \neq 1$, go to 37

37. If $E > 1$, go to 27

(physically unreal, loop back)

If $E \leq 1$, go to 38

Elliptical or circular orbits:

Compute sines and cosines of eccentric anomalies.

38. SE1 = (SQRT(E12))(Sin(V1))/(ECV11)

39. CE1 = (COSV1 + E)/(ECV11)

40. SE2 = (SQRT(E12))(Sin(V2))/(ECV21)

41. CE2 = (COSV2 + E)/(ECV21)

Find eccentric anomalies:

42. CALL QUAD (CE1, SE1, E1)

43. CALL QUAD (CE2, SE2, E2)

Set difference positive:

44. If E2 - E1 < 0, E2 = E2 + 6.28318531

If E2 - E1 ≥ 0, go to 45

Define difference in eccentric anomalies:

45. E2E1 = E2 - E1

Compute time error from Kepler's equation:

46. F = T - ((E2E1 + ((E)(Sin(E1) - Sin(E2)))$(A^{1.5})$)/(SQMU)

Set flag for circular or elliptic orbit:

47. KON = 1

48. If NV = 1, go to 63

(physically feasible region found, arbitrarily perturb anomaly estimates first time through)

If NV = 2, go to 68

(get next successive approximation)

Parabolic case:

Compute semi-latus rectum.

49. P = (R1)(1 + ((E)(COSV1)))

50. SQRTP = SQRT(P)

Define auxiliary variables:

51. D1 = (SQRTP)(Tan((V1)/(2)))

52. D2 = (SQRTP)(Tan((V2)/(2)))

Compute time error in parabolic case:

53. $F = T - (((.5)(P)(D2 - D1)) - ((D2^3 - D1^3)/(6)))/(SQMU)$

Set flag for parabola:

54. KON = 2

Continue successive approximations:

55. Go to 48

Hyperbolic case:

Define auxiliary variables.

56. SINHF1 = (SQRT(E^{2} - 1))(SIN(V1))/(1 + ((E)(COSV1)))

57. SINHF2 = (SQRT(E^{2} - 1))(SIN(V2))/(1 + ((E)(COSV2)))

58. FF1 = ALOG(SINHF1 + SQRT(1 + SINHF1^{2}))

59. FF2 = ALOG(SINHF2 + SQRT(1 + SINHF2^{2}))

Find error in time of flight from hyperbolic form of Kepler's equation:

60. F = T - (((E)(SINHF2 - SINHF1) + FF1 - FF2)(-A)$^{1.5}$)/(SQMU)

Set flag for hyperbola:

61. KON = 3

Continue successive approximations:

62. Go to 48

Successive approximations to true anomaly, temporarily store initial estimate:

63. V1I = V1

The first time through, arbitrarily perturb anomaly:

64. V1 = V1 + (.17453292)(10^{-1})

Set flag for succeeding passes:

65. NV = 2

Store time error:

66. F1 = F

Loop back to compute time error with revised estimate of true anomaly:

67. Go to 21

Step counter:

68. LZ = LZ + 1

69. If LZ ≠ 30, go to 72 (next approximation)

If LZ = 30, go to 70

Divergent solution.

70. CALL SETERR (6HVITER2)

71. Go to 100

Get numerical derivative of time with respect to anomaly:

72. FP = (F - F1)/(V1 - V1I)

Get approximate anomaly change corresponding to time error:

73. DV1 = (F)/(FP)

Store last estimate of anomaly:

74. V1I = V1

Correct anomaly:

75. V1 = V1 - DV1

Store last time error:

76. F1 = F

Is anomaly change within tolerance?

77. If $|DV1| >$ CON(3), go to 21 (No: get next approximation)

If $|DV1| \leq$ CON(3), go to 78 (Yes: proceed to velocity computation)

78. If KON = 1, go to 87

If KON = 2, go to 79

If KON = 3, go to 82

F and G coefficients for velocities.

Velocities in parabolic orbit:

79. $F = ((0.5)(P - D2^2) + (D2)(D1))/(R1)$

80. $G = (0.5)((P - (D2)(D1))(D2 + D1))/SQMU$

81. $FV = (SQMU)(D2 - D1)/((R1)(R2))$

82. $GV = ((0.5)(P - D1^2) - (D1)(D2))/R2$

83. Go to 96

Hyperbolic case:

Get corresponding F and G coefficients.

```
84.  A = -A
85.  F2F1 = FF2 - FF1
86.  F = 1 - ((A)/(R1))(COSH(F2F1) - 1)
87.  G = T - (($A^{1.5}$)/((SQMU)(F2F1 - SINH(F2F1))))
88.  FV = ((SQMU)$(A)^{.5}$ SINH(F2F1))/((R1)(R2))
89.  GV = 1. - (A/R2)(COSH(F2F1) - 1.)
90.  Go to 96
91.  Continue
```

F and G coefficients for elliptic orbits:

```
92.  F = 1 - ((A)/(R1))(1 - Cos(E2E1))
93.  G = T - ((($A^{1.5}$)/(SQMU))(E2E1 - Sin(E2E1)))
94.  FV = -((SQMU)$(A)^{.5}$ Sin(E2E1))/((R1)(R2))
95.  GV = 1. - (A/R2)(1. - Cos(E2E1)
```

Get output positions and velocities:

```
96.  Do 99, K = 1,3
97.  RO(K) = (RF(K) - (F)(RI(K)))/(G)
98.  RO(K + 3) = (FV)(RI(K)) + (GV)(RO(K))
99.  Continue
100. Return
```

SUBROUTINE VITER NOMENCLATURE

Symbol	Description	Units	Reference
A	Semi-major axis of transfer ellipse	c.u.	32, 33, 34, 46, 60, 84, 86, 87, 88, 89, 92, 93, 94, 95
CE1	Cosine of eccentric anomaly at first position	—	39, 42
CE2	Cosine of eccentric anomaly at second position	—	41, 43
CON	Array of input constants (see comments on calling sequence)	c.u.	DIM, CALL, 2, 4, 77
COSV1	Cosine of true anomaly at first position	—	22, 24, 39, 49, 57
COSV2	Cosine of true anomaly at second position	—	23, 24, 31, 41, 57
CV2V1	Cosine of true anomaly difference	—	11, 12, 13, 15, 29
D1	Auxiliary variable for parabolic motion	—	51, 53, 79, 80, 81, 82
D2	Auxiliary variable for parabolic motion	—	52, 53, 79, 80, 81, 82
DT	Time of free flight	min (for earth) days (for sun)	3, 4
DV1	Differential correction to true anomaly	radian	73, 75, 77
E	Eccentricity of transfer orbit	—	24, 25, 26, 29, 30, 31, 33, 35, 36, 37, 39, 41, 46, 49, 56, 57, 60

SUBROUTINE VITER NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
E1	Eccentric anomaly at first position	radian	42, 44, 45, 46
E2	Eccentric anomaly at second position	radian	43, 44, 45, 46
E12	Auxiliary variable $(1 - E^2)$	—	30, 32, 38, 40
E2E1	Eccentric anomaly difference	radian	45, 46, 88, 89
ECV11	Auxiliary variable (1 + E Cos V1)	—	29, 32, 38, 39
F	Time error	c.u.	46, 53, 60, 66, 72, 73, 76
F	f coefficient of fg series	—	79, 86, 92, 97
F1	Previous value of time error	c.u.	66, 72, 76
F2F1	Difference in hyperbolic anomalies	—	85, 86, 87, 88, 89
FF1	Auxiliary variable	—	58, 60, 85
FF2	Auxiliary variable	—	59, 60, 85
FV	Coefficient in fg velocity series	—	81, 88, 94, 98
FP	Numerical derivative of time with respect to anomaly	c.u.	72, 73
G	g coefficient in fg series	—	80, 87, 93, 97
GV	Coefficient in fg velocity series	—	82, 89, 95, 98

SUBROUTINE VITER NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
KON	Indicator for type of conic = 1 for ellipse = 2 for parabola = 3 for hyperbola	—	47, 54, 61, 78
L	Indicator for type of motion = 1 for direct motion = 2 for retrograde motion	—	CALL, 14
LZ	Iteration counter	—	1, 68, 69
NV	Flag indicator = 1 for arbitrary correction of true anomaly = 2 for computed corrections on successive approximations	—	20, 48, 65
P	Semi-latus rectum	c.u.	49, 50, 53, 72, 79, 80, 82
R1	First position magnitude	c.u.	7, 8, 12, 24, 49, 72, 79, 81, 86, 88, 94
R2	Second position magnitude	c.u.	9, 10, 12, 24, 82, 88, 89, 94, 95
RF	Array for storing second position	c.u.	6, 9, 11, 13
RI	Input array RI(1), RI(2), RI(3) first position RI(4), RI(5), RI(6) second position	c.u.	DIM, CALL, 6, 7, 11, 13, 97, 98

SUBROUTINE VITER NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RO	Output array RO(1), RO(2), RO(3) first velocity RO(4), RO(5), RO(6) second velocity	c.u.	DIM, CALL, 97, 98
SE1	Sine of eccentric anomaly at first position	—	38, 42
SE2	Sine of eccentric anomaly at second position	—	40, 43
SINHF1	Hyperbolic sine of parameter at first position	—	56, 58, 60
SINHF2	Hyperbolic sine of parameter at second position	—	57, 59, 60
SQMU	Square root of sum of masses	c.u.	2, 46, 53, 60, 80, 81, 87, 88
SQRTP	Square root of semi-latus rectum for parabolic motion	c.u.	50, 51, 52
SV2V1	Sine of transfer angle	—	13, 14, 15
T	Transfer time	c.u.	4, 46, 53, 60, 87, 93
TI	TI(1) = first time TI(2) = second time	min (earth) or days (sun)	DIM, CALL, 3
V1	True anomaly at first position	radian	19, 21, 22, 27, 38, 51, 56, 63, 64, 72, 74, 75
V1I	Previous value of true anomaly at first position	radian	63, 72, 74

SUBROUTINE VITER NOMENCLATURE (cont.)

Symbol	Description	Units	Description
V2	True anomaly at second position	radian	21, 23, 40, 52, 57
V2V1	Transfer angle	radian	15, 16, 21

```
$IBFTC VITER   FULIST,REF
C     ØRBIT DETERMINATIØN SUBRØUTINE FØR ITERATIØN ØN TRUE ANØMALY
      SUBRØUTINE VITER(RI,RØ,TI,CØN,L)
      DIMENSIØN RI(6),RØ(6),TI(2),CØN(3)
      DIMENSIØN RF(3)
C     DIMENSIØN RI(6),RØ(6),TI(2),CØN(3)
C     RI(1)=X1     RØ(1)=X DØT 1     TI(1)=T1     CØN(1)=K(GRAVITATIØNAL
C     RI(2)=Y1     RØ(2)=Y DØT 1     TI(2)=T2              CØNSTANT)
C     RI(3)=Z1     RØ(3)=Z DØT 1                  CØN(2)=MU (SUM ØF MASSES)
C     RI(4)=X2     RØ(4)=X DØT 2                  CØN(3)=TØLERANCE ØN
C     RI(5)=Y2     RØ(5)=Y DØT 2                           ITERATIØN
C     RI(6)=Z2     RØ(6)=Z DØT 2
C
C     L=1 (DIRECT MØTIØN) ,L=2 (RETRØGRADE MØTIØN)
C     USES SUBRØUTINE QUAD--ØBTAINS X BETWEEN 0 AND 360 DEG. FRØM
C                                 CØS(X) AND SIN(X)
C
      CØSH(X)=0.5*(EXP(X)+EXP(-X))
      SINH(X)=0.5*(EXP(X)-EXP(-X))
      LZ=0
      SQMU=SQRT(CØN(2))
      DT= TI(2)-TI(1)
      T= CØN(1)*DT
      DØ 1 K=1,3
    1 RF(K)=RI(K+3)
      CALL DTPRD(RI,RI,R1)
      R1=SQRT(R1)
      CALL DTPRD(RF,RF,R2)
      R2=SQRT(R2)
      CALL DTPRD(RI,RF,CV2V1)
      CV2V1=CV2V1/R1/R2
      SV2V1=SIGN(SQRT(1.-CV2V1*CV2V1),RI(1)*RF(2)-RI(2)*RF(1))
      IF (L .NE. 1)  SV2V1 = -SV2V1
   12 CALL QUAD(CV2V1,SV2V1,V2V1)
      IF (ABS(V2V1 - 3.1415927) .GT. .00005)  GØ TØ 14
C         INDETERMINATE SØLUTIØN
   13 CALL SETERR(6HVITER1)
      GØ TØ 66
 14   V1=0.0
 15   NV=1
 16   V2=V1+V2V1
      CØSV1=CØS(V1)
      CØSV2=CØS(V2)
      E=(R2-R1)/(R1*CØSV1-R2*CØSV2)
      IF (ABS(E) .LT. .000001)  E = 0.0
      IF (E .GE. 0.0)  GØ TØ 19
   18 V1=V1+1.
      GØ TØ 16
 19   ECV11=1.0+E*CØSV1
      E12= 1.0-E*E
```

FIGURE 65. SYMBOLIC LISTING OF SUBROUTINE VITER

```
      ECV21=1.0+E*CØSV2
      A=R1*ECV11/E12
      IF((E.LT.1..AND.A.LT.0.).ØR.(E.GT.1..AND.A.GT.0.))GØ TØ 18
      IF (A .LT. 0.0)  GØ TØ 200
      IF (ABS(E - 1.0) .LT. .000001)  E = 1.0
      IF (E .EQ. 1.0)  GØ TØ 300
      IF (E .GT. 1.0)  GØ TØ 18
C         ELLIPTICAL ØR CIRCULAR ØRBIT
   21 SE1=SQRT(E12)*SIN(V1)/ECV11
      CE1= (CØSV1+E)/ECV11
      SE2= SQRT( E12)*SIN(V2)/ECV21
      CE2= (CØSV2+E)/ECV21
      CALL QUAD (CE1,SE1,E1)
      CALL QUAD(CE2,SE2,E2)
      IF ((E2 - E1) .LT. 0.0)  E2 = E2 + 6.28318531
  212 E2E1=E2-E1
  215 F = T- (E2E1 +E*(SIN(E1)-SIN(E2))) *A**1.5/SQMU
      KØN=1
  400 GØ TØ (22,30), NV
C         PARABØLIC ØRBIT
  300 P = R1 * (1.0 + E * CØSV1)
      SQRTP = SQRT(P)
      D1 = SQRTP * TAN(V1/2.0)
      D2 = SQRTP * TAN(V2/2.0)
      F= T-(.5*P*(D2-D1)-(D2**3-D1**3)/6.)/SQMU
      KØN=2
      GØ TØ 400
C         HYPERBØLIC ØRBITS
  200 SINHF1 = SQRT(E**2-1.)*SIN(V1)/(1.+E*CØSV1)
      SINHF2 = SQRT(E**2-1.)*SIN(V2)/(1.+E*CØSV2)
      FF1 = ALØG(SINHF1 + SQRT(1.0 + SINHF1**2))
      FF2 = ALØG(SINHF2 + SQRT(1.0 + SINHF2**2))
      F = T - (E * (SINHF2 - SINHF1) + FF1 - FF2) * (-A)**1.5/SQMU
      KØN=3
      GØ TØ 400
 22   V1I=V1
      V1 = V1 + .17453292E-1
      NV=2
      F1=F
      GØ TØ 16
 30   LZ=LZ+1
      IF (LZ .NE. 30)  GØ TØ 33
C        DIVERGENT SØLUTIØN
      CALL SETERR(6HVITER2)
      GØ TØ 66
   33 FP = (F - F1)/(V1 - V1I)
      DV1 = F/FP
      V1I = V1
      V1 = V1 - DV1
      F1 = F
```

FIGURE 65. SYMBOLIC LISTING OF SUBROUTINE VITER (cont.)

```
      IF (ABS(DV1) .GT. CØN(3)) GØ TØ 16
C
      GØ TØ (2,3,4),KØN
    3 F=(0.5*(P-D2**2)+D2*D1)/R1
      G=0.5*((P-D2*D1)*(D2+D1))/SQMU
      FV= SQMU*(D2-D1)/(R1*R2)
      GV=(.5*(P-D1**2)-D1*D2)/R2
      GØ TØ 5
    4 A=-A
      F2F1=FF2-FF1
      F=1.-A/R1*(CØSH(F2F1)-1.)
      G=T-(A**1.5/SQMU*(F2F1-SINH(F2F1)))
      FV= SQMU*SQRT(A)*SINH(F2F1)/(R1*R2)
      GV= 1.- A/R2*(CØSH(F2F1)-1.)
      GØ TØ 5
    2 CØNTINUE
      F = 1.0 - A/R1 * (1.0 - CØS(E2E1))
      G = T - (A**1.5/SQMU * (E2E1 - SIN(E2E1)))
      FV= -SQMU*SQRT(A)*SIN(E2E1)/(R1*R2)
      GV= 1.- A/R2*(1.-CØS(E2E1))
    5 DØ 60 K=1,3
      RØ(K)=(RF(K)-F*RI(K))/G
      RØ(K+3)= FV*RI(K)+ GV*RØ(K)
 60   CØNTINUE
 66   RETURN
      END
```

FIGURE 65. SYMBOLIC LISTING OF SUBROUTINE VITER (cont.)

LII. SUBROUTINE CRPRD

This subroutine computes the component terms of the cross product of two vectors, such that $\underline{V_c} = \underline{V_a} \times \underline{V_b}$.

CALLING SEQUENCE

The calling sequence is:

CALL CRPRD (A, B, C)

A and B are arrays containing, for each of the two vectors $\underline{V_a}$ and $\underline{V_b}$ respectively, the magnitudes of the projections of the vectors along the coordinate Ox, Oy and Oz axes.

SOLUTION METHOD

The appropriate equations are:

1. C(1) = (A(2))(B(3)) - (A(3))(B(2))
2. C(2) = (A(3))(B(1)) - (A(1))(B(3))
3. C(3) = (A(1))(B(2)) - (A(2))(B(1))

Return

SUBROUTINE CRPRD NOMENCLATURE

Symbol	Description	Units	Reference
A	Table of first vector Ox, Oy and Oz coordinates	—	CALL, DIM, 1, 2, 3
B	Table of second vector Ox, Oy and Oz coordinates	—	CALL, DIM, 1, 2, 3
C	Components of cross-product terms	—	CALL, DIM, 1, 2, 3

```
$IBFTC CRPRD   REF,LIST
      SUBROUTINE CRPRD (A,B,C)
      DIMENSION A(3),B(3),C(3)
C     DIMENSION A(3),B(3),C(3)
      C(1)= A(2)*B(3)-A(3)*B(2)
      C(2)= A(3)*B(1) - A(1)*B(3)
      C(3) = A(1)*B(2) - A(2)*B(1)
      RETURN
      END
```

FIGURE 66. SYMBOLIC LISTING OF SUBROUTINE CRPRD

LIII. SUBROUTINE DTPRD

This subroutine determines the dot product of two vectors.

CALLING SEQUENCE

The calling sequence is:

CALL DTPRD (A, B, C)

A and B are arrays containing, for each of the two vectors $\underline{V_a}$ and $\underline{V_b}$ respectively, the magnitudes of the projections of the vectors along the coordinate Ox, Oy and Oz axes.

SOLUTION METHOD

The appropriate equation is:

1. C = (A(1))(B(1)) + (A(2))(B(2)) + (A(3))(B(3))

   Return

SUBROUTINE DTPRD NOMENCLATURE

Symbol	Description	Units	Reference
A	Table of first vector Ox, Oy and Oz coordinates	—	CALL, DIM, 1
B	Table of second vector Ox, Oy and Oz coordinates	—	CALL, DIM, 1
C	Dot product	—	CALL, 1

```
$IBFTC DTPRD   REF,FULIST
C          SUBROUTINE TO FIND THE DOT PRODUCT OF 2 VECTORS
      SUBROUTINE DTPRD(A,B,C)
      DIMENSION A(3),B(3)
      C = A(1)*B(1) + A(2)*B(2) + A(3)*B(3)
      RETURN
      END
```

FIGURE 67. SYMBOLIC LISTING OF SUBROUTINE DTPRD

LIV. SUBROUTINE MOON

This subroutine provides lunar ephemeris data as given in Battin, Astronautical Guidance, pages 379-383. This is a truncated form of the Brown lunar theory and is accurate to about three minutes of arc. The input is a double-precision Julian date, and the outputs are a unit vector pointing to the moon in a geocentric equatorial coordinate system and the distance to the moon in earth radii.

CALLING SEQUENCE

The calling sequence is:

CALL MOON (DATE, SEL, RSEL)

SOLUTION METHOD

Define a function for expressing in radians an angle given as a time series with coefficients given in revolutions, degrees, minutes, and seconds.

1. ANG(A, B, C, D, E) = AMOD(AMOD((B)(TC), AINT((B)(TC)))(TWOPI) + AMOD(A + (TC)(C + (TC)(D + (TC)(E))), 1 296 000) (RPS), TWOPI)

Find time in Julian centuries after epoch Jan 0., 1900 as argument of time series.

2. TC = (DATE - 2 415 020. D0) (.27378507E - 4)

Evaluate geocentric mean longitude of moon.

3. ELM = ANG(973 571.71, 1336, 1 108 406.06, 7.14, .0068)

Evaluate difference between longitude of moon and sun.

4. A = ANG(1 262 663.67, 1236, 1 105 637.93, 6.05, .0068)

Evaluate difference between lunar longitude and longitude of perigee.

5. B = ANG(1 065 985.31, 1324, 715 883.54, 44.31, .0518)

Evaluate difference between longitude of sun and its perigee.

6. C = ANG(1 290 513, 99, 1 292 579.10, -.54, -.012)

Evaluate difference between lunar longitude and node.

7. D = ANG(40 511.92, 1342, 295 317.29, -.34, -.0012)

Lunar longitude, latitude, and parallax are expressed in a trigonometric series involving algebraic sums of simple multiples of A, B, C, and D.

Evaluate true longitude in the plane of the ecliptic.

8. SLONG = ELM + ((22 639.5)(Sin(B)) - (4586.426)
(Sin (B - (2)(A)))) + (2369.902)(Sin((2)(A)))
+ (769.016)(Sin((2)(B))) - (668.111)(Sin(C))
- (411.608)(Sin((2)(D))) - (211.656)
(Sin((2)(B - A))) - (205.962)(Sin(B + C
- (2)(A))) - (125.154)(Sin(A)) + (191.953)
(Sin(B + (2)(A))) - (165.145)(Sin(C - (2)(A)))
+ (147.693)(Sin(B - C)) - (109.667)
(Sin(B + C))(RPS)

Evaluate latitude above the plane of the ecliptic.

9. SLAT = ((18461.48)(Sin(D)) + (1010.18)(Sin(B+D))
- (999.695)(Sin(D - B)) - (623.658)(Sin(D
- (2)(A))) + (117.262)(Sin(D + (2)(A)))
+ (199.485)(Sin(D + (2)(A) - B)) - (166.577)
(Sin(B + D - (2)(A))) + (61.913)(Sin((2)(B)
+ D))(RPS)

Evaluate the "horizontal parallax" (half the angular diameter of the earth as seen from the moon).

10. SPAR = (3422.7 + (186.5398)(Cos(B)) + (34.3117)
(Cos(B - (2)(A))) + (28.2333)(Cos((2)(A)))
+ (10.1657)(Cos((2)(B))) + (3.0861)(Cos(B
+ (2)(A))) + (1.9202)(Cos(C - (2)(A)))
+ (1.4455)(Cos(B + C - (2)(A))) + (1.1542)
(Cos(B - C)))(RPS)

Find obliquity of the ecliptic in radians.

11. E = (ECLIP)(PI)/(180)

Find sine of obliquity of the ecliptic.

12. SE = Sin(E)

Find cosine of obliquity of the ecliptic.

13. CE = Cos(E)

Find y component of unit vector in ecliptic system.

14. Y = (Cos(SLAT))(Sin(SLONG))

Find z component of unit vector in ecliptic system.

15. Z = Sin(SLAT)

Find x component of unit vector (same in ecliptic and equatorial systems).

16. SEL(1) = (Cos(SLAT))(Cos(SLONG))

Change y coordinate to equatorial system.

17. SEL(2) = (Y)(CE) - (Z)(SE)

Change z coordinate to equatorial system.

18. SEL(3) = (Y)(SE) + (Z)(CE)

Compute distance to moon in earth radii.

19. RSEL = (1)/(Sin(SPAR))

20. Return

SUBROUTINE MOON NOMENCLATURE

Symbol	Description	Units	Reference
ANG	Function to evaluate functional form of time series	radians	1, 3, 4, 5, 6, 7
A	Difference between longitude of moon and sun	radians	1, 4, 8, 9, 10
B	Difference between longitude of moon and perigee	radians	1, 5, 8, 9, 10
CE	Cosine of obliquity of ecliptic	—	13, 17, 18
C	Difference between longitude of sun and its perigee	radians	1, 6, 8, 10
D	Difference between lunar longitude and node	radians	1, 7, 8, 9
DATE	Julian date	days	CALL, DP, 2
ECLIP	Obliquity of ecliptic	degrees	DATA, 11
ELM	Mean geocentric lunar longitude	radians	3, 8
E	Obliquity of ecliptic	radians	1, 11, 12, 13
PI	3.1415927	—	DATA, 11
RPS	Radians per second of arc = $(0.48481368)(10^{-5})$	—	DATA, 1, 8, 9, 10
RSEL	Distance to moon	earth radii	CALL, 19
SE	Sine of obliquity of ecliptic	—	12, 17, 18

SUBROUTINE MOON NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
SEL	Unit vector in direction of moon in geocentric equatorial coordinate system	length	CALL, DIM, 16, 17, 18
SLAT	Latitude of moon above ecliptic	radians	9, 14, 15, 16
SLONG	Longitude of moon in ecliptic	radians	8, 14, 16
SPAR	Horizontal parallax	radians	10, 19
TC	Time since epoch Jan 0., 1900	Julian centuries	1, 2
TWOPI	(2.)(PI)	—	DATA, 1
Y	y coordinate of unit vector in ecliptic system	—	14, 17, 18
Z	z coordinate of unit vector in ecliptic system	—	15, 17, 18

```
$IBFTC MØØN    FULIST,REF
      SUBRØUTINE MØØN(DATE,SEL,RSEL)
      DØUBLE PRECISIØN DATE
      DIMENSIØN SEL(3)
      DATA RPS,TWØPI/0.48481368E-5,6.2831853/
      DATA ECLIP/23.4497/,PI/3.1415927/
      ANG(A,B,C,D,E)=AMØD(AMØD((B*TC),AINT(B*TC))*TWØPI+AMØD(A+TC*(C+
     1 TC*(D+TC*E)),1296000.)*RPS ,TWØPI)
      TC=(DATE-2415020.D0)*0.27378507E-4
      ELM=ANG(973571.71,1336.,1108406.06,7.14,.0068)
      A=ANG(1262663.67,1236.,1105637.93,6.05,.0068)
      B=ANG(1065985.31,1325.,715883.54,44.31,.0518)
      C=ANG(1290513.00,  99.,1292579.10,-.54,-.012)
      D=ANG( 40511.92 ,1342.,295317.29,-.34,-.0012)
      SLØNG=ELM+(22639.5*SIN(B)-4586.426*SIN(B-2.*A)+2369.902*SIN(2.*A)
     1+769.016*SIN(2.*B)-668.111*SIN(C)-411.608*SIN(2.*D)-211.656*SIN(2.
     2*(B-A))-205.962*SIN(B+C-2.*A)-125.154*SIN(A)+191.953*SIN(B+2.*A)-
     3 165.145*SIN(C-2.*A)+147.693*SIN(B-C)-109.667*SIN(B+C))*RPS
      SLAT=(18461.48*SIN(D)+1010.18*SIN(B+D)-999.695*SIN(D-B)-623.658*
     1 SIN(D-2.*A)+117.262*SIN(D+2.*A)+199.485*SIN(D+2.*A-B)-166.577*
     2 SIN(B+D-2.*A)+61.913*SIN(2.*B+D))*RPS
      SPAR=(3422.7+186.5398*CØS(B)+34.3117*CØS(B-2.*A)+28.2333*CØS(2.*A)
     1 +10.1657*CØS(2.*B)+3.0861*CØS(B+2.*A)+1.9202*CØS(C-2.*A)+1.4455*
     2 CØS(B+C-2.*A)+1.1542*CØS(B-C))*RPS
      E=ECLIP*PI/180.
      SE=SIN(E)
      CE=CØS(E)
      Y=CØS(SLAT)*SIN(SLØNG)
      Z=SIN(SLAT)
      SEL(1)=CØS(SLAT)*CØS(SLØNG)
      SEL(2)= Y*CE- Z*SE
      SEL(3)= Y*SE+ Z*CE
      RSEL= 1./SIN(SPAR)
      RETURN
      END
```

FIGURE 68. SYMBOLIC LISTING OF SUBROUTINE MOON

LV. SUBROUTINE LUNAR

This subroutine computes the angle required for a transfer from a circular parking orbit to lunar impact. The angle is measured from perigee of the elliptical transfer orbit. The required inputs are launch date, flight time in the transfer orbit from injection to impact (hours), and altitude of the circular parking orbit (feet). Outputs are declination and right ascension of the vector to the moon at impact, the transfer angle, and the velocity magnitudes in the parking orbits and at injection.

CALLING SEQUENCE

The calling sequence is:

CALL LUNAR (LDATE, DELT, RL)

SOLUTION METHOD

Set iteration counter.

1. $K = 0$

Set iteration tolerance.

2. $N = 6$

Convert transfer time to minutes.

3. TFL = (DELT)(60)

Compute Julian date at impact.

4. TJD = LDATE + (DELT)/(24)

Get direction cosines and distance of moon at impact.

5. CALL MOON (TJD, SEL, RT)

Get declination and right ascension of moon at impact.

6. PLAT = ASIN(SEL(3))

   (ASIN is the inverse sine function.)

Define equatorial projection of unit vector to moon.

7. D = $(\mathrm{SEL}(1)^2 + \mathrm{SEL}(2)^2)^{.5}$

Find right ascension.

8. CRA = (SEL(1))/(D)
9. SRA = (SEL(2))/(D)
10. CALL QUAD (CRA, SRA, PSIDT)

Compute parabolic transfer time, which is the elliptic limit, to see if elliptic transfer is possible.

11. TLIM = $(0.33333333)(\mathrm{CON})((2)(\mathrm{RT} - \mathrm{RL}))^{.5}(\mathrm{RT} + (2)(\mathrm{RL}))$

12. If TFL ≤ TLIM, go to 39 (error exit)

If TFL > TLIM, go to 13

Arbitrary initial guess at transfer angle.

13. CTA = 3

Test for consistency of transfer angle with elliptic orbit.

14. If (1 + Cos (CTA)) < ((2)(RL)/(RT)), go to 17

If (1 + Cos (CTA)) ≥ ((2)(RL)/(RT)), go to 15

Re-estimate transfer angle in feasible region.

15. CTA = (.5)(CTA + PI)

Loop back for test of feasibility.

16. Go to 14

Compute semimajor axis of estimated transfer orbit.

17. A = (RL)(RL - (RT)(Cos (CTA)))/((2)(RL) - (RT)
(1 + Cos (CTA)))

Compute chord between injection and impact.

18. CC = (RL)(RL) + (RT)(RT) - (2)(RL)(RT)(Cos (CTA))

19. C = $CC^{.5}$

Define semiperimeter of triangle.

20. $S = (.5)(RL + RT + C)$

Define auxiliary variables.

21. $ALP = (2)(PIBY2 - ACOS(((.5)(S)/(A))^{.5}))$

22. $BET = (2)(PIBY2 - ACOS(((.5)(S - C)/(A))^{.5}))$

Compute error in time of flight.

23. $ERR = (A^{3/2})(CON)(PI + SIGN(1, Sin(CTA))((ALP - Sin(ALP) - PI) - (BET - Sin(BET)))) - TFL$

Check tolerance.

24. If $|(ERR)/(TFL)| < 10^{-N}$, go to 36

If $|(ERR)/(TFL)| \geq 10^{-N}$, go to 25

Step iteration counter.

25. $K = K + 1$

Check iteration count.

26. If $K \geq 10$, go to 41

If $K < 10$, go to 27

Check special case of straight angle.

```
27. If CTA = PI, go to 31
    If CTA ≠ PI, go to 28
```

Compute derivatives for Newton-Raphson.

```
28. DSDCT = (.5)(RL)(RT)(Sin(STA))/(C)
29. DADCT = (-2)(RT)(A - RL)(Sin(CTA)/((RL)(RL) + CC - (RT)(RT))
30. Go to 33
31. DTDCT = $((.5)(RT)^3 (RL + RT)/(RL))^{.5}$ (CON)
32. Go to 34
33. DTDCT = (1.5)(TFL)(DADCT) + $(A)^{.5}$ (CON) (SIGN(1, Sin(CTA))(Tan((.5)(ALP)) (DSDCT - (S)(DADCT)) + Tan((.5)(BET))) (DSDCT + (S - C)(DADCT)))
```

Get next approximation for transfer angle.

```
34. CTA = CTA - (ERR)/(DTDCT)
```

Loop back for iteration on transfer angle.

```
35. Go to 14
```

Compute transfer velocity magnitude from *vis-viva* equation.

36. TRVEL = $((2)/(RL) - (1)/(A))^{.5}$

Compute circular-orbit velocity magnitude.

37. COVEL = $((1)/(RL))^{.5}$

38. Return

39. CALL SETERR (6HLUNAR1)

40. Return

41. CALL SETERR (6HLUNAR2)

42. Return

SUBROUTINE LUNAR NOMENCLATURE

Symbol	Description	Units	Reference
ALP	Auxiliary variable $(\text{Sin}(\text{ALP}/2) = (\text{S}/(2\text{A}))^{.5})$	—	21, 23, 33
A	Semimajor axis of transfer ellipse	c.u.	17, 21, 22, 23, 29, 33, 36
BET	Auxiliary variable $(\text{Sin}(\text{BET}/2) = ((\text{S} - \text{C})/(2\text{A}))^{.5})$	—	22, 23, 33
CC	Square of chord (C)	c.u.	18, 19, 29
CON	1/ke	$(\text{e.r.})^{-1.5}(\text{min})$	EQUIV, 11, 23, 31, 33
COVEL	Coasting orbit velocity	$\text{e.r.}/\text{K}_e^{-1}\text{min}$	/OBART/, 37
CRA	Cosine of lunar right ascension	—	8, 10
C	Chord between injection and target positions	c.u.	19, 20, 22, 28, 33
CTA	Transfer angle	radians	EQUIV, 13, 14, 15, 17, 18, 23, 27, 28, 29, 33, 34
DADCT	Derivative of semimajor axis with respect to transfer angle	c.u.	29, 33
D	Component of lunar position in xy plane	c.u.	7, 8, 9
DELT	Transfer time	hours	CALL, 3, 4
DSDCT	Derivative of semiperimeter (S) with respect to transfer angle	c.u.	28, 33

SUBROUTINE LUNAR NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DTDCT	Derivative of time with respect to transfer angle	c.u.	31, 33, 34
ERR	Error in time of flight	c.u.	23, 24, 34
K	Iteration counter	—	1, 25, 26
LDATE	Launch date	Julian days	CALL, DP, 4
N	Exponent of tolerance check (= 10^{-N})	—	2, 24
PIBY2	$\pi/2$	—	EQUIV, 21, 22
PI	π = 3.14159265	—	DATA, 15, 23, 27
PLAT	Declination of moon	—	/OBART/, 6
PSIDT	Right ascension of moon	—	/OBART/, 10
RL	Distance of injection point from center of earth	c.u.	CALL, 11, 14, 17, 18, 20, 28, 29, 31, 36, 37
RT	Distance of moon at impact from center of earth	c.u.	5, 11, 14, 17, 18, 20, 28, 29, 31
SEL	Unit vector to moon	c.u.	DIM, 5, 6, 7, 8, 9
SRA	Sine of lunar right ascension	—	9, 10
S	Semiperimeter of triangle formed by RL, RT, C	c.u.	20, 21, 22, 33
TFL	Time of flight from injection to impact	min	2, 12, 24, 33

SUBROUTINE LUNAR NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TJD	Julian day at lunar impact	days	DP, 4, 5
TLIM	Limiting time for elliptical orbit (parabolic transfer time)	c.u.	11, 12
TRVEL	Required transfer velocity for lunar orbit	e.r./K_e^{-1}min	/OBART/, 36

```
$IBFTC LUNAR   FULIST,REF
      SUBRØUTINE LUNAR(LDATE,DELT,RL)
      CØMMØN/ASTRØ/KSUBS,KSUBE,ECLIP,KINVE
      REAL  KSUBS,KSUBE,KINVE
      CØMMØN /ØBART/PLAT,PSIDT,TRVEL,TRANG,CØVEL
      EQUIVALENCE(CTA,TRANG),(CØN,KINVE)
      DIMENSIØN SEL(3)
      DØUBLE PRECISIØN  LDATE
      DØUBLE PRECISIØN TJD
      DATA  PI,PIBY2/3.14159265,1.57079633/
      K=0
      N=6
C     GET PØSITIØN AND DISTANCE ØF MØØN AT IMPACT
      TFL = DELT * 60.0
      TJD = LDATE + DELT/24.0
      CALL MØØN(TJD,SEL,RT)
C     GET LATITUDE AND RIGHT ASCENSIØN ØF MØØN AT IMPACT
      PLAT = ASIN(SEL(3))
      D=SQRT(SEL(1)**2+SEL(2)**2)
      CRA=SEL(1)/D
      SRA=SEL(2)/D
      CALL QUAD(CRA,SRA,PSIDT)
C     Q IS ELLIPTIC TRANSFER PØSSIBLE IN TIME
      TLIM= 0.33333333*CØN* SQRT(2.*(RT-RL))*(RT+2.*RL)
      IF(TFL.LE.TLIM) GØ TØ 30
C     CØMPUTE TRANSFER ANGLE
C     START ITERATIØN WITH ESTIMATE ØF TRANSFER ANGLE
      CTA=3.
    2 IF(1.+CØS(CTA).LT.2.*RL/RT) GØ TØ 1
      CTA=.5*(CTA+PI)
      GØ TØ 2
C     FIND SEMI-MAJØR AXIS ØF TRANSFER ELLIPSE
    1 A=RL*(RL-RT*CØS(CTA))/(2.*RL-RT*(1.+CØS(CTA)))
      CC=RL*RL+RT*RT-2.*RL*RT*CØS(CTA)
      C=SQRT(CC)
      S=.5*(RL+RT+C)
      ALP=2.*(PIBY2-ACØS(SQRT(.5*S/A)))
      BET=2.*(PIBY2-ACØS(SQRT(.5*(S-C)/A)))
C     TIME-ØF-FLIGHT ERRØR
      ERR=A**(1.5)*CØN*(PI+SIGN(1.,SIN(CTA))*((ALP-SIN(ALP)-PI)-(BET
     1 -SIN (BET)))) - TFL
      IF(ABS(ERR/TFL).LT.10.**(-N)) GØ TØ 3
      K=K+1
      IF(K.GE.10)GØ TØ 31
      IF(CTA.EQ.PI) GØ TØ 10
C     CØMPUTE DERIVATIVES FØR NEWTØN-RAPHSØN
      DSDCT=.5*RL*RT*SIN(CTA)/C
      DADCT=-2.*RT*(A-RL)*SIN(CTA)/(RL*RL+CC-RT*RT)
      GØ TØ 11
   10 DTDCT= SQRT(.5*RT**3*(RL+RT)/RL)*CØN
```

FIGURE 69. SYMBOLIC LISTING OF SUBROUTINE LUNAR

```
      GØ TØ 12
   11 DTDCT= 1.5*TFL*DADCT+ SQRT(A)*CØN*SIGN(1.,SIN(CTA))*(TAN(.5*ALP)
     1 *(DSDCT-S*DADCT)+ TAN(.5*BET)*(DSDCT+(S-C)*DADCT))
C     REVISE TRANSFER ANGLE
   12 CTA= CTA - ERR/DTDCT
      GØ TØ 2
C     VELØCITIES FØR TRANSFER AND CIRCULAR ØRBIT
    3 TRVEL=SQRT(2./RL-1./A)
      CØVEL=SQRT(1./RL)
      RETURN
C        TRANSFER IMPØSSIBLE
   30 CALL SETERR(6HLUNAR1)
      RETURN
C        MAX ITERATIØNS
   31 CALL SETERR(6HLUNAR2)
      RETURN
      END
```

FIGURE 69. SYMBOLIC LISTING OF SUBROUTINE LUNAR (cont.)

LVI. SUBROUTINE BART

DISCUSSION

The chief functions of Subroutine BART are to compute the time of lift-off of the booster, the time of reignition, and the reignition guidance parameters. These functions may be broken into the following ten subfunctions:

a. Compute the sidereal time of the vehicle at end-of-boost.

b. Compute the sidereal time of the launch point on the Earth at end-of-boost.

c. Compute the sidereal time of the launch point on the Earth at the time of launch.

e. Compute the direction cosines of the pierce point.

f. Compute the direction cosines of the position vector at end-of-boost.

g. Compute the position and velocity vectors at end-of-boost in Earth centered inertial (ECI) coordinates.

h. Compute the coast angle and coast time from end-of-boost to reignition.

i. Predict position and velocity vectors at reignition.

j. Compute thrust attitude (direction cosines TX, TY, TZ) during reignition.

k. Compute required velocity vector at the end of thrusting of the reignition rocket.

BART is utilized in the simulation of interplanetary and lunar missions. Its inputs are the position and velocity at the end-of-boost in local Earth coordinates and the outputs of Subroutine PATCH (for interplanetary missions) or Subroutine LUNAR (for lunar missions).

LIMITATIONS OF SUBROUTINE BART

The assumptions which limit the flexibility of Subroutine BART are:

a. Firing azimuth of the booster is 90° (due east).

b. The coasting orbit is circular.

c. The latitude of the pierce point must be equal to or less than the latitude of the vehicle at the end-of-boost.

d. All free-flights (free-falls) are considered to be two-body or Keplerian motion.

Assumption (c) allows assumption (a) as will be seen later in this analysis. A due east launch takes maximum advantage of the Earth's rotation and it will be shown that such a firing azimuth is almost always possible for the missions being considered. Assumption (b) is realistic since the least energy coasting orbit is a circular orbit just above the atmosphere. Assumption (d) is considered quite satisfactory for the accuracies required for this program.

The reignition powered flight simulation may hold the thrust at a constant attitude, or it may steer so as to achieve the required velocity. All of the computations (Sub-functions (a) through (k)) are easily understood except perhaps (a), (g) and (h). They will be derived in detail in the next section.

END-OF-BOOST TIME AND COAST ANGLE

The launch position and the altitude of the desired circular orbit are known, therefore, since the firing azimuth is always 90°, the boost trajectory may be simulated before the launch time is determined. The position and velocity at the end-of-boost in local Earth coordinates (latitude, longitude, altitude, azimuth, etc.) are inputs to Subroutine BART. Since the boost flight must remain in or very near a plane (to minimize aerodynamic moments and heating), the azimuth of the velocity vector at the end-of-boost will also be approximately 90°.

A launch azimuth of due east is always possible and will still cause the vehicle to pass over the pierce point as long as the declination (latitude) of the vehicle at the end-of-boost is greater than the declination of the pierce point (another effect of the due east launch is that the latitude at end-of-boost is essentially the launch latitude). This assumption concerning the declinations is valid for lunar and planetary missions. Consider the ecliptic inclinations of the following orbits:

Orbit	Ecliptic Inclination (i)
Venus	3.394 degrees
Mars	1.849 "
Mercury	7.00 "
Saturn	2.487 "
Moon (Earth's)	5.15 "

The latitude of the Atlantic Missile Range is 28.45° and the equator plane is inclined 23.444° to the ecliptic. Therefore, on only rare occasions, for lunar and Mercury mission, may the latitude of the pierce point be greater than the latitude at launch. (The maximum possible latitude of the pierce point is the sum of the target body orbital inclination and the inclination of the equator plane.) In the rare instances where the pierce point latitude is greater than the launch latitude, a near east launch would be desired; a small out-of-plane maneuver would be utilized in the insertion into the escape orbit. This situation is not considered here.

In order to launch east and pass over the pierce point, the time of launch can be one of two values, depending on whether it is desired to pass over the pierce point during the first half (0 - 180°) of the coasting orbit or the second half (180 - 360°). Only the first half is considered here. Figure 70 illustrates the first condition when the point of injection into the escape orbit ($\underline{r}_o$) is between the end-of-boost position ($\underline{r}_b$) and the pierce point position ($\underline{r}_p$). The North Pole, $\underline{r}_b$, and $\underline{r}_p$ constitute a right-spherical triangle since the launch azimuth is 90°. The angle at the pole is the difference in right ascensions (sidereal times) of $\underline{r}_p$ and $\underline{r}_b$ and can be computed from a spherical geometry formula from two known sides:

$$\cos(\alpha_p - \alpha_b) = \tan(90 - \delta_b)\ \cot(90 - \delta_p) \qquad (1)$$

or

$$\cos(\alpha_p - \alpha_b) = \cot(\delta_b)\ \tan(\delta_p) \qquad (2)$$

and

$$0 < \alpha_p - \alpha_b < 90^\circ \quad \text{for} \quad \delta_p > 0\ , \qquad (3)$$

$$\alpha_p - \alpha_b = 90^\circ \quad \text{for} \quad \delta_p = 0 \qquad (4)$$

$$90 < \alpha_p - \alpha_b < 180 \quad \text{for} \quad \delta_p < 0 \tag{5}$$

$$\alpha_p - \alpha_b = 180 \quad \text{for} \quad \delta_p = -\delta_b \tag{6}$$

These equations assume that the magnitude of δ_b is equal to or greater than the magnitude of δ_p, and that launch is in the northern hemisphere.

The mid-point of reignition thrusting will occur after passing through the angle ψ (Figure 70) from the end-of-boost position. To find ψ first solve for the total angle $(\psi + \bar{v})$:

$$\cos(\psi + \bar{v}) = \frac{\underline{r}_b \cdot \underline{r}_p}{r_b \, r_p} \tag{7}$$

where:

$$0 \leq (\psi + \bar{v}) \leq 180^{\circ} \quad \text{for} \quad \dot{\underline{r}}_b \cdot \underline{r}_p \geq 0$$

$$180 < (\psi + \bar{v}) < 360^{\circ} \quad \text{for} \quad \dot{\underline{r}}_b \cdot \underline{r}_p < 0 \; .$$

The angle from end-of-boost to the mid-point of reignition is then:

$$\psi = (\psi + \bar{v}) - \bar{v} \, . \tag{8}$$

If ψ is negative, its complement with 360° must be taken as ψ since the rocket cannot reverse directions instantaneously.

$\underline{r}_b$ = end-of-boost position vector

$\underline{r}_o$ = position vector at mid-point of reignition

$\underline{r}_p$ = pierce point position vector

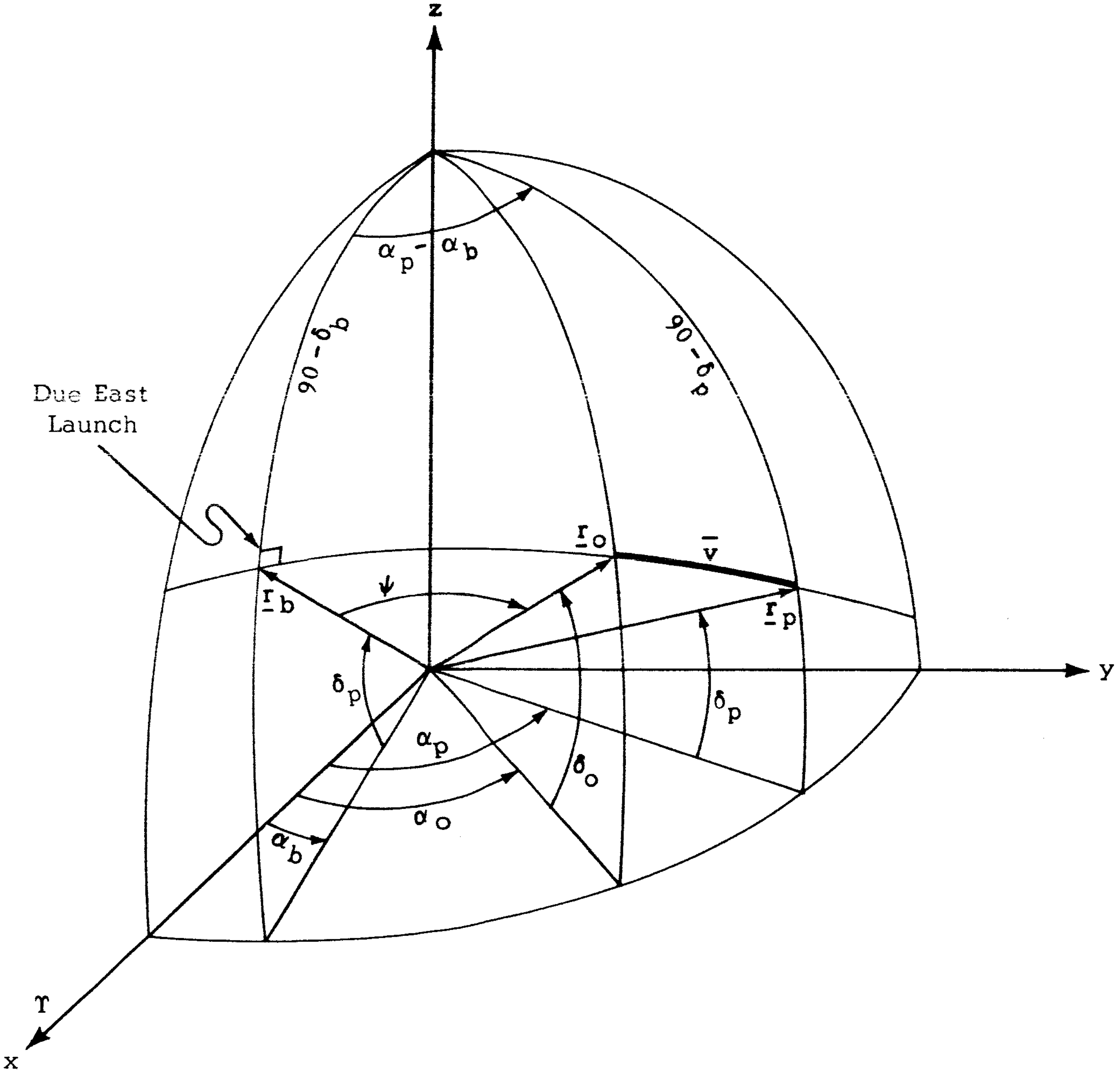


FIGURE 70. GEOMETRY OF THE LAUNCH CONDITIONS

PREDICT POSITION AND VELOCITY VECTORS AT REIGNITION

The position and velocity vectors at reignition are predicted by means of the closed form expressions of the f and g series. The general form of the position and velocity at reigntion based on the position and velocity at end-of-boost is:

$$\underline{r}_r = f\ \underline{r}_b + g\ \dot{\underline{r}}_b \tag{9}$$

$$\dot{\underline{r}}_r = \dot{f}\ \underline{r}_b + g\ \dot{\underline{r}}_b \tag{10}$$

where

$$\left.\begin{array}{l}\underline{r}_r\\ \\ \dot{\underline{r}}_r\end{array}\right\}\ \text{position and velocity at reignition}$$

$$\left.\begin{array}{l}\underline{r}_b\\ \\ \dot{\underline{r}}_b\end{array}\right\}\ \text{position and velocity at end-of-boost}$$

and

$$f = 1 - a\left[1 - \cos(\Delta E)\right]/r_b \tag{11}$$

$$\begin{aligned} g = \frac{1}{\sqrt{\mu}}\Big\{ & r_b\sqrt{a}\ \sin(\Delta E) + (\underline{r}_b \cdot \dot{\underline{r}}_b) \\ & a\left[1 - \cos(\Delta E)\right]\Big\} \end{aligned} \tag{12}$$

$$\dot{f} = -\frac{\sqrt{\mu}}{r_r r_b} \sqrt{a} \sin(\Delta E) \qquad (13)$$

$$\dot{g} = 1 - a\left[1 - \cos(\Delta E)\right]/r_r \qquad (14)$$

$$r_r = r_b + \left(1 - \frac{r_b}{a}\right) a\left[1 - \cos(\Delta E)\right] + \frac{\underline{r}_b \cdot \dot{\underline{r}}_b}{\sqrt{\mu}} \sqrt{a} \sin(\Delta E) \qquad (15)$$

The angle ΔE is the difference in eccentric anomalies of $\underline{r}_r$ and $\underline{r}_b$ and is known from the coast time, T_{cst}:

$$\Delta E = \Delta M = T_{cst} \frac{k_e}{a^{3/2}} \qquad (16)$$

where:

k_e = gravitational constant,

a = semi-major axis of circular coasting orbit.

It is easily seen from inspection that the expressions (11), (12), (13) and (14) may be reduced for the special case of a circular orbit to:

$$f = \cos(\Delta E) \qquad (17)$$

$$g = \frac{a^{3/2}}{\sqrt{\mu}} \sin(\Delta E) \qquad (18)$$

$$\dot{f} = -\frac{\sqrt{\mu}}{a^{3/2}} \sin(\Delta E) \qquad (19)$$

$$\dot{g} = f \qquad (20)$$

Expressions (17) through (20) are the ones actually implemented in Subroutine BART.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL BART

SOLUTION METHOD

Define earth equatorial radius.

1. RE = RSUBO

Convert pierce-point declination to degrees.

2. DLATP = (PLAT) (RTD)

Convert pierce-point right ascension to degrees.

3. THEP = (PSIDT) (RTD)

Convert transfer angle to degrees.

4. VBAR = (TRANG) (RTD)

Test to see if east launch is possible.

5. If $|DLATB| \geq |DLATP|$, go to 7

Otherwise,

6. CALL SETERR (4HBART)

Return.

Compute sidereal time of booster burnout from spherical geometry (assume a firing azimuth of 90 degrees):

7. CTPL = ((Cos ((DTR) (DLATB)))/(Sin ((DTR) (DLATB))))
   (Tan ((DTR) (DLATP)))

8. CALL QUAD (CTPL, $(1 - CTPL^2)^{.5}$, TPL)

9. THEB = (THEP) (DTR) - TPL

Compute sidereal time at launch (booster lift-off):

10. THELB = (THEB) (RTD) - (DLONB - DLONL)

11. THEL = THELB - (OMEGA) (TIMB) (RTD)

Compute coast angle from end of boost to mid-point of re-ignition. Begin by computing direction cosines of pierce-point vector:

12. UPX = (Cos ((DLATP) (DTR))) (Cos ((THEP) (DTR)))

13. UPY = (Cos ((DLATP) (DTR))) (Sin ((THEP) (DTR)))

14. UPZ = Sin ((DLATP) (DTR))

Compute direction cosines of end-of-boost position vector:

15. UBX = (Cos ((DLATB) (DTR))) (Cos (THEB))

16. UBY = (Cos ((DLATB) (DTR))) (Sin (THEB))

17. UBZ = Sin ((DLATB) (DTR))

Compute position and velocity vectors at end of boost:

18. SIGMA = (RVCTR)/(RSUBO)

19. RXB = (SIGMA) (UBX)

20. RYB = (SIGMA) (UBY)

21. RZB = (SIGMA) (UBZ)

22. CALL ROT (0, 0, (VMB)(60)/((RE)(DGC)), AZB, 1, 1,
SX1, SY1, SZ1)

23. CALL ROT (SX1, SY1, SZ1, DLATB, 1, 2, SX2, SY2, SZ2)

24. CALL ROT (SX2, SY2, SZ2, -THEB, 2, 3, RXBD,
RYBD, RZBD)

(ROT computes a coordinate transformation for rotation about an axis.)

Compute the range angle from end of boost to the pierce-point:

25. CSVB = (UPX) (UBX) + (UPY) (UBY) + (UPZ) (UBZ)

26. SSVB = $(1 - \text{CSVB}^2)^{.5}$

Compute auxiliary quantity SIGMA1.

27. SIGMA1 = $\text{SIGMA}^{1.5}$

Determine sign of SSVB:

28. PON = (RXBD) (UPX) + (RYBD) (UPY) + (RZBD) (UPZ)

29. If PON < 0, go to 30

If PON ≥ 0, go to 31

30. SSVB = - SSVB

31. CALL QUAD (CSVB, SSVB, SVB)

(QUAD determines the proper quadrant for
an angle given its cosine and sine.)

Calculate range angle:

32. SI = (SVB) (RTD) - VBAR

33. If SI < 0, go to 34

If SI ≥ 0, go to 35

34. SI = 360 + SI

Compute period of coasting orbit:

35. DP = (2) (PI) (SIGMA1) / (DGC)

Estimate thrusting time of re-ignition rocket:

36. TIMR = (RMAS) (| VMF - VMB |) / ((RAT) (60))

Compute coasting time:

37. TCST = (DP) (SI) (DTR) / ((2) (PI)) - (.5) (TIMR)

38. If TCST < 0, go to 39

If TCST ≥ 0, go to 40

39. TCST = DP + TCST

Predict position and velocity at re-ignition in canonical units:

Compute coast angle.

40. DEA = (TCST) (DGC) / (SIGMA1)

Compute f and g coefficients.

41. F = Cos (DEA)

42. G = (SIGMA1)(Sin(DEA))

43. FD = -(Sin(DEA))/(SIGMA1)

Compute position components.

44. RXR = (F)(RXB) + (G)(RXBD)

45. RYR = (F)(RYB) + (G)(RYBD)

46. RZR = (F)(RZB) + (G)(RZBD)

Compute velocity components.

47. RDXR = (FD)(RXB) + (F)(RXBD)

48. RDYR = (FD)(RYB) + (F)(RYBD)

49. RDZR = (FD)(RZB) + (F)(RZBD)

Perform transformation to oblate spheroidal coordinates.

50. CALL G2OB (RXR, RYR, RZR, RDXR, RDYR, RDZR, 1,
FLAT, DR, THER, H, VR, GAMR, AZR, 0)

Compute time at mid-point of re-ignition.

51. TTE = TCST + (.5)(TIMR)

Estimate range angle from start of re-ignition to mid-point.

52. DTEA = (DEA)(TTE)/(TCST)

Compute transformation coefficients.

53. FE = Cos (DTEA)

54. FED = (FD)(Sin(DTEA))/(Sin(DEA))

Compute thrust attitude.

55. $TX = (FED)(RXB) + (FE)(RXBD)$

56. $TY = (FED)(RYB) + (FE)(RYBD)$

57. $TZ = (FED)(RZB) + (FE)(RZBD)$

Normalize thrust.

58. $DNM = (TX^2 + TY^2 + TZ^2)^{.5}$

59. $TX = (TX)/(DNM)$

60. $TY = (TY)/(DNM)$

61. $TZ = (TZ)/(DNM)$

Allocate required transfer velocity in specified direction.

62. $SOX = (VMF)(TX)$

63. $SOY = (VMF)(TY)$

64. $SOZ = (VMF)(TZ)$

Return.

SUBROUTINE BART NOMENCLATURE

Symbol	Description	Units	Reference
AZB	Inertial velocity azimuth at end of boost	deg	/TOL/, 22
AZR	Azimuth of re-ignition thrust vector midway into thrusting period	deg	50
CSVB	Cos (SI + VBAR)	—	25, 26, 31
CTPL	Cos (THEP - THEB)	—	7, 8
DEA	Coast angle from end of boost to re-ignition	rad	40, 41, 42, 43, 52, 54
DGC	Gravitational constant	$e.r.^{1.5}/min$	EQUIV, 22, 35, 40
DLATB	Latitude at end of boost	deg	/TOL/, 5, 7, 15, 16, 17, 23
DLATP	Latitude at pierce-point	deg	2, 5, 7, 12, 13, 14
DLONB	Longitude east at end of boost	deg	/TOL/, 10
DLONL	Longitude east at launch point	deg	/TOL/, 10
DNM	Temporary storage	—	58, 59, 60, 61
DP	Period of the circular coasting orbit	min	35, 37, 39
DR	Declination (latitude) mid-way into re-ignition	deg	50
DTEA	Range angle from start of re-ignition to mid-point	rad	52, 53, 54

SUBROUTINE BART NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DTR	Converts degrees to radians	rad/deg	DATA, 7, 9, 12, 13, 14, 15, 16, 17, 37
F	f coefficient of the f and g series for the coasting orbit	—	41, 44, 45, 46, 47 48, 49
FD	$\dot{f}$ coefficient for coasting orbit	—	43, 47, 48, 49, 54
FE	f transformation coefficient to mid-point of re-ignition	—	53, 55, 56, 57
FED	$\dot{f}$ transformation coefficient to mid-point of re-ignition	—	54, 55, 56, 57
FLAT	Flattening of Earth	—	DATA, 50
G	g coefficient of f and g series for the coasting orbit	—	42, 44, 45, 46
GAMR	Flight path angle with respect to geoid	deg	50
H	Height above reference spheroid	e.r.	50
OMEGA	Spin rate of the Earth	rad/min	CALL, 11
PI	3.14159269	—	DATA, 35, 37
PLAT	Pierce-point declination	rad	/OBART/, 2
PON	Plus or minus test value	—	28, 29
PSIDT	Sidereal time of pierce-point	rad	/OBART/, 3
RAT	Average value of re-ignition thrust	lb	/TOL/, 36

SUBROUTINE BART NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RDXR	Velocity component in the X direction at the beginning of re-ignition	e.r./K_e^{-1} min	/PBART/, 47, 50
RDYR	Velocity component in the Y direction at the beginning of re-ignition	e.r./K_e^{-1} min	/PBART/, 48, 50
RDZR	Velocity component in the Z direction at the beginning of re-ignition	e.r./K_e^{-1} min	/PBART/, 49, 50
RE	Earth equatorial radius	ft	1, 22
RMAS	Total average mass of re-ignition vehicle during re-ignition	slugs	/TOL/, 36
RSUBO	Earth equatorial radius	ft	CALL, 1, 18
RTD	Converts radians to degrees	deg/rad	DATA, 2, 3, 4, 10, 11, 32
RVCTR	Magnitude of position vector	ft	CALL, 18
RXB	ECI position component in the X direction at end of boost	e.r.	/PBART/, 19, 44, 47, 55
RXBD	ECI velocity component in the X direction at end of boost	e.r./K_e^{-1} min	/PBART/, 24, 28, 44, 47, 55
RXR	ECI position component of vehicle in the X direction at re-ignition	e.r.	/PBART/, 44, 50

SUBROUTINE BART NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RYB	ECI position component in the Y direction at end of boost	e.r.	/PBART/, 20, 45, 48, 56
RYBD	ECI velocity component in the Y direction at end of boost	e.r./K_e^{-1} min	/PBART/, 24, 28, 45, 48, 56
RYR	ECI position component of vehicle in the Y direction at re-ignition	e.r.	/PBART/, 45, 50
RZB	ECI position component in the Z direction at end of boost	e.r.	/PBART/, 21, 46, 49, 57
RZBD	ECI velocity component in the Z direction at end of boost	e.r./K_e^{-1} min	/PBART/, 24, 28, 46, 49, 57
RZR	ECI position component of vehicle in the Z direction at re-ignition	e.r.	/PBART/, 46, 50
SI	Range angle from end of boost to mid-point of re-ignition	deg	32, 33, 34, 37
SOX	Required velocity component in the X direction at the end of re-ignition in ECI	ft/sec	/PBART/, 62
SOY	Required velocity component in the Y direction at the end of re-ignition in ECI	ft/sec	/PBART/, 63
SOZ	Required velocity component in the Z direction at the end of re-ignition in ECI	ft/sec	/PBART/, 64

SUBROUTINE BART NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
SSVB	Sin(SI + VBAR)	—	26, 30, 31
SVB	Sum of SI and VBAR	rad	31, 32
SX1	First intermediate component in the series of rotations which produce RXBD, RYBD, RZBD velocity vectors	e.r./K_e^{-1} min	22, 23
SX2	Second intermediate component in the series of rotations which produce RXBD, RYBD, RZBD velocity vectors	e.r./K_e^{-1} min	23, 24
SY1	First intermediate component in the Y direction	e.r./K_e^{-1} min	22, 23
SY2	Second intermediate component in the Y direction	e.r./K_e^{-1} min	23, 24
SZ1	First intermediate component in the Z direction	e.r./K_e^{-1} min	22, 23
SZ2	Second intermediate component in the Z direction	e.r./K_e^{-1} min	23, 24
SIGMA	Magnitude of position vector	e.r.	18, 19, 20, 21, 27
SIGMA1	$\text{SIGMA}^{1.5}$	$\text{e.r.}^{1.5}$	27, 35, 40, 42, 43
TCST	Coast time from end of boost to re-ignition	min	/PBART/, 37, 38, 39, 40, 51, 52
THEB	Sidereal time at end of boost	rad	9, 10, 15, 16, 24

SUBROUTINE BART NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
THEL	Sidereal time at launch	deg	/PBART/, 11
THELB	Sidereal time of launch point at end of boost	deg	10, 11
THEP	Sidereal time at the pierce-point	deg	3, 9, 12, 13
THER	Sidereal time at mid-point of re-ignition	deg	50
TIMB	Time of booster thrusting	min	/TOL/, 11
TIMR	Estimated time of thrusting during re-ignition	min	36, 37, 51
TPL	THEP - THEB	rad	8, 9
TRANG	Central transfer angle (injection to pierce-point)	rad	/OBART/, 4
TTE	Time at mid-point of re-ignition	min	51, 52
TX	Direction cosines of thrust component during re-ignition in ECI	—	/PBART/, 55, 58, 59, 62
TY	Direction cosines of thrust component during re-ignition in the Y direction	—	/PBART/, 56, 58, 60, 63
TZ	Direction cosines of thrust component during re-ignition in the Z direction	—	/PBART/, 57, 58, 61, 64
UBX	Direction cosines of the position component at end of boost in ECI	—	15, 19, 25

SUBROUTINE BART NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
UBY	Direction cosines of the position component at end of boost in the Y direction	—	16, 20, 25
UBZ	Direction cosines of the position component at end of boost in the Z direction	—	17, 21, 25
UPX	Direction of the pierce-point vector in ECI	—	12, 25, 28
UPY	Direction of the pierce-point vector in the Y direction	—	13, 25, 28
UPZ	Direction of the pierce-point vector in the Z direction	—	14, 25, 28
VBAR	Angle from mid-point of re-ignition to pierce-point	deg	3, 32
VMB	Magnitude of the inertial velocity at end of boost	ft/sec	/OBART/, 22, 36
VMF	Magnitude of the required inertial velocity at end of re-ignition	ft/sec	/OBART/, 36, 62 63, 64
VR	Velocity magnitude with respect to geoid	e.r./K_e^{-1} min	/PBART/, 50

```
$IBFTC BART    DECK
      SUBRØUTINE BART(RVCTR,RSUBØ,ØMEGA)
C  BART CØMPUTES BØØST AND REIGNITIØN TIMES
      CØMMØN/ASTRØ/KSUBS,KSUBE,ECLIP,KINVE
      REAL  KSUBS,KSUBE,KINVE
      CØMMØN /TØL/ AZB,DLATB,DLØNB,DLØNL,RAT,RMAS,TIMB
      CØMMØN /ØBART/ PLAT,PSIDT,VMF,TRANG,VMB
      CØMMØN/PBART/RXB,RYB,RZB,RXBD,RYBD,RZBD,
     $             RXR,RYR,RZR,RDXR,RDYR,RDZR,
     $             TX,TY,TZ,PITR,YAWR,SØX,SØY,SØZ,VR,
     $             THEL,TCST
      EQUIVALENCE (DGC,KSUBE)
      DATA  RTD,DTR/57.2957795,0.0174532925/
      DATA  PI,FLAT/3.14159265,0.3352105E-2/
      RE = RSUBØ
      DLATP=PLAT*RTD
      THETP=PSIDT*RTD
      VBAR=TRANG*RTD
C  TEST TØ SEE IF EASTERN LAUNCH IS PØSSIBLE.
      IF (ABS(DLATB) .GE. ABS(DLATP))  GØ TØ 210
      CALL SETERR(4HBART)
      RETURN
C  FØR DUE EAST FIRING AZIMUTH CØMPUTE BØØST BØ TIME FRØM SPHERICAL TRIG
  210 CTPL= CØS(DTR*DLATB)/SIN(DTR*DLATB)*TAN(DTR*DLATP)
      CALL QUAD(CTPL,SQRT(1.0-CTPL**2),TPL)
      THEB= THEP*DTR-TPL
C  CØMPUTE SIDEREAL TIME AT LAUNCH (BØØSTER LIFT-ØFF).
      THELB= THEB*RTD-(DLØNB-DLØNL)
      THEL = THELB-ØMEGA*TIMB*RTD
C  CØMPUTE CØAST ANGLE SI (FRØM END-ØF-BØØST TØ MID-PØINT ØF REIGNITIØN)
C  CØMPUTE DIRECTIØN CØSINES ØF PIERCE PØINT VECTØR
  300 UPX= CØS(DLATP*DTR)*CØS(THEP*DTR)
      UPY= CØS(DLATP*DTR)*SIN(THEP*DTR)
      UPZ= SIN(DLATP*DTR)
C  CØMPUTE DIRECTIØN CØSINES ØF END-ØF-BØØST PØSITIØN VECTØR
      UBX= CØS(DLATB*DTR)*CØS(THEB)
      UBY= CØS(DLATB*DTR)*SIN(THEB)
      UBZ= SIN(DLATB*DTR)
C  CØMPUTE PØSITIØN AND VELØCITY AT END ØF BØØST.
      SIGMA = RVCTR/RSUBØ
  302 RXB = SIGMA * UBX
      RYB = SIGMA * UBY
      RZB = SIGMA * UBZ
      CALL RØT(0.0,0.0, VMB*60.0/(RE*DGC),AZB,1,1,SX1,SY1,SZ1)
      CALL RØT(SX1,SY1,SZ1, DLATB,1,2,SX2,SY2,SZ2)
      CALL RØT(SX2,SY2,SZ2,-THEB,2,3,RXBD,RYBD,RZBD)
C  CØSINE ØF END-ØF-BØØST TØ PIERCE PØINT RANGE ANGLE
      CSVB= UPX*UBX+UPY*UBY+UPZ*UBZ
      SSVB = SQRT(1.0-CSVB**2)
      SIGMA1 = SIGMA**1.5
```

FIGURE 71. SYMBOLIC LISTING OF SUBROUTINE BART

```
C  DETERMINE SIGN ØF SSVB
C      PØN =  RXBD*UPX+RYBD*UPY+RZBD*UPZ
       IF (PØN) 304,306,306
  304 SSVB = -SSVB
  306 CALL QUAD(CSVB,SSVB,SVB)
       SI = SVB *RTD-VBAR
       IF (SI) 308,310,310
  308 SI= 360.0+ SI
C  CØMPUTE PERIØD ØF CØASTING ØRBIT
  310 DP = (2.0*PI*SIGMA1)/DGC
C  ESTIMATE REIGNITIØN THRUSTING TIME,TIMR.
       TIMR= RMAS*ABS(VMF-VMB)/(RAT*60.0)
C  CØMPUTE CØAST TIME FRØM END ØF BØØST TØ REINITIØN
       TCST = (DP/(2.0*PI))*SI*DTR -0.5*TIMR
       IF(TCST)312,314,314
  312 TCST= DP+TCST
C  PREDICT PØSITIØN AND VELØCITY AT REIGNITIØN.
  314 DEA = TCST*DGC/SIGMA1
       F= CØS(DEA)
       G = SIGMA1 * SIN(DEA)
       FD = -SIN(DEA)/SIGMA1
       RXR= F*RXB +G*RXBD
       RYR= F*RYB +G*RYBD
       RZR= F*RZB +G*RZBD
       RDXR= FD*RXB +F*RXBD
       RDYR= FD*RYB +F*RYBD
       RDZR= FD*RZB +F*RZBD
       CALL G20B(RXR,RYR,RZR,RDXR,RDYR,RDZR,1.,FLAT,DR,THER,H,VR,
      $GAMR,AZR,0          )
C  CØMPUTE THRUST ATTITUDE(DIRECTIØN CØSINES TX,TY,TZ) DURING REIGNITIØN
       TTE=TCST+0.5*TIMR
       DTEA= DEA*TTE/TCST
       FE= CØS(DTEA)
       FED= FD*SIN(DTEA)/SIN(DEA)
       TX=(FED*RXB+FE*RXBD)
       TY=(FED*RYB+FE*RYBD)
       TZ=(FED*RZB+FE*RZBD)
       DNM= SQRT(TX**2+TY**2+TZ**2)
       TX= TX/DNM
       TY= TY/DNM
       TZ= TZ/DNM
C  CØMPUTE REQUIRED VELØCITY AT END ØF REIGNITIØN.
       SØX = VMF*TX
       SØY = VMF*TY
       SØZ = VMF*TZ
       RETURN
       END
```

FIGURE 71. SYMBOLIC LISTING OF SUBROUTINE BART (cont.)

LVII. SUBROUTINE ROT

This subroutine computes the components of a vector in an orthogonal set which has been rotated about one axis from another orthogonal set. The rotation is assumed to be counterclockwise, looking toward the origin along the axis of rotation.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL ROT (XO, YO, ZO, BET, IB, N, X1, Y1, Z1)

SOLUTION METHOD

Set degrees-to-radians conversion factor:

1. DTR = .017453292

Test if rotation angle is in degrees or radians:

2. If IB $\geq$ 2, go to 5

   If IB $<$ 2, go to 3

Convert rotation angle to radians:

```
3.  BETA = (BET)(DTR)

4.  Go to 6

5.  BETA = BET

Determine axis of rotation:

6.  If N = 1, go to 7
    If N = 2, go to 11
    If N = 3, go to 15

Rotation about X-axis:

7.  X1 = XO
8.  Y1 = (YO)(Cos(BETA)) + (ZO)(Sin(BETA))
9.  Z1 = -(YO)(Sin(BETA)) + (ZO)(Cos(BETA))
10. Go to 18

Rotation about Y-axis:

11. X1 = (XO)(Cos(BETA)) - (ZO)(Sin(BETA))
12. Y1 = YO
13. Z1 = (XO)(Sin(BETA)) + (ZO)(Cos(BETA))
14. Go to 18

Rotation about Z-axis:
```

15. $X1 = (XO)(\mathrm{Cos}(BETA)) + (YO)(\mathrm{Sin}(BETA))$

16. $Y1 = -(XO)(\mathrm{Sin}(BETA)) + (YO)(\mathrm{Cos}(BETA))$

17. $Z1 = ZO$

18. Return

SUBROUTINE ROT NOMENCLATURE

Symbol	Description	Units	Reference
BET	Rotation angle	degrees or radians	CALL, 3, 5
BETA	Rotation angle	radians	3, 5, 8, 9, 11, 13, 15, 16
DTR	Degrees-to-radians conversion factor	rad/deg	1, 3
IB	Indicator denoting units of BET IB = 1, BET in degrees IB = 2, BET in radians	—	CALL, 2
N	Indicator denoting axis of rotation N = 1, X-axis N = 2, Y-axis N = 3, Z-axis	—	CALL, 6
XO	Original X-component	—	CALL, 7, 11, 13, 15, 16
X1	Rotated X-component	—	CALL, 7, 11, 15
YO	Original Y-component	—	CALL, 8, 9, 12, 15, 16
Y1	Rotated Y-component	—	CALL, 8, 12, 16
ZO	Original Z-component	—	CALL, 8, 9, 11, 13, 17
Z1	Rotated Z-component	—	CALL, 9, 13, 17

```
$IBFTC RØT     REF,LIST
      SUBRØUTINE RØT(XØ,YØ,ZØ,BET ,IB,N,X1,Y1,Z1)
C     GIVEN THE CØMPØNENTS ØF A VECTØR (XØ,YØ,ZØ) IN AN ØRIGINAL CØØRD
C     INATE SYSTEM.  RØT FINDS THE CØMPØNENTS (X1,Y1,Z1) IN A CØØRDINATE
C     SYSTEM WHICH IS RØTATED THRU BETA (CCW LØØKING TØWARD ØRIGIN ALØNG
C     AXIS ØF RØTATIØN) FRØM THE ØRIGINAL.
C     N=1 FØR RØTATIØN ABØUT XØ AXIS, N=2 FØR YØ AXIS, N=3 FØR ZØ AXIS.
C     IB=1 IF BETA IS IN DEGREES.    IB=2 IF IN RADIANS.
      DTR=.017453292
      IF (IB-2)10,20,20
   10 BETA=BET*DTR
      GØ TØ 29
   20 BETA = BET
   29 GØ TØ (30,40,50),N
   30 X1=XØ
      Y1= YØ*CØS(BETA)+ZØ*SIN(BETA)
      Z1= -YØ*SIN(BETA)+ZØ*CØS(BETA)
      GØ TØ 60
   40 X1= XØ*CØS(BETA)-ZØ*SIN(BETA)
      Y1=YØ
      Z1= XØ*SIN(BETA)+ZØ*CØS(BETA)
      GØ TØ 60
   50 X1=XØ*CØS(BETA)+YØ*SIN(BETA)
      Y1=-XØ*SIN(BETA)+YØ*CØS(BETA)
      Z1=ZØ
   60 RETURN
      END
```

FIGURE 72. SYMBOLIC LISTING OF SUBROUTINE ROT

LVIII. SUBROUTINE INJEC

The purpose of this subroutine is to store the guidance velocity components required to transfer the vehicle into the transfer orbit.

CALLING SEQUENCE

The calling sequence is:

CALL INJEC

SOLUTION METHOD

Compute velocity conversion factor.

1. CON = (RSUBO)/((13.447052)(60.0))

Specify inertial velocity as the section ending parameter.

2. PARAM = LOC(VSUBI)

Specify magnitude of velocity for section ending.

3. VE = (VR)(CON)

Compute required pitch and yaw kicks.

4. PKIK = (PITR - PIT)(DPRAD)

5. YKIK = (YAWR - YAW) (DPRAD)

Set pitch and yaw rates to zero.

6. PDOT = 0.0

7. YDOT = 0.0

Set components of velocity-to-go.

8. VGX = SOX

9. VGY = SOY

10. VGZ = SOZ

Return.

SUBROUTINE INJEC NOMENCLATURE

Symbol	Description	Units	Reference
CON	Conversion factor	$\frac{K_e \text{ min ft}}{\text{e.r. sec}}$	1, 3
DPRAD	Conversion factor	deg/rad	/CONST/, 4, 5
PARAM	Section ending parameter	—	/WORKD/, 2
PDOT	Pitch rate	rad/sec	/WORKD/, 6
PITR	Pitch angle at reignition	rad	/PBART/, 4
PKIK	Pitch kick	deg	/WORKD/, 4
PIT	Pitch angle at beginning of section	rad	/DATA/, 4
RSUBO	Earth equatorial radius	ft	EQUIV, 1
SOX	X-component of inertial velocity-to-go	ft/sec	/PBART/, 8
SOY	Y-component of inertial velocity-to-go	ft/sec	/PBART/, 9
SOZ	Z-component of inertial velocity-to-go	ft/sec	/PBART/, 10
VE	Magnitude of inertial velocity-to-go	ft/sec	/WORKD/, 3
VGX	X-component of inertial velocity-to-go	ft/sec	/WORKD/, 8
VGY	Y-component of inertial velocity-to-go	ft/sec	/WORKD/, 9
VGZ	Z-component of inertial velocity-to-go	ft/sec	/WORKD/, 10

SUBROUTINE INJEC NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
VR	Magnitude of velocity-to-go	e.r./K_e^{-1} min	/PBART/, 3
VSUBI	Name of section ending parameter	ft/sec	/DATA/, 2
YAWR	Yaw angle at re-ignition	rad	/PBART/, 5
YDOT	Yaw rate	rad/sec	/WORKD/, 7
YKIK	Yaw kick	deg	/WORKD/, 5
YAW	Yaw angle at beginning of section	rad	/DATA/, 5

```
$IBFTC INJEC   FULIST,REF
      SUBRØUTINE INJEC
C
C     STØRE GUIDANCE VELØCITIES FØR INJECTIØN
C
      CØMMØN/CØNST/RSUBO,FLAT,GSUBO,GSUB1,ØMEGA,PSL,CMAX,HMIN,HMAX,
     $            TAGS(7),INFIN,PI,PIØ2,DTRAD,DPRAD
      EQUIVALENCE  (RSUBØ,RSUBO),(GSUBØ,GSUBO)
      REAL  INFIN
      CØMMØN/WØRK+/PARAM,VE,WIØWJ,STIND,THIND,ATIND,DTSBS,DTSBP,
     $            PDØT,YDØT,PKIK,YKIK,VGX,VGY,VGZ,SECTNØ,
     $            PADSK,VESK,PADRT1,PADRT2,
     $          SREF,WIØWJS,WØIS,WFIS,WPRPS,WBØS,FVAC,WDØTV,EPSIS,
     $           ASUBTS,ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA(4)
      REAL  ISP,KSUBF,KSUBW,KSUBP,KSUBY,KSUBN,KSUBA
      INTEGER  STIND,THIND,ATIND,SECTNØ
      CØMMØN /SAVE/JD,SIDETØ,PHIØ,SPHIØ,CPHIØ,THETAØ,STHETØ,CTHETØ,
     $            AZIR,RØX,RØY,RØZ,TSTG,TSUBS,PSUBS,YSUBS,
     $            WØR(6),WFR(6),AVMR(6)
      DØUBLE PRECISIØN  JD
      CØMMØN /DATA/ TFLIT,TSTAG,TSECT,TSUBP,DTLST,RXIN,RYIN,RZIN,RVCTR,
     $            PHI,THETA,HATM,VSUBI,GAMIN,AZIN,VSUBR,GAMMA,AZ,BETA,
     $            GAMRF,AZREF,PIT,YAW,PRATE,YRATE,RATM,CATM,TATM,PATM,
     $            QATM,MACH,ALPHAP,ALPHAY,ALPHA,ALPHAD,CSUBN,CSUBA,
     $            CSUBL,CSUBD,LIFT,DRAG,THRST,GRAV,LAT,LØNG,RANGE,
     $            ESUBP,ESUBK,ERATIØ
      REAL  LAT,LØNG,MACH,LIFT
      CØMMØN/PBART/RXB,RYB,RZB,RXBD,RYBD,RZBD,
     $            RXR,RYR,RZR,RDXR,RDYR,RDZR,
     $            TX,TY,TZ,PITR,YAWR,SØX,SØY,SØZ,VR,
     $            THEL,TCST
      INTEGER  PARAM
      PARAM = LØC(VSUBI)
      VE = SQRT(SØX**2 + SØY**2 + SØZ**2)
      PKIK = (PITR - PIT) * DPRAD
      YKIK = (YAWR - YAW) * DPRAD
      PDØT = 0.0
      YDØT = 0.0
      VGX = SØX
      VGY = SØY
      VGZ = SØZ
      RETURN
      END
```

FIGURE 73. SYMBOLIC LISTING OF SUBROUTINE INJEC

LIX. SUBROUTINE QUAD

This subroutine calculates the value of an angle between 0 and 2π radians, given the sine and cosine of the angle.

CALLING SEQUENCE

The calling sequence is:

```
CALL QUAD (Y, X, A)
```

SOLUTION METHOD

```
1.  PI = 3.14159265

2.  AB = | ARCTAN ((X)/(Y)) |

3.  If X < 0, go to 5

    If X = 0, go to 14

    If X > 0, go to 4

4.  If Y < 0, go to 8

    If Y = 0, go to 19

    If Y > 0, go to 6
```

```
5.  If $Y < 0$, go to 10
    If $Y = 0$, go to 19
    If $Y > 0$, go to 12
6.  $A = AB$
7.  Go to 23
8.  $A = PI - AB$
9.  Go to 23
10. $A = PI + AB$
11. Go to 23
12. $A = (2.0)(PI) - AB$
13. Go to 23
14. If $Y < 0$, go to 17
    If $Y \geq 0$, go to 15
15. $A = 0$
16. Go to 23
17. $A = PI$
18. Go to 23
19. If $X < 0$, go to 22
    If $X \geq 0$, go to 20
20. $A = (PI)/(2.0)$
21. Go to 23
```

```
22.  A = (3.0)(PI)/(2.0)
23.  Return
```

SUBROUTINE QUAD NOMENCLATURE

Symbol	Description	Units	Reference
A	Angle solution	Radians	CALL, 6, 8, 10, 12, 15, 17, 20, 22
AB	Arctangent	Radians	2, 6, 8, 10, 12
PI	3.14159265	Radians	1, 8, 10, 12, 17, 20, 22
X	Sine of angle	—	CALL, 2, 3, 19
Y	Cosine of angle	—	CALL, 2, 4, 5, 14

```
      SUBROUTINE QUAD (Y,X,A)
C     X=SINE OF ANGLE
C     Y=COSINE OF ANGLE
C
      PI=.314159265E+01
      AB=ABS( ATAN(X/Y) )
      IF (X) 1,2,3
    3 IF (Y) 5,8,4
    1 IF (Y) 6,8,7
    4 A=AB
      GO TO 25
    5 A=PI-AB
      GO TO 25
    6 A=PI+AB
      GO TO 25
    7 A= 2.0*PI-AB
      GO TO 25
    2 IF (Y) 12,14,14
   14 A=0.0
      GO TO 25
   12 A=PI
      GO TO 25
    8 IF (X) 15,16,16
   16 A= PI/2.0
      GO TO 25
   15 A= 3.0*PI/2.0
   25 RETURN
      END
```

FIGURE 74. SYMBOLIC LISTING OF SUBROUTINE QUAD

LX. FUNCTION TAN

Function TAN calculates the tangent of an angle.

CALLING SEQUENCE

The function TAN is used in the following way:

Z = TAN(X)

SOLUTION METHOD

```
1. TAN = (Sin(X))/(Cos(X))
   Return
```

FUNCTION TAN NOMENCLATURE

Symbol	Description	Units	Reference
TAN	Tangent of angle X	—	1
X	Angle	Radians	1

```
$IBFTC TAN     FULIST,REF
      FUNCTIØN TAN(X)
C        CALCULATE THE TANGENT ØF AN ANGLE IN RADIANS
      TAN = SIN(X)/CØS(X)
      RETURN
      END
```

FIGURE 75. SYMBOLIC LISTING OF SUBROUTINE TAN

LXI. FUNCTION ASIN

Function ASIN calculates the value of an angle between ± 90 degrees, given the sine of the angle.

CALLING SEQUENCE

The function ASIN is used in the following way:

$$Z = ASIN(X)$$

where Z is any legal FORTRAN variable and X is the sine of the angle being sought.

SOLUTION METHOD

```
1.  $Y = (1.0 - X^2)^{.5}$
2.  $T = \left| (X)/(Y) \right|$
3.  $A = \text{Arctangent}(T)$
4.  If $X < 0$, go to 5
    If $X \geq 0$, go to 6
5.  $A = -A$
6.  $ASIN = A$
    Return
```

FUNCTION ASIN NOMENCLATURE

Symbol	Description	Units	Reference
A	Angle solution	Radians	3,5,6
ASIN	Arcsine	Radians	6
T	Tangent of angle	—	2,3
X	Sine of angle	—	1,2,4
Y	Cosine of angle	—	1,2

```
$IBFTC ASIN    REF,LIST
      FUNCTION ASIN (X)
      Y=SQRT(1.0-X**2)
      T=ABS(X/Y)
      A=ATAN(T)
      IF(X) 1,2,2
    1 A=-A
    2 ASIN =A
      RETURN
      END
```

FIGURE 76. SYMBOLIC LISTING OF SUBROUTINE ASIN

LXII. FUNCTION ACOS

Function ACOS calculates the value of an angle between 0 and 180 degrees, given the cosine of the angle.

CALLING SEQUENCE

The function ACOS is used in the following way:

$$Z = ACOS(X)$$

where Z is any legal FORTRAN variable and X is the cosine of the angle being sought.

SOLUTION METHOD

1. $Y = (1.0 - X^2)^{.5}$
2. $T = \left| (Y)/(X) \right|$
3. $A = \text{Arctan}(T)$
4. If $X < 0$, go to 5
   If $X \geq 0$, go to 6
5. $A = 3.14159265 - A$
6. $ACOS = A$

   Return

FUNCTION ACOS NOMENCLATURE

Symbol	Description	Units	Reference
A	Angle solution	Radians	3,5,6
ACOS	Arccosine	Radians	6
T	Tangent of angle	—	2,3
X	Cosine of angle	—	1,2,4
Y	Sine of angle	—	1,2

```
$IBFTC ACOS     REF,LIST
      FUNCTION ACOS (X)
      Y=SQRT(1.0-X**2)
      T=ABS(Y/X)
      A=ATAN(T)
      IF(X) 1,2,2
    1 A=3.14159265-A
    2 ACOS =A
      RETURN
      END
```

FIGURE 77. SYMBOLIC LISTING OF SUBROUTINE ACOS

LXIII. SUBROUTINE CNTROL

CNTROL is the control routine for Block I (non-linear engine balance/rebalance package) of the TENSOR I computer program. It controls input and output for Block I along with logically sequencing the flow of these programs.

Input for Block I is accomplished by the standard FORTRAN IV NAMELIST feature (see section "Input for Engine Programs"). Input and output for Block II (powered flight simulation), and Block III (interplanetary transfer simulation) are handled internal to those routines.

A breakdown of correct operation and running instructions for the TENSOR computer program is outlined in the following sections.

CALLING SEQUENCE

CALL CNTROL(F)

OPERATION OF THE PROGRAM

TENSOR Header Card

This card is checked before each case. (One TENSOR header card is needed for each case.)

Columns 1 - 6 = "TENSOR"

Columns 7 - 12 = a digit from 0 to 5 (or blank)

A digit from 1 to 4 represents the particular engine related to the stage number. The digit codes for the engines are:

1 = H1 type engine

2 = F1 type engine

3 = J2 type engine

4 = RL10-A1 type engine

Zeros or blanks will end the particular vehicle configuration if less than six stages are to be run.

The digit 5 signifies that the particular stage's influence coefficients will be input to the trajectory portion of the program.

Since there may be a maximum of 6 stages for the vehicle, each column represents a stage (Col. 7 = Stage 1, Col. 8 = Stage 2, ..., ..., Col. 12 = Stage 6).

Col. 15 = 0 or 1: A 0 (zero) signals that the engine programs will be run, but the trajectory programs will not be run. A 1 (one) signals that the trajectory program will be run along with any specified engine programs.

See Figure 78 for a sample TENSOR HEADER card.

Input for Engine Programs

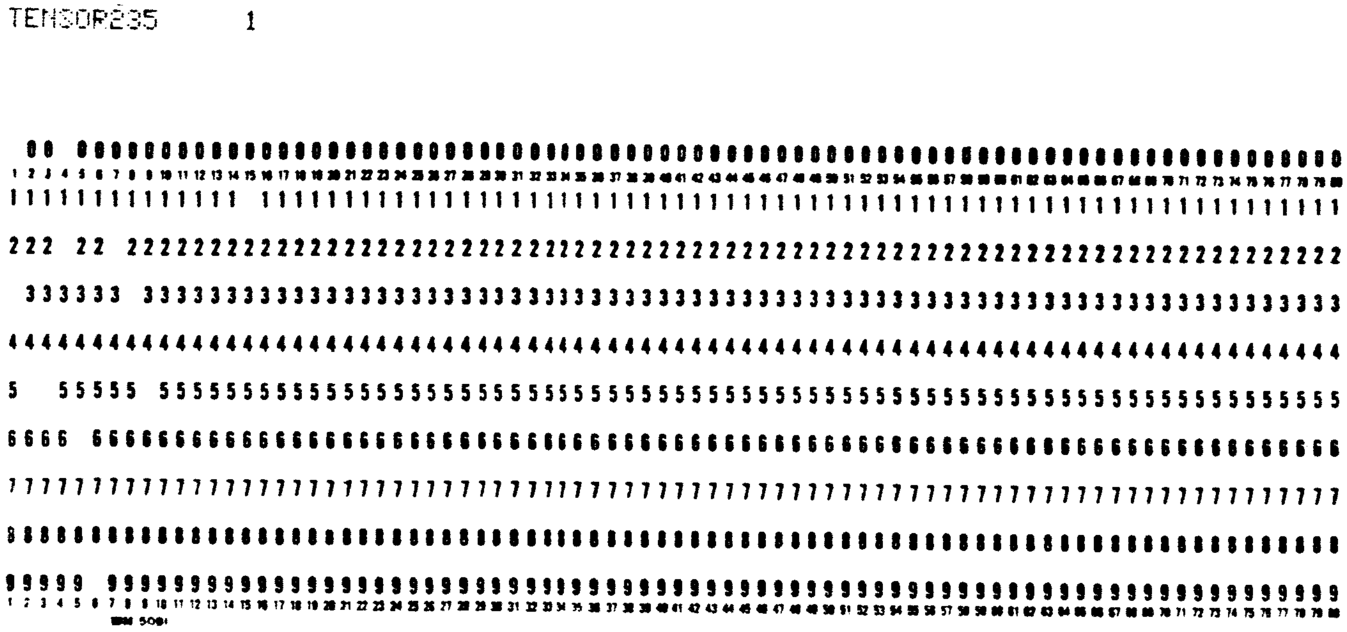
As stated previously, input to the engine programs is accomplished with the standard FORTRAN IV NAMELIST feature.

The NAMELIST names for the various engines are:

ENGH1 = H1 type engine

ENGF1 = F1 type engine

ENGJ2 = J2 type engine



TENSOR235 1

This HEADER card signifies:

1. The flight of a three-stage vehicle will be simulated.
2. Block I will be used to generate F-1 and J-2 nominal engine performance and influence coefficients for use on Stages 1 and 2 respectively.
3. Influence coefficients will be input to Block II for the third stage of the vehicle.

FIGURE 78. SAMPLE TENSOR HEADER CARD

ENRL10 = RL10 type engine

ENDENG = signals the end of engine (Block I) input data for this case

The variables that are included for input to the engine programs are:

a. <u>H1 Engine</u>: PPI, TPIR, TRSG, SG, DPI, DPD, DEL1, DEL2, DEL3, DLTP, C1, RGG, C3, C4, C5, C6, C7, DRI, DRD, DEL4, DEL5, DLTR, R, PTI, PTES, ATI, DW, TN, IPB, GR, XNTST, WMIS, ETAGR, HPAUX, DEL6, ATE, DELTX, RX, AXD, DEL7, PA, DEL8, KBAL, ETCF, EPS, PNS, ATC, FX, FEN, ETCS, XMRTC, INDPR, XMREN, DEL9, CON, CPCX, IINFL

b. <u>F1 Engine</u>: PPI, TPIR, TRSG, SG, DPI, DPD, DEL1, DEL2, DEL3, DLTP, C1, RGG, C3, C4, C5, C6, C7, DRI, DRD, DEL4, DEL5, DLTR, R, PTI, PTES, ATI, DW, IPB, GR, XNTST, WMIS, ETAGR, DEL9, DEL6, ATE, XMREN, RX, FEN, DEL7, PA, DEL8, KBAL, ETCF, EPS, PNS, ATC, INDPR, ETCS, XMRTC, CON, CPCX, TN, IINFL

c. <u>J2 Engine</u>: PPI, TPIS, TRSG, SG, DPI, DPD, DEL1, DEL2, DEL3, DEL4, DEL5, DEL7, DEL20, DEL21, DEL22, DEL23, DEL24, DEL25, DEL26, DEL27, DEL28, DEL29, DEL35, DLTP, C1, C3, C4, C5, C6, C8, DRI, DRD, DLTR, R, PTI, ATI, DW, IPB, XNTST, HPAUX, ATE, PA, KBAL, ETCF, EPS, PNS, ATC, FEN, ETCS, XMRTC, INDPR, XMREN, CON, IINFL, TTI, FAUX, CNST1, CNST2, CNST3, CNST4, CNST5, CNST6, EPSM, ABPNI, PUCOF, PUSEC, DMOO, DMFO, WPRESO, WPRESF

d. RL10 Engine: ATC, KBAL, PCIEL, XMRTC, Q, TPIS, TOL1, TOL2, XNPN, CON1, CON2, CON3, CON4, CON5, CD2, PPI, XNP, RHPI, TOL3, TOL4, TOL5, CON6, CON7, CON8, TOL6, TOL7, TOL9, FCL, AVT, HPAUX, ETGER, CON9, CON10, CON11, CON12, CON13, CON14, CON15, TOL10, CON, IINFL, EPS

The input data should be stacked with all the H1 data first, followed by all the F1, J2 and RL10 data for the first case. For subsequent cases, the procedure is repeated, starting with the TENSOR HEADER card.

After the nominal data for each engine has been input once, there is no need to input them again during one computer run. Only changes from the nominal need to be input, and these must be input for each stage of the vehicle. If a nominal case is to be run, one of the nominal input variables should be re-entered as change data. Also, if more than one stage of the vehicle is comprised of the same engine with an identical configuration, the change data for the subsequent engine(s) should contain the variable IINFL set equal to 1. This will allow the program to bypass recalculating the influence coefficients, and it will use the same set that was generated for the previous engine.

The very last card for the engine input data, for each case, should be a $ENDENG EEND = 0.$. This card signals the end of Block I data and the beginning of Block II data.

See Figure 79 for a sample deck setup.

Changing Variables for Subroutine INFL

If an addition, deletion, or change of any variables used to compute influence coefficients is desired for a specific engine, the following must be done:

a. The value of the independent or dependent variable must be set into Table TABI or TABD (removed for a deletion) and also reset from TABI with the perturbed value.

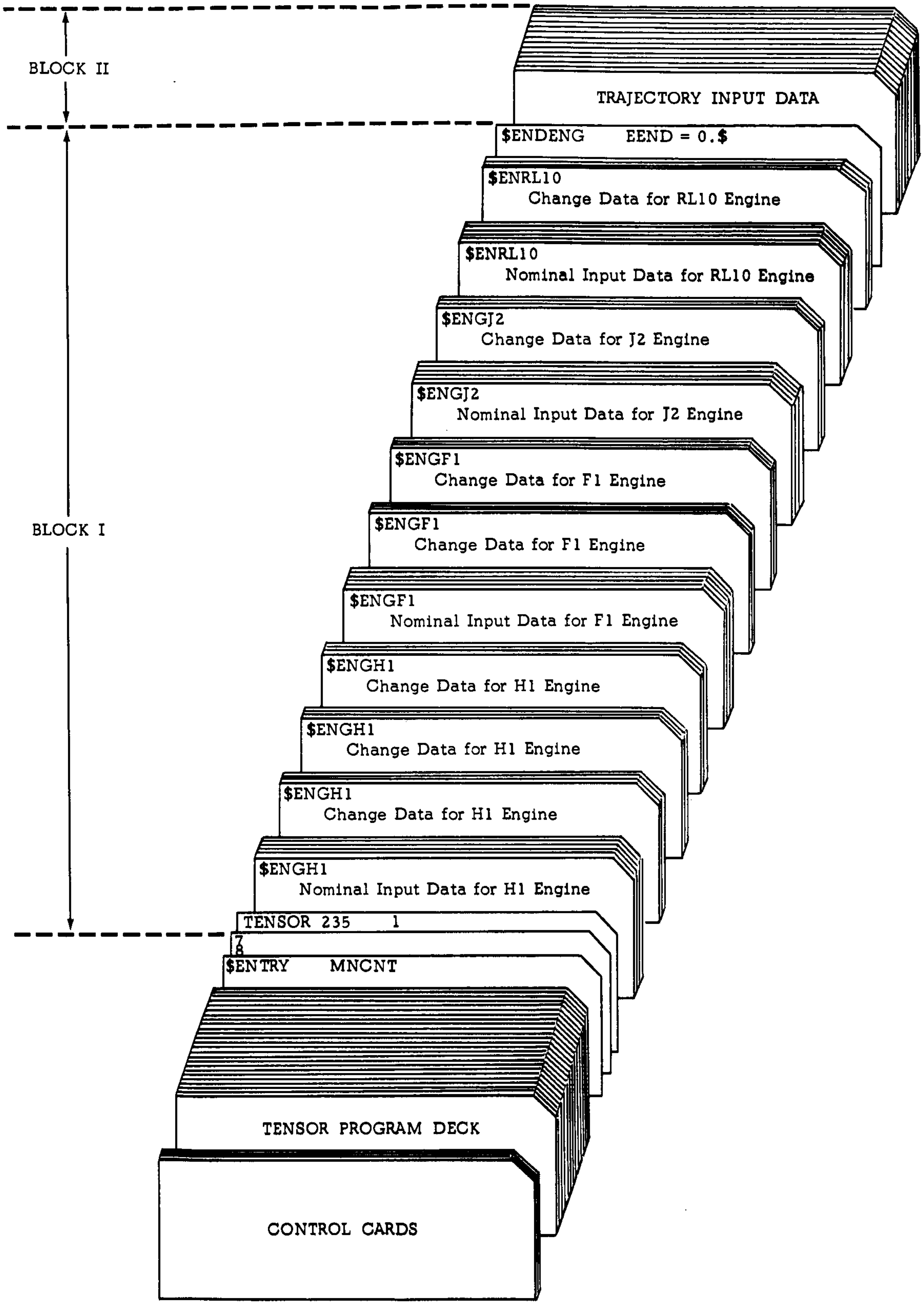


FIGURE 79. SAMPLE DECK SETUP FOR RUNNING OF ONE CASE THROUGH TENSOR PROGRAM

b. The appropriate counter, NIND or ND, must be modified (if needed).

c. Revise the appropriate DATA statement.

The program, as now written, includes the following independent and dependent variables used to compute influence coefficients:

H1 and F1 Engines

$$\left.\begin{array}{l}\text{PPI(1)}\\ \text{PPI(2)}\\ \text{TPIR(1)}\\ \text{TPIR(2)}\\ \text{ETCS}\\ \text{FEN}\end{array}\right\}\quad\text{Independent}$$

$$\left.\begin{array}{l}\text{WCO}\\ \text{WCF}\\ \text{WGO}\\ \text{WGF}\\ \text{PNS}\end{array}\right\}\quad\text{Dependent}$$

J2 Engine

$$\left.\begin{array}{l}\text{PPI(1)}\\ \text{PPI(2)}\\ \text{TPIS(1)}\\ \text{TPIS(2)}\\ \text{ETCS}\\ \text{FEN}\\ \text{PUVA}\end{array}\right\}\quad\text{Independent}$$

$$\left.\begin{array}{l}\text{WCO}\\ \text{WCF}\\ \text{WGO}\\ \text{WGF}\\ \text{PNS}\end{array}\right\}\quad\text{Dependent}$$

RL10 Engine

$$\left.\begin{array}{l}\mathrm{PPI(1)}\\ \mathrm{PPI(2)}\\ \mathrm{TPIS(1)}\\ \mathrm{TPIS(2)}\\ \mathrm{ETCS}\\ \mathrm{FCL}\end{array}\right\}\ \text{Independent}$$

$$\left.\begin{array}{l}\mathrm{WCO}\\ \mathrm{WCF}\\ \mathrm{PNSPR}\end{array}\right\}\ \text{Dependent}$$

SOLUTION METHOD

Determine if error indicator set from Subroutine ESTM.

```
1.  If F = 0,  go to 2
    If F ≠ 0,  go to 177
```

Read in TENSOR HEADER card and error check.

```
2.  READ ITENSR, ISTG, ITRJ

3.  L = 0

4.  If ISTG - 5 ≤ 0,  go to 5
    If ISTG - 5 > 0,  go to 174

5.  If ISTG ≤ 0,  go to 174
    If ISTG > 0,  go to 6

6.  Do 12 K = 2, 6

7.  If ISTG(K) < 0,  go to 174
    If ISTG(K) = 0,  go to 8
    If ISTG(K) > 0,  go to 10
```

```
8.  $L = 1$
9.  Go to 12
10. If $L \neq 0$, go to 174
    If $L = 0$, go to 11
11. If $ISTG(K) - 5 \leq 0$, go to 12
    If $ISTG(K) - 5 > 0$, go to 174
12. Continue
13. If $(ITENS - ITENSR) \neq 0$, go to 174
    If $(ITENS - ITENSR) = 0$, go to 14
```

Set loop control for engine number and stages.

```
14. Do 166  $J = 1, 4$
15. Do 167  $I = 1, 6$
```

Clear indicator flags.

```
16. $KEND = 0$
17. $JA = 0$
18. $JB = 0$
19. $JC = 0$
20. $JD = 0$
```

Check if stage 1 (2, 3, . . . , 6) equal to engine number 1 (2, 3, 4).

21. If ISTG(I) - J ≠ 0, go to 168

If ISTG(I) - J = 0, go to 22

Go to correct engine control section.

22. If J = 1, go to 23

If J = 2, go to 58

If J = 3, go to 93

If J = 4, go to 130

H1 Engine Control

Check if nominal input data has been read in previously.

23. If JJA = 0, go to 24

If JJA ≠ 0, go to 25

Read nominal data.

24. READ

Set nominal input data for re-use, or reset previously-set nominal data.

25. CALL SETNOM (J, JJA)

Set indicator to reset nominal values for re-entry to this engine subroutine.

26. JJA = 1

Read changes to nominal data.

27. READ

Print out input data.

28. CALL EPRINT(J, JA)

See whether engine routine to be bypassed.

29. If IINFL $\neq$ 0, go to 163
    If IINFL = 0, go to 30

Enter H1 engine subroutine.

30. CALL H1(NERR)

Check if output required.

31. If JA = 0, go to 32
    If JA $\neq$ 0, go to 34

Reset print control counter and print output.

32. JA = 1
33. CALL EPRINT (J, JA)
34. If NERR $\neq$ 0, go to 177
    If NERR = 0, go to 35

Check if Subroutine INFL to be entered.

35. If CON = 0, go to 168

If CON ≠ 0, go to 36

Set up independent variables for Subroutine INFL.

36. TABI(1) = PPI(2)

37. TABI(2) = PPI(1)

38. TABI(3) = TPIR(2)

39. TABI(4) = TPIR(1)

40. TABI(5) = ETCS

41. TABI(6) = FEN

Set counter for number of independent variables.

42. NIND = 6

Set up dependent variables for Subroutine INFL.

43. TABD(1) = WCO

44. TABD(2) = WCF

45. TABD(3) = WGO

46. TABD(4) = WGF

47. TABD(5) = PNS

Set counter for number of dependent variables.

48. ND = 5

Enter Subroutine INFL.

49. CALL INFL (CON, KEND, TABI, TABD, INDV, DEPV,
NIND, ND, Y, VIND, VDEP, NLIST(J))

Check if INFL has been entered for each independent variable.

50. If KEND = 0, go to 51
If KEND ≠ 0, go to 163

Reset independent variable with perturbed value.

51. PPI(2) = TABI(1)

52. PPI(1) = TABI(2)

53. TPIR(2) = TABI(3)

54. TPIR(1) = TABI(4)

55. ETCS = TABI(5)

56. FEN = TABI(6)

57. Go to 30

F1 Engine Control

58. If JJB = 0, go to 59
If JJB ≠ 0, go to 60

Read nominal data.

59. READ

Set nominal data for re-use, or reset previously-set nominal data.

60. CALL SETNOM (J, JJB)

Set indicator to reset nominal values for re-entry to to this engine subroutine.

61. JJB = 1

Read changes to nominal data.

62. READ

Print out input data.

63. CALL EPRINT (J, JB)

See if engine program to be bypassed.

64. If IINFL ≠ 0, go to 163

If IINFL = 0, go to 65

Enter F1 engine subroutine.

65. CALL F1(NERR)

Check if output required.

66. IF JB = 0, go to 67

If JB ≠ 0, go to 69

Reset print control counter and print output.

67. JB = 1

68. CALL EPRINT (J, JB)

Determine if error occurred in engine program.

69. If NERR $\neq$ 0, go to 177

If NERR = 0, go to 70

Determine if Subroutine INFL to be entered.

70. If CON = 0, go to 168

If CON $\neq$ 0, go to 71

Set up independent variables for Subroutine INFL.

71. TABI(1) = PPI(2)

72. TABI(2) = PPI(1)

73. TABI(3) = TPIR(2)

74. TABI(4) = TPIR(1)

75. TABI(5) = ETCS

76. TABI(6) = FEN

Set counter for number of independent variables.

77. NIND = 6

Set up dependent variables for Subroutine INFL.

78. TABD(1) = WCO

79. TABD(2) = WCF

80. TABD(3) = WGO

81. TABD(4) = WGF

82. TABD(5) = PNS

Set counter for number of dependent variables.

83. ND = 5

Enter Subroutine INFL.

84. CALL INFL (CON, KEND, TABI, TABD, INDV, DEPV,
NIND, ND, Y, VIND, VDEP, NLIST(J))

Determine if INFL has been entered for each independent variable.

85. If KEND = 0, go to 86

If KEND ≠ 0, go to 163

Reset independent variable with perturbed value.

86. PPI(2) = TABI(1)

87. PPI(1) = TABI(2)

88. TPIR(2) = TABI(3)

89. TPIR(1) = TABI(4)

90. ETCS = TABI(5)

91. FEN = TABI(6)

92. Go to 65

J2 Engine Control

Check if nominal input data has been read in previously.

93. If JJC = 0, go to 94

If JJC $\neq$ 0, go to 95

Read nominal data.

94. READ

Set nominal data for re-use, or reset previously-set nominal data.

95. CALL SETNOM (J, JJC)

Set indicator to reset nominal values for re-entry to this engine subroutine.

96. JJC = 1

Read changes to nominal data.

97. READ

Print out input data.

98. CALL EPRINT (J, JC)

See if engine subroutine to be bypassed.

99. If IINFL $\neq$ 0, go to 163
If IINFL = 0, go to 100

Enter J2 engine subroutine.

100. CALL J2 (NERR)

Determine if output required.

101. If JC = 0, go to 102
If JC $\neq$ 0, go to 104

Reset print control counter and print output.

102. JC = 1

103. CALL EPRINT (J, JC)

Determine if error occurred in engine program.

104. If NERR $\neq$ 0, go to 177
If NERR = 0, go to 105

Determine if Subroutine INFL to be entered.

105. If CON = 0, go to 168
If CON $\neq$ 0, go to 106

Set up independent variables for Subroutine INFL.

106. TABI(1) = PPI(2)

107. TABI(2) = PPI(1)

108. TABI(3) = TPIS(2)

109. TABI(4) = TPIS(1)

110. TABI(5) = ETCS

111. TABI(6) = PUVA

112. TABI(7) = FEN

Set counter for number of independent variables.

113. NIND = 7

Set up dependent variables for Subroutine INFL.

114. TABD(1) = WCO

115. TABD(2) = WCF

116. TABD(3) = WGO

117. TABD(4) = WGF

118. TABD(5) = PNS

Set counter for number of dependent variables.

119. ND = 5

Enter Subroutine INFL.

120. CALL INFL (CON, KEND, TABI, TABD, INDJ, DEPV,
NIND, ND, Y, VIND, VDEP, NLIST(J))

Determine if INFL has been entered for each independent variable.

121. If KEND = 0, go to 122

If KEND ≠ 0, go to 163

Reset independent variable with perturbed value.

122. PPI(2) = TABI(1)

123. PPI(1) = TABI(2)

124. TPIR(2) = TABI(3)

125. TPIR(1) = TABI(4)

126. ETCS = TABI(5)

127. PUVA = TABI(6)

128. FEN = TABI(7)

129. Go to 100

RL10 Engine Control

Check if nominal input data has been read in previously.

130. If JJD = 0, go to 131

If JJD ≠ 0, go to 132

Read nominal data.

131. READ

Set nominal data for re-use or reset previously-set nominal data.

132. CALL SETNOM (J, JJD)

Set indicator to reset nominal values for re-entry to this engine subroutine.

133. JJD = 1

Read changes to nominal data.

134. READ

Print out input data.

135. CALL EPRINT (J, JD)

See if engine program to be bypassed.

136. If IINFL $\neq$ 0, go to 162

If IINFL = 0, go to 137

Enter RL10 engine subroutine.

137. CALL RL10(NERR)

Determine if output required.

138. If JD = 0, go to 139

If JD $\neq$ 0, go to 141

Reset print control counter and print output.

139. JD = 1

140. CALL EPRINT (J, JD)

Determine if error occurred in engine program.

141. If NERR ≠ 0, go to 177

If NERR = 0, go to 142

Determine if Subroutine INFL to be entered.

142. If CON = 0, go to 168

If CON ≠ 0, go to 143

Set up independent variables for Subroutine INFL.

143. TABI(1) = PPI(2)

144. TABI(2) = PPI(1)

145. TABI(3) = TPIS(2)

146. TABI(4) = TPIS(1)

147. TABI(5) = ETCS

148. TABI(6) = FCL

Set counter for number of independent variables.

149. NIND = 6

Set up dependent variables for Subroutine INFL.

150. TABD(1) = WCO

151. TABD(2) = WCF

152. TABD(3) = PNSPR

Set counter for number of dependent variables.

153. ND = 3

Enter Subroutine INFL.

154. CALL INFL (CON, KEND, TABI, TABD, INDD, DEPD, NIND, ND, Y, VIND, VDEP, NLIST(J))

Determine if INFL has been entered for each independent variable.

155. If KEND = 0, go to 156

If KEND ≠ 0, go to 163

Reset independent variable with perturbed value.

156. PPI(2) = TABI(1)

157. PPI(1) = TABI(2)

158. TPIS(2) = TABI(3)

159. TPIS(1) = TABI(4)

160. ETCS = TABI(5)

```
161.  FCL = TABI(6)

162.  Go to 137
```

Enter Subroutine STINFL to relocate influence coefficients for use by the trajectory program.

```
163.  CALL STINFL

164.  EAE(1,I) = EPS

165.  EAE(2,I) = ATC

166.  EAE(3,I) = ETCF

167.  IINFL = 0

168.  Continue

169.  Continue
```

Read to end of case data.

```
170.  READ
```

Determine if trajectory program is to be called.

```
171.  If ITRJ ≤ 0, go to 173

      If ITRJ > 0, go to 172

172.  F = 1

173.  Return
```

Write error message.

```
174.  WRITE
```

Read to end of case data.

```
175.  READ
176.  Go to 179
```

Write error message.

```
177.  WRITE
178.  Go to 175
179.  If ITRJ $\le$ 0, go to 182
      If ITRJ $>$ 0, go to 180
180.  HEP = -10.0
181.  Go to 183
182.  HEP = 0.
```

Call Subroutine COMONS to return to MNCNTL.

```
183.  CALL COMONS (HEP)
184.  Return
185.  END
```

SUBROUTINE CNTROL NOMENCLATURE

Symbol	Description	Units	Reference
ATC	Thrust chamber expansion area ratio	in^2	/CCHAM/, 165
CON	Amount to perturb independent variables in Subroutine INFL	—	/ENGH1/, /ENGF1/, /ENGJ2/, /ENRL10/, /INCON/, 35, 49, 70, 84, 105, 120, 142, 154
DEPD	Array containing BCD representation of dependent variables used for Subroutine INFL	—	DIM, DATA, 154
DEPV	Array containing BCD representation of dependent variables used for Subroutine INFL	—	DIM, DATA, 49, 84, 120
EAE	Array for Block II usage	—	/INFLTB/, 164, 165, 166
EPS	Thrust chamber expansion area ratio	—	/CCHAM/, 164
ETCF	Thrust chamber nozzle efficiency	—	/CCHAM/, 166
ETCS	Thrust chamber combustion efficiency	—	/CCHAM/, /ENGH1/, /ENGF1/, /ENGJ2/, 40, 55, 75, 90, 126, 147, 160
F	Indicator to determine if Block II is to be run or if an error was detected during processing	—	CALL, 1, 172, 180
FCL	Loaded value of thrust (RL10)	lbf	/CHRL10/, /ENRL10/, 148, 161

SUBROUTINE CNTROL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
FEN	Nominal engine thrust	lb	/CCHAM/, /ENGH1/, /ENGF1/, /ENGJ2/, 41, 56, 76, 91, 112, 127
HEP	Error indicator	—	180, 182, 183
I	Loop control counter for vehicle stage numbers	—	15, 21
IINFL	Indicator to determine if the engine program is to by bypassed	—	/ENGH1/, /ENGF1/, /ENGJ2/, /ENRL10/, 29, 64, 99, 135
INDD	Array containing BCD representation of independent variables used for Subroutine INFL	—	DIM, DATA, 154
INDV	Array containing BCD representation of independent variables used for Subroutine INFL	—	DIM, DATA, 49, 84, 120
ISTG	Array containing engine control information from TENSOR HEADER card	—	/STAGES/, 2, 4, 5, 7, 11, 21
ITENS	Used for error checking of TENSOR HEADER card	—	DATA, 13
ITENSR	Used for error checking of TENSOR HEADER card	—	2, 13
ITRJ	Indicator to determine if Block II to be run	—	2, 171, 179
J	Loop control counter for engine code number	—	14, 21, 22, 25, 28, 33, 49, 60, 63, 68, 84, 95, 98, 103, 120, 132, 135, 140, 154

SUBROUTINE CNTROL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
JA	Print control indicator	—	17, 28; 31, 32, 33
JB	Print control indicator	—	18, 63, 66, 67, 68
JC	Print control indicator	—	19, 98, 101, 102, 103
JD	Print control indicator	—	20, 135, 138, 139, 140
JJA	Indicator used for setting or resetting nominal input data for H1 engine	—	DATA, 23, 25, 26
JJB	Indicator used for setting or resetting nominal input data for F1 engine	—	DATA, 58, 60, 61
JJC	Indicator used for setting or resetting nominal input data for J2 engine	—	DATA, 93, 95, 96
JJ2	Indicator used for setting or resetting nominal input data for RL10 engine	—	DATA, 129, 132, 133
K	Loop control counter	—	6, 7, 11
KEND	Terminating flag for Subroutine INFL	—	16, 49, 50, 84, 85, 120, 121, 154, 155
L	Indicator used in error checking of TENSOR HEADER card	—	3, 8, 10
ND	Number of dependent variables for Subroutine INFL	—	/INFLTB/, 48, 49, 83, 84, 119, 120, 153, 154

SUBROUTINE CNTROL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
NERR	Error indicator	—	30, 34, 65, 69, 100, 104, 137, 141
NIND	Number of independent variables for Subroutine INFL	—	/INFLTB/, 42, 49, 77, 84, 113, 120, 149, 154
NLIST	Array containing BCD representation of engine names for Subroutine INFL	—	49, 84, 120, 154
PNS	Thrust chamber nozzle stagnation pressure	psia	/CCHAM/, /ENGH1/, /ENGF1/, /ENGJ2/, 47, 82, 118
PNSPR	Thrust chamber nozzle stagnation pressure	lbf/in^2	/PNSPRI/, 152
PPI	Pump inlet total pressure	psia	/CPUMP/, /ENGH1/, /ENGF1/, /ENGJ2/, /ENRL10/, 36, 37, 51, 52, 71, 72, 86, 87, 122, 123, 143, 144, 156, 157
PUVA	Propellant utilization valve angle	deg	/CJ2IN/, 111, 127
TABD	Array containing the dependent variables for Subroutine INFL	—	DIM, 43, 44, 45, 46, 47, 49, 78, 79, 80, 81, 82, 84, 114, 115, 116, 117, 118, 120, 150, 151, 152, 154

SUBROUTINE CNTROL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TABI	Array containing the independent variables for Subroutine INFL	—	DIM, 36, 37, 38, 39, 40, 41, 49, 51, 52, 53, 54, 55, 56, 71, 72, 73, 74, 75, 76, 84, 86, 87, 88, 89, 90, 91, 106, 107, 108, 109, 110, 111, 120, 122, 123, 124, 125, 126, 127, 143, 144, 145, 146, 147, 148, 154, 156, 157, 158, 159, 160, 161
TPIR	Pump inlet propellant temperature	^oR	/CMAIN/, /ENGH1/, /ENGF1/, /ENGJ2/, 38, 39, 53, 54, 73, 74, 88, 89, 124, 125
TPIS	Pump inlet static temperature	^oR	/CPUMP/, /ENRL10/, 145, 146, 158, 159
VDEP	Storage for Subroutine INFL	—	/INFLTB/, 49, 84, 120, 154
VIND	Storage for Subroutine INFL	—	/INFLTB/, 49, 84, 120, 154
WCF	Thrust chamber fuel flow rate	lb/sec	/CCHAM/, 44, 79, 115, 151
WCO	Thrust chamber oxidizer flow rate	lb/sec	/CCHAM/, 43, 78, 114, 150
WGF	Gas generator fuel flow rate	lb/sec	/CMAIN/, 46, 81, 117
WGO	Gas generator oxidizer flow rate	lb/sec	/CMAIN/, 45, 80, 116
Y	Array containing influence coefficients	—	/INFLTB/, 49, 84, 120, 154

```
$IBFTC CNTRØL  FULIST,REF,M94,DECK
      SUBRØUTINE CNTRØL(F)
      CØMMØN /PNSPRI/ PNSPR
C
      CØMMØN /MNRL10/ PCIEL,Q(2),XMRTCL,PII(2)
C
      CØMMØN /LXRL10/ TØL3(1),TØL5(1),CD2(1),CØN1(1),CØN2(1),CØN3(1),
     1CØN4(1),CØN5(1)
C
      CØMMØN /FLRL10/ CØN8(1),CØN6(1),TØL4(1),FLØPT(1),CØN7(1),TØJE(1),
     1WCF1(1),TØL6(1),TØL7(1),CØN10(1),AVT(1)
C
      CØMMØN /CHRL10/ TØL8(1),TØL9(1),FCL(1)
C
      CØMMØN /PPRL10/ XRATIØ(2), TØL1, TØL2, XNPN(2)
C
      CØMMØN /TBRL10/CØN9(1),ETGER(1),CØN11(1),CØN12(1),CØN13(1),CØN14(1
     1),CØN15(1),TØL10(1), HP1(1),HP2(1)
C
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),     PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
C
      CØMMØN /CGPRØP/ C3(6),C4(6),C5(6),C6(6),CSG(1),GAMG(1),CPG(1),
     1XMWG(1),XMRG(1),RG(1),XØMG(1)
C
      CØMMØN /CRES/ PRI(20),TRIS(20),PVRI(20),RHRI(20),VHRI(20),WRD(20),
     1DRI(20),DEL4(20),TRDS(20),DLTR(20),RHRD(20),RHRA(20),PRD(20),
     2R(20),VHRD(20),DRD(20),PRDS(20),PVRD(20),DEL5(20),WRI(20),PRIS(20)
C
      CØMMØN /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
C
      CØMMØN /CTURBX/ TXD(1),TXI(1), DELTX(1),PXDS(1),PXIS(1),RX(1),
     1RGX(1),TXIS(1),WX(1),CSXD(1),XØMX(1),GAMX(1),XMXD(1),PRXD(1),
     2PXD(1),AXD(1),FX(1),TXDS(1),CPCX(1)
C
      CØMMØN /CCHAM/ INDPR(1),CF(1),CFTHE(1),ETCF(1),EPS(1),PNS(1),FC(1)
     1,ATC(1),FAUX(1),FEN(1),DEL7(1),CS(1),CSTHE(1),ETCS(1),WC(1),XMRTC(
     21),WCF(1),WCØ(1)
C
      CØMMØN /CBAL/ KBAL(1)  /CPA/ PA(1) /CRESGG/ RESØG(1),RESFG(1)
C
      CØMMØN /CMAIN/ WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),
     1TN(1),ØRES(4),WG(1),WGØ(1),WGF(1),PCIE(1),XMRE(1),PGG(1),XISPC(1),
     2XISPE(1),FE(1),WEØ(1),WEF(1),TPIR(2)
C
```

FIGURE 80. SYMBOLIC LISTING OF SUBROUTINE CNTROL

```
      CØMMØN /CDEN/TRSG(2),SG(2)
      CØMMØN /H1F1J2/ RGG(1),DEL8(1),DEL9(1),XMREN(1)
      CØMMØN /INCØN/ CØN
      CØMMØN /STAGES/ISTG(6)
      CØMMØN /INFLTB/ YY(10,10,6),VINDA(10,6),VDEPA(10,6),NINDA(6),NIND,
     1NDA(6),Y(10,10),VDEP(10),VIND(10),ND ,EAE(3,6)
      CØMMØN /CTURMP/ PP(2),C8(4,2)
      CØMMØN /CJ2IN/DEL20,DEL21,DEL22,DEL23,DEL24,DEL25,DEL26,CNST1,
     1CNST2,CNST3,CNST4,CNST5,CNST6,PUCØF(6,10),PUSEC(10),PUVA,
     2 EPSM,ABPNI, DMØØ, DMFØ  ,WPRESF, WPRESØ ,DEL27,DEL28,DEL29,CNST7
     3,DEL35
      DIMENSIØN NLIST(4) ,  INDJ(10)
      DIMENSIØN TABI(10),TABD(10),INDV(10),DEPV(10),INDD(10),DEPD(10)
      DATA (NLIST(I),I=1,4)/  5H   H1,5H   F1,5H   J2,6H RL10    /
      DATA (DEPV(I),I=1,10)/6H   WCØ,6H   WCF,6H   WGØ,6H   WGF,6H   PNS
     A,6HXXXXXX,6HXXXXXX,6HXXXXXX,6HXXXXXX,6HXXXXXX/
      DATA (DEPD(I),I=1,10)/6H   WCØ,6H   WCF,6H PNSPR,1HX,1HX,1HX,1HX,
     A1HX,1HX,1HX /
      DATA (INDV(I),I=1,10) /6HPPI(2),6HPPI(1),6HTPIR-2,6HTPIR-1,
     A6H  ETCS,6H   FEN,1HX,1HX,1HX,1HX      /
      DATA (INDD(I),I=1,10) /6HPPI(2),6HPPIY1H,6HTPIS-2,6HTPIS-1,
     A6H  ETCS,6H   FCL,1HX,1HX,1HX,1HX      /
      DATA (INDJ(I),I=1,10)/6HPPI(2),6HPPI(1),6HTPIS-2,6HTPIS-1,
     A6H  ETCS,6H  PUVA,6H   FEN,1HX,1HX,1HX              /
      DATA ITENS /6HTENSØR/
      DIMENSIØN IINFL(1)
      DATA IINFL /0/
      NAMELIST /ENGJ2/ TTI,  R  ,XMRTC,PNS,EPS,ETCS,ATC,ETCF,PA,KBAL,
     1FAUX,FEN,C3,C4,C5,C6,ATI,PTI,CNST1,XMREN,DEL20,DEL21,DEL22,DEL23,
     2DEL24,DEL25,DEL26,DLTP,POVA,DEL7,DPD,PPI,TPIS,DPI,IPB,TRSG,SG,
     3DEL1,DEL2,DEL3,C1,C8,DW,ATE,CNST4, XNTST,   EPSM,CNST5,HPAUX,CØN,
     4ABPNI,CNST6,PUCØF,PUSEC,DMØØ,DMFØ,WPRESØ,WPRESF,CNST2,CNST3,DEL4,
     5DEL5,DRI,DRD,DLTR ,INDPR ,DEL27,DEL28,DEL29,CNST7,DEL35,IINFL
      NAMELIST/ENRL10/ ATC,KBAL,PCIEL,XMRTC,Q,TPIS,TØL1,TØL2,XNPN,CØN1,
     1CØN2,CØN3,CØN4,CØN5,CD2,PPI,XNP,RHPI,TØL3,TØL4,TØL5,CØN6,CØN7,
     2CØN8,TØL6,TØL7,TØL8,TØL9,FCL,AVT,HPAUX,ETGER,CØN9,CØN10,CØN11,
     3CØN12,CØN13,CØN14,CØN15,TØL10,CØN,IINFL,EPS
      NAMELIST /ENGF1/ PPI,TPIR,TRSG,SG,DPI,DPD,DEL1,DEL2,DEL3,DLTP,C1,
     1RGG,C3,C4,C5,C6,C7,DRI,DRD,DEL4,DEL5,DLTR,R,PTI,PTES,ATI,DW,IPB,
     2GR,XNTST,WMIS,ETAGR,DEL9,DEL6,ATE,XMREN,RX,FEN,DEL7,PA,DEL8,KBAL,
     3ETCF,EPS,PNS,ATC,INDPR,ETCS,XMRTC,CØN,CPCX,TN,IINFL
      NAMELIST /ENGH1/ PPI,TPIR,TRSG,SG,DPI,DPD,DEL1,DEL2,DEL3,DLTP,
     1C1,RGG,C3,C4,C5,C6,C7,DRI,DRD,DEL4,DEL5,DLTR,R,PTI,PTES,ATI,DW,TN,
     2IPB,GR,XNTST,WMIS,ETAGR,HPAUX,DEL6,ATE,DELTX,RX,AXD,DEL7,PA,DEL8,
     3KBAL,ETCF,EPS,PNS,ATC,FX,FEN,ETCS,XMRTC,INDPR,XMREN,DEL9,CØN,CPCX
     4,IINFL
      NAMELIST /ENDENG/ EEND
      DATA NERR/0/, JJA/0/, JJB/0/, JJC/0/, JJD/0/
      IF(F)7777,9000,7777
 9000 READ(5,1)  ITENSR,(ISTG(I),I=1,6),ITRJ
    1 FØRMAT(A6,6I1,2X,I1)
      L = 0
```

FIGURE 80. SYMBOLIC LISTING OF SUBROUTINE CNTROL (cont.)

```
      IF(ISTG-5)89,89,9999
   89 IF(ISTG)9999,9999,90
   90 DØ 100 K = 2,6
      IF(ISTG(K))9999,91,95
   91 L = 1
      GØ TØ 100
   95 IF(L)9999,97,9999
   97 IF(ISTG(K) - 5)100,100,9999
  100 CØNTINUE
      IF(ITENS-ITENSR)9999,2,9999
    2 DØ 20 J=1,4
      DØ 10 I=1,6
      KEND = 0
      JA=0
      JB=0
      JC=0
      JD=0
      IF(ISTG(I)-J )10,8,10
    8 GØ TØ (3,4,5,6),J
    3 IF(JJA)31,30,31
   30 READ(5,ENGH1)
   31 CALL SETNØM(J,JJA)
      JJA = 1
      READ(5,ENGH1)
      CALL EPRINT(J,JA)
      IF(IINFL)9,39,9
   39 CALL H1(NERR)
      IF (JA)33,32,33
   32 JA = 1
      CALL EPRINT(J,JA)
   33 IF(NERR)7777,34,7777
   34 IF(CØN)35,10,35
C SET UP INDEPENDENT VARIABLES FØR SUBRØUTINE INFL *HI*
   35 TABI(1) = PPI(2)
      TABI(2) = PPI(1)
      TABI(3) = TPIR(2)
      TABI(4) = TPIR(1)
      TABI(5) = ETCS
      TABI(6) = FEN
C SET CØUNTER FØR NUMBER ØF INDEPENDENT VARIABLES  *HI*
      NIND=6
C SET UP DEPENDENT VAIRABLES FØR SUBRØUNTINE INFL  *HI*
      TABD(1) = WCØ
      TABD(2) = WCF
      TABD(3) = WGØ
      TABD(4) = WGF
      TABD(5) = PNS
C SET CØUNTER FØR NUMBER ØF DEPENDENT VARIABLES    *HI*
      ND = 5
      CALL INFL (CØN,KEND,TABI,TABD,INDV,DEPV,NIND,ND,Y,VIND,VDEP,
```

FIGURE 80. SYMBOLIC LISTING OF SUBROUTINE CNTROL (cont.)

```
     1NLIST(J) )
      IF(KEND)9,38,9
C RESET PERTURBED INDEPENDENT VARIABLES
   38 PPI(2) = TABI(1)
      PPI(1) = TABI(2)
      TPIR(2) = TABI(3)
      TPIR(1) = TABI(4)
      ETCS   = TABI(5)
      FEN    = TABI(6)
      GO TO 39
    4 IF(JJB)41,40,41
   40 READ(5,ENGF1)
   41 CALL SETNOM(J,JJB)
      JJB = 1
      READ(5,ENGF1)
      CALL EPRINT(J,JB)
      IF(IINFL)9,49,9
   49 CALL F1 (NERR)
      IF(JB)43,42,43
   42 JB = 1
      CALL EPRINT(J,JB)
   43 IF(NERR)7777,44,7777
   44 IF(CON)45,10,45
C SET UP INDEPENDENT VARIABLES FOR SUBROUTINE INFL *FI*
   45 TABI(1) = PPI(2)
      TABI(2) = PPI(1)
      TABI(3) = TPIR(2)
      TABI(4) = TPIR(1)
      TABI(5) = ETCS
      TABI(6) = FEN
C SET COUNTER FOR NUMBER OF INDEPENDENT VARIABLES  *FI*
      NIND = 6
C SET UP DEPENDENT VARIABLES FOR SUBROUTINE INFL   *FI*
      TABD(1) = WC0
      TABD(2) = WCF
      TABD(3) = WG0
      TABD(4) = WGF
      TABD(5) = PNS
C SET COUNTER FOR NUMBER OF DEPENDENT VARIABLES    *FI*
      ND = 5
      CALL INFL (CON,KEND,TABI,TABD,INDV,DEPV,NIND,ND,Y,VIND,VDEP,
     1NLIST(J) )
      IF(KEND)9,48,9
C RESET PERTURBED INDEPENDENT VARIABLES
   48 PPI(2) = TABI(1)
      PPI(1) = TABI(2)
      TPIR(2)= TABI(3)
      TPIR(1)= TABI(4)
      ETCS   = TABI(5)
      FEN    = TABI(6)
```

FIGURE 80. SYMBOLIC LISTING OF SUBROUTINE CNTROL (cont.)

```
      GO TO 49
    5 IF(JJC)51,50,51
   50 READ(5,ENGJ2)
   51 CALL SETNOM(J,JJC)
      JJC = 1
      READ(5,ENGJ2)
      CALL EPRINT(J,JC)
      IF(IINFL)9,59,9
   59 CALL J2(NERR)
      IF(JC)53,52,53
   52 JC = 1
      CALL EPRINT(J,JC)
   53 IF(NERR)7777,54,7777
   54 IF(CON)55,10,55
C SET UP INDEPENDENT VARIABLES FOR SUBROUTINE INFL *J2*
   55 TABI(1) = PPI(2)
      TABI(2) = PPI(1)
      TABI(3) = TPIS(2)
      TABI(4) = TPIS(1)
      TABI(5) = ETCS
      TABI(6) = FEN
      TABI(7) = PUVA
C  SET COUNTER FOR NUMBER OF INDEPENDENT VARIABLES  *J2*
      NIND = 7
C SET UP DEPENDENT VARIABLES FOR SUBROUTINE INFL   *J2*
      TABD(1) = WCO
      TABD(2) = WCF
      TABD(3) = WGO
      TABD(4) = WGF
      TABD(5) = PNS
C SET COUNTER FOR NUMBER OF DEPENDENTS VARIABLES   *J2*
      ND = 5
      CALL INFL (CON,KEND,TABI,TABD,INDV,DEPV,NIND,ND,Y,VIND,VDEP,
     1NLIST(J) )
      IF(KEND)9,58,9
C RESET PERTURBED INDEPENDENT VARIABLES
   58 PPI(2) = TABI(1)
      PPI(1) = TABI(2)
      TPIS(2)= TABI(3)
      TPIS(1)= TABI(4)
      ETCS   = TABI(5)
      FEN    = TABI(6)
      PUVA = TABI(7)
      GO TO 59
    6 IF(JJD)61,60,61
   60 READ(5,ENRL10)
   61 CALL SETNOM(J,JJD)
      JJD = 1
      READ(5,ENRL10)
      CALL EPRINT(J,JD)
```

FIGURE 80. SYMBOLIC LISTING OF SUBROUTINE CNTROL (cont.)

```
      IF(IINFL)9,69,9
   69 CALL RL10(NERR)
      IF(JD)63,62,63
   62 JD = 1
      CALL EPRINT(J,JD)
   63 IF (NERR)7777,64,7777
   64 IF(CØN)65,10,65
C SET UP INDEPENDENT VARIABLES FØR SUBRØUTINE INFL *RL10*
   65 TABI(1) = PPI(2)
      TABI(2) = PPI(1)
      TABI(3) = TPIS(2)
      TABI(4) = TPIS(1)
      TABI(5) = ETCS
      TABI(6) = FCL
C SET CØUNTER FØR NUMBER ØF INDEPENDENT VARIABLES  *RL10*
      NIND = 6
C SET UP DEPENDENT VARIABLES FØR SUBRØUTINE INFL   *RL10*
      TABD(1) = WCØ
      TABD(2) = WCF
      TABD(3) = PNSPR
C SET CØUNTER FØR NUMBER ØF DEPENDENT VARIABLES    *RL10*
      ND = 3
      CALL INFL (CØN,KEND,TABI,TABD,INDV,DEPV,NIND,ND,Y,VIND,VDEP,
     1NLIST(J) )
      IF(KEND)9,68,9
C RESET PERTURBED INDEPENDENT VARIABLES
   68 PPI(2) = TABI(1)
      PPI(1) = TABI(2)
      TPIS(2)= TABI(3)
      TPIS(1)= TABI(4)
      ETCS   = TABI(5)
      FCL    = TABI(6)
      GØ TØ 69
    9 CALL STINFL(I)
      EAE(1,I) = EPS
      EAE(2,I) = ATC
      EAE(3,I) = ETCF
      IINFL =0
   10 CØNTINUE
   20 CØNTINUE
   21 READ (5,ENDENG)
C  DATA SHØULD NØW BE PØSITIØNED FØR TRAJECTØRY PRØGRAM
      IF(ITRJ)75,75,70
   70 F = 1.
   75 RETURN
 9999 WRITE(6,9998)
 9998 FØRMAT(47H1  HEADER CARD IN ERRØR. CASE TERMINATED              )
   79 READ(5,ENDENG)
      GØ TØ 81
 7777 WRITE(6,9977)
```

FIGURE 80. SYMBOLIC LISTING OF SUBROUTINE CNTROL (cont.)

```
9977 FØRMAT(55HO ERRØR ENCØUNTERED DURING PRØCESSING. CASE TERMINATED )
     GØ TØ 79
  81 IF(ITRJ)178,178,77
  77 HEP = -10.
     GØ TØ 179
 178 HEP =0.
 179 CALL CØMØNS(HEP)
     RETURN
     END
```

FIGURE 80. SYMBOLIC LISTING OF SUBROUTINE CNTROL (cont.)

LXIV. SUBROUTINE H1

This subroutine provides the logic required to simulate the steady-state performance of an H-1 type engine system. It may be used to perform a balance or a rebalance. The variable KBAL is the balance/rebalance indicator which controls the program flow. When KBAL is set equal to 1, a balance is performed. When KBAL is set equal to 2, a rebalance is performed.

Figure 81 is a schematic of the type engine system which may be simulated.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL H1 (NEGGG)

This subroutine uses COMMON blocks CPUMP, CGPROP, CRES, CTURB, CTURBX, CCHAM, H1F1J2, CBAL, CRESGG and CMAIN.

SOLUTION METHOD

Set propellant density indicators:

1. KPO = INDPR

```
    2.  KPF = INDPR + 1

    Test for balance or rebalance:

    3.  If KBAL = 1, go to 6

        If KBAL = 2, go to 4

    Initialize variables for rebalance:

    4.  CALL SWITCH (6, DRI(7), DRI7, DRD(7), DRD7,

            DRI(10), DRI10, DRD(10), DRD10, R(11),

            RESFG, R(12), RESOG)

    5.  Go to 10

    Initialize variables for balance:

    6.  CALL SWITCH (5, DRI7, DRI(7), DRD7, DRD(7),

            DRI10, DRI(10), DRD10, DRD(10), T, TN)

    7.  Do 9, J = 1, 2

    8.  PPDA(J) = 0

    9.  ORES(J) = 0

Initialize control indicators:

10.     CALL SWITCH(10, IPBO, 0, IPBR, 0, IDLT, 0, IDLFE,

            0, IDLMRC, 0, IDLPX, 0, IDLPGF, 0, IDLPGO,

            0, IGGLUP, 0, IPB2, 0)
```

Convert pump inlet propellant temperatures from degrees Rankine to degrees Fahrenheit:

11. TPIS(1) = TPIR(1) - 459.688

12. TPIS(2) = TPIR(2) - 459.688

Compute gas generator gas properties as a function of combustion temperature:

13. CALL GPROP (T)

Test magnitude of IGGLUP. IGGLUP is set to 1 in a rebalance if a gas generator oxidizer side injector end pressure balance is not accomplished.

14. If IGGLUP $\leq$ 0, go to 15

If IGGLUP $>$ 0, go to 18

Compute thrust chamber performance:

15. FAUX = FX

16. CALL CHAMBR

17. PCIE = (1.084)(PNS)

Compute gas generator flow rates:

18. WG = (32.174)(ATI(1))(PTI(1))/(CSG)

19. WGF = (WG)/(XMRG + 1)

20. WGO = WG - WGF

Compute pump flow rates:

```
21.  WP(1)  =  WCF + WGF + WMIS(1)

22.  WP(2)  =  WCO + WGO + WMIS(2)
```

Compute engine flow rates:

```
23.  WEO  =  WCO + WGO

24.  WEF  =  WCF + WGF
```

Compute engine mixture ratio:

```
25.  XMRE  =  (WEO)/(WEF)
```

If in a balance, compare calculated engine mixture ratio with desired value. If the convergence criterion is not satisfied, re-evaluate thrust chamber mixture ratio:

```
26.  If KBAL = 1, go to 27
     If KBAL = 2, go to 30
27.  If |XMRE - XMREN| ≤ DEL9, go to 30
     If |XMRE - XMREN| > DEL9, go to 28
28.  CALL ESTM (IDLMRC, 2, XMRCT, DXMRCT,
          XMRE - XMREN, XMRTC, LUP1)
29.  Go to 15
```

Main Line Pressure Drop Computations

The pressure drops in the main lines are computed working from injector end pressure up to pump discharge. Subroutine SWITCH is used to initialize variables for Subroutine RES. Subroutine RES performs the actual pressure drop calculations.

```
Fuel side:

30.  TRIS(1) = TPIS(1) + DLTP(1)

31.  CALL SWITCH (3, PRI(1), PCIE, WRI(1), WCF,
         WRD(1), WEF)

32.  CALL RES (1, 1, -1, -1, KPF)

33.  CALL SWITCH (4, TRIS(2), TRDS(1), PRI(2), PRD(1),
         WRI(2), WRD(1), WRD(2), WP(1))

34.  CALL RES (2, 1, -1, -1, KPF)

35.  If KBAL = 1, go to 38
     If KBAL = 2, go to 36

36.  CALL SWITCH (4, TRIS(13), TRDS(2), PRI(13),
         PRD(2), WRI(13), WRD(2), WRD(13), WRI(13))

37.  CALL RES (13, 1, -1, -1, KPF)

Oxidizer side:

38.  TRIS(3) = TPIS(2) + DLTP(2)
```

```
39.  CALL SWITCH (3, PRI(3), PCIE, WRI(3), WCO, WRD(3), WEO)
40.  CALL RES (3, 2, -1, -1, KPO)
41.  CALL SWITCH (4, TRIS(4), TRDS(3), PRI(4), PRD(3), WRI(4), WRD(3), WRD(4), WP(2))
42.  CALL RES (4, 2, -1, -1, KPO)
43.  If KBAL = 1, go to 49
     If KBAL = 2, go to 44
44.  CALL SWITCH (4, TRIS(14), TRDS(4), PRI(14), PRD(4), WRI(14), WRD(4), WRD(14), WRI(14))
45.  CALL RES (14, 2, -1, -1, KPO)
46.  PPD(1) = PRD(13)
47.  PPD(2) = PRD(14)
48.  Go to 52
49.  PPD(1) = PRD(2)
50.  PPD(2) = PRD(4)
```

Pump Performance Computations

```
51.  JC = 1
```

Compute oxidizer pump performance:

```
52.  CALL PUMP (2, KPO)
```

Test magnitude of IPBO:

53. If IPBO = 1, go to 67

If IPBO ≠ 1, go to 54

Test magnitude of IPBR:

54. If IPBR = 1, go to 56

If IPBR = 1, go to 55

Set desired fuel pump speed equal to oxidizer pump speed:

55. XNPP(1) = XNP(2)

Compute fuel pump performance:

56. CALL PUMP (1, KPF)

Test magnitude of IPBR:

57. If IPBR = 1, go to 58

If IPBR ≠ 1, go to 67

Test for balance or rebalance:

58. If KBAL = 1, go to 59

If KBAL = 2, go to 64

If the computed fuel pump speed is greater than the desired fuel pump speed in a balance, the fuel pump, rather than the oxidizer pump, will have to determine pump speed:

```
59.  IPBO = 1

60.  XNPP(2) = XNP(1)

61.  IPB2 = IPB(2)

62.  IPB(2) = 1

63.  Go to 52
```

In a rebalance, iterate on thrust chamber mixture ratio to match pump speeds:

```
64.  CALL ESTM (IDLMRC, 2, XMRCT, DXMRCT,

          XNPP(1) - XNP(1), XMRTC, LUP5)

65.  IPBR = 0

66.  Go to 15
```

Test for balance or rebalance:

```
67.  If KBAL = 1, go to 68

     If KBAL = 2, go to 89
```

Main Line Orifice Requirements

```
68.  Do 88, J = 1,2
```

Compare actual pump head with pump head without main line orifice. For at least one pump these will be equal:

```
69.  If H(J) < HPRIM(J), go to 88

     If H(J) = HPRIM(J), go to 70

     If H(J) > HPRIM(J), go to 71
```

Test value of JC. JC is used for loop control and is initially set to 1 (Statement 51) before the pump computations are performed. The initial pump computations are based upon pump discharge pressures which do not include the effect of a main line orifice. Therefore, the pump discharge densities are not quite right. In order to obtain better values of pump discharge density, and pump performance, the pump computations are performed twice if required. The second time is based upon a more accurate pump discharge pressure.

70. If JC = 1, go to 88

If JC = 2, go to 71

Compute pump discharge pressure based upon actual pump head:

71. PPDA(J) = PPI(J) + (RHP(J))(H(J))/(144)

Test value of JC:

72. If JC = 1, go to 73

If JC = 2, go to 78

Compare pump discharge pressures with and without main line orificing. If the convergence criterion is not satisfied, recompute pump performance for the side (oxidizer or fuel) requiring an orifice:

73. If $|\text{PPDA(J)} - \text{PPD(J)}| \leq 1$, go to 79

If $|\text{PPDA(J)} - \text{PPD(J)}| > 1$, go to 74

74. PPD(J) = PPDA(J)

75. HPRIME = HPRIM(J)

76. JC = 2

```
77.  If J = 1, go to 56

     If J = 2, go to 52

78.  HPRIM(J) = HPRIME
```

Test value of J and determine main line orificing for either fuel or oxidizer side:

```
79.  If J = 1, go to 80

     If J = 2, go to 84

80.  ORES(1) = (72)(RHPD(1) + RHRD(2))(PPDA(1)

          - PRD(2))/(WP(1)$^2$ )

81.  R(13) = ORES(1)

82.  PPD(1) = PPDA(1)

83.  Go to 89

84.  ORES(2) = (72)(RHPD(2) + RHRD(4))(PPDA(2)

          - PRD(4))/(WP(2)$^2$ )

85.  R(14) = ORES(2)

86.  PPD(2) = PPDA(2)

87.  Go to 89

88.  Continue
```

Reset control parameters:

```
89.  CALL SWITCH (4, IPBR, 0, IPBO, 0, IDLMRC,

          0, IPB(2), IPB2)
```

90. If KBAL = 1, go to 91

If KBAL = 2, go to 118

Turbine Performance Computations

Compute turbine inlet temperature and turbine speed:

91. TTI(1) = T + 459.688

92. XNT(1) = (GR(1))(XNP(1))

Initialize variables for Subroutine TURBINE and determine turbine performance:

93. CALL SWITCH (4, CPT(1), CPG, RGT(1), RG,

GAMT(1), GAMG, CST(1), CSG)

94. CALL TURBIN(1)

Compute required turbine horsepower:

95. HPTR = (BHPP(1) + BHPP(2))(ETAGR(1))

Compare required turbine horsepower with available turbine horsepower. If the convergence criterion is not satisfied, iterate on turbine inlet pressure in a balance, or thrust chamber nozzle stagnation pressure in a rebalance:

96. If $|\text{HPTR} - \text{HPT}(1)| \leq$ DEL6, go to 107

If $|\text{HPTR} - \text{HPT}(1)| >$ DEL6, go to 97

97. If KBAL = 1, go to 98

If KBAL = 2, go to 100

```
 98.  PARAM = PTI(1)

 99.  Go to 101

100.  PARAM = PNS

101.  CALL ESTM (IDLT, 2, RTHP, DRTHP,
          HPTR - HPT(1), PARAM, LUP2)

102.  If KBAL = 1, go to 103
      If KBAL = 2, go to 105

103.  PTI(1) = PARAM

104.  Go to 18

105.  PNS = PARAM

106.  Go to 15
```

Initialize variables for Subroutine TURBEX and determine turbine exhaust system performance:

```
107.  CALL SWITCH (7, IDLT, 0, TXI, TTD(1), RGX, RG,
          TXIS, TTDS(1), WX, WT(1), XOMX, XOMG,
          GAMX, GAMG)

108.  CALL TURBEX
```

Compare turbine discharge static pressure with turbine exhaust duct inlet static pressure. If the convergence criterion is not satisfied, iterate on turbine discharge static pressure:

```
109.  If | PXIS - PTES(1) | ≤ DEL8, go to 112
      If | PXIS - PTES(1) | > DEL8, go to 110
```

```
110. CALL ESTM (IDLPX, 2, TPXT, DTPXT,
         PXIS - PTES(1), PTES(1), LUP3)

111. Go to 94
```

Compute engine thrust. In a balance, compare computed thrust with desired thrust. If the convergence criterion is not satisfied, iterate on thrust chamber nozzle stagnation pressure:

```
112. FE = FC + FX

113. IDLPX = 0

114. If KBAL = 1, go to 115
     If KBAL = 2, go to 157

115. If $| FE - FEN | \leq DEL7$, go to 118
     If $| FE - FEN | > DEL7$, go to 116

116. CALL ESTM (IDLFE, 2, FET, DFET, FE - FEN,
         PNS, LUP4)

117. Go to 15
```

Compute gas generator injector end pressure working from the turbine inlet:

```
118. $PGG = PTI(1) + (RG)(TTI(1))(RGG)(WG^2)$
         $/((20736)(PTI(1)))$
```

<u>Gas Generator Line Pressure Drop Computations</u>

Fuel side:

```
119.  CALL SWITCH (4, TRIS(5), TRDS(1), PRI(5), PRD(1), WRI(5), WEF, WRD(5), WGF)
120.  CALL RES (5, 1, -1, 1, KPF)
121.  CALL SWITCH (4, TRIS(6), TRDS(5), PRI(6), PRD(5), WRI(6), WRD(5), WRD(6), WRI(6))
122.  CALL RES (6, 1, 1, 1, KPF)
123.  If KBAL = 1, go to 124
      If KBAL = 2, go to 129
124.  CALL SWITCH (4, TRIS(7), TRDS(6), PRI(7), PGG, WRI(7), WGF, WRD(7), WRI(7))
125.  CALL RES (7, 1, 1, -1, KPF)
126.  $R(11) = (PRD(6) - PRD(7))(RHRD(6) + RHRD(7))(72)/(WGF^2)$
127.  RESFG = R(11)
128.  If RESFG < 0, go to 160
      If RESFG ≥ 0, go to 137
129.  CALL SWITCH (4, TRIS(11), TRDS(6), PRI(11), PRD(6), WRI(11), WRD(6), WRD(11), WRI(11))
130.  CALL RES (11, 1, 1, 1, KPF)
```

```
131. CALL SWITCH (6, TRIS(7), TRDS(11), PRI(7),
         PRD(11), WRI(7), WRD(11), WRD(7), WRI(7),
         DRI(7), DRD7, DRD(7), DRI7)

132. CALL RES (7, 1, 1, 1, KPF)
```

Compare the gas generator fuel side injector end pressure, as computed from pressure drop calculations through the gas generator bleed lines, with PGG. If the convergence criterion is not satisfied, iterate on turbine inlet pressure:

```
133. If $| PRD(7) - PGG | \le DEL4(7)$, go to 136
     If $| PRD(7) - PGG | > DEL4(7)$, go to 134

134. CALL ESTM (IDLPGF, 2, PGGFT, DPGGFT,
         PRD(7) - PGG, PTI, LUP6)

135. Go to 18
```

Reset gas generator fuel side injector end pressure balance iteration loop counter:

```
136. IDLPGF = 0
```

Oxidizer side:

```
137. CALL SWITCH (4, TRIS(8), TRDS(3), PRI(8),
         PRD(3), WRI(8), WEO, WRD(8), WGO)

138. CALL RES(8, 2, -1, 1, KPO)

139. CALL SWITCH (4, TRIS(9), TRDS(8), PRI(9),
         PRD(8), WRI(9), WRD(8), WRD(9), WRI(9))
```

```
140.  CALL RES (9, 2, 1, 1, KPO)

141.  If KBAL = 1, go to 142

      If KBAL = 2, go to 147

142.  CALL SWITCH (4, TRIS(10), TRDS(9), PRI(10),
          PGG, WRI(10), WGO, WRD(10), WGO)

143.  CALL RES (10, 2, 1, -1, KPO)

144.  R(12) = (PRD(9) - PRD(10))(RHRD(9) + RHRD(10))
          $(72)/(WGO^2)$

145.  RESOG = R(12)

146.  If RESOG < 0, go to 160

      If RESOG ≥ 0, go to 157

147.  CALL SWITCH (4, TRIS(12), TRDS(9), PRI(12),
          PRD(9), WRI(12), WRD(9), WRD(12), WRI(12))

148.  CALL RES (12, 2, 1, 1, KPO)

149.  CALL SWITCH (6, TRIS(10), TRDS(12), PRI(10),
          PRD(12), WRI(10), WRD(12), WRD(10), WRI(10),
          DRI(10), DRD10, DRD(10), DRI10)

150.  CALL RES (10, 2, 1, 1, KPO)
```

Compare the gas generator oxidizer side injector end pressure, as calculated from pressure drop calculations through the gas generator bleed lines, with PGG. If the convergence criterion is not satisfied, iterate on gas generator combustion temperature:

```
151.  If $| PRD(10) - PGG | \le DEL4(10)$, go to 155

      If $| PRD(10) - PGG | > DEL4(10)$, go to 152

152.  CALL ESTM (IDLPGO, 2, PGGOT, DPGGOT,
          PRD(10) - PGG, T, LUP7)

153.  IGGLUP = 1

154.  Go to 13
```

Reset gas generator oxidizer side injector end pressure balance iteration loop counter:

```
155.  IDLPGO = 0

156.  Go to 91
```

Compute thrust chamber and engine specific impulses:

```
157.  XISPC = (FC)/(WC)

158.  XISPE = (FE)/(WEO + WEF)

159.  Return
```

Set error indicator and print message if a gas generator orifice is negative:

```
160.  NEGGG = 1

161.  Write error message

      Return
```

SUBROUTINE H1 NOMENCLATURE

Symbol	Description	Units	Reference
ATI	Turbine inlet nozzle area	in^2	/CTURB/, 18
BHPP	Pump brake horsepower	bhp	/CPUMP/, 95
CPG	Gas generator gas specific heat	btu/lb-°R	/CGPROP/, 93
CPT	Turbine gas specific heat	btu/lb-°R	/CTURB/, 93
CSG	Gas generator gas characteristic velocity	ft/sec	/CGPROP/, 18, 93
CST	Turbine gas inlet characteristic velocity	ft/sec	/CTURB/, 93
DEL4	Resistance inlet static pressure tolerance	psi	/CRES/, 133, 151
DEL6	Turbine horsepower tolerance	bhp	/CMAIN/, 96
DEL7	Engine thrust tolerance	lb	/CCHAM/, 115
DEL8	Turbine exit static pressure tolerance	psi	/H1F1J2/, 109
DEL9	Engine mixture ratio tolerance	—	/H1F1J2/, 27
DFET	Storage array for dependent argument, FE-FEN, in Subroutine ESTM	—	DIM, 116
DLTP	Pump propellant temperature rise	°F	/CPUMP/, 30, 38

SUBROUTINE H1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DPGGFT	Storage array for dependent argument, PRD(7) - PGG, in Subroutine ESTM	—	DIM, 134
DPGGOT	Storage array for dependent argument, PRD(10) - PGG, in Subroutine ESTM	—	DIM, 152
DRD10	DRD(10)	in	4, 6, 149
DRD7	DRD(7)	in	4, 6, 131
DRD	Resistance exit diameter	in	/CRES/, 4, 6, 131, 149
DRI10	DRI(10)	in	4, 6, 149
DRI7	DRI(7)	in	4, 6, 131
DRI	Resistance inlet diameter	in	/CRES/, 4, 6, 131, 149
DRTHP	Storage array for dependent argument, HPTR - HPT(1), in Subroutine ESTM	—	DIM, 101
DTPXT	Storage array for dependent argument, PXIS - PTES(1), in Subroutine ESTM	—	DIM, 110
DXMRCT	Storage array for dependent arguments, XMRE - XMREN or XNPP(1) - XNP(1), in Subroutine ESTM	—	DIM, 28, 64
ETAGR	Turbine to pump gearing efficiency	—	/CMAIN/, 95
FAUX	Auxiliary engine thrust	lb	/CCHAM/, 15

SUBROUTINE H1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
FC	Thrust chamber thrust	lb	/CCHAM/, 112, 157
FEN	Nominal engine thrust	lb	/CCHAM/, 115, 116
FE	Engine thrust	lb	/CMAIN/, 112, 115, 116, 158
FET	Storage array for independent argument, PNS, in Subroutine ESTM	—	DIM, 116
FX	Turbine exhaust thrust	lb	/CTURBX/, 15, 112
GAMG	Gas generator gas specific heat ratio	—	/CGPROP/, 93, 107
GAMT	Turbine gas specific heat ratio	—	/CTURB/, 93
GAMX	Turbine exhaust gas specific heat ratio	—	/CTURBX/, 107
GR	Turbine to pump gear ratio	—	/CMAIN/, 92
HPRIME	Pump head without main line orificing	ft	75, 78
HPRIM	Pump head without main line orificing	ft	/CPUMP/, 69, 75, 78
HPTR	Required turbine horsepower	bhp	95, 96, 101
HPT	Turbine horsepower	bhp	/CTURB/, 96, 101
H	Pump head	ft	/CPUMP/, 69, 71
IDLFE	Engine thrust balance iteration counter	—	10, 116

SUBROUTINE H1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
IDLMRC	Engine mixture ratio balance or pump speed balance iteration counter	—	10, 28, 64, 89
IDLPGF	Gas-generator fuel-side injector-end balance iteration counter	—	10, 134, 136
IDLPGO	Gas-generator oxidizer-side injector-end balance iteration counter	—	10, 152, 155
IDLPX	Turbine discharge static pressure balance iteration counter	—	10, 110, 113
IDLT	Turbine horsepower balance iteration counter	—	10, 101, 107
IGGLUP	Gas-generator loop control indicator	—	10, 14, 153
INDPR	Propellant combination indicator	—	/CCHAM/, 1, 2
IPB2	IPB(2)	—	10, 61, 89
IPBO	Pump speed balance loop control indicator	—	10, 53, 59, 89
IPBR	Indicator used in pump speed balance loop	—	/CPUMP/, 10, 54, 57, 65, 89
IPB	Pump speed balance indicator	—	/CPUMP/, 61, 62, 89
JC	Pump discharge static pressure balance loop control indicator	—	51, 70, 72, 76

SUBROUTINE H1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
J	Loop control indicator	—	7, 8, 9, 68, 69, 71, 73, 74, 75, 77, 78, 79
KBAL	Engine balance/rebalance indicator	—	/CBAL/, 3, 26, 35, 43, 58, 67, 90, 97, 102, 114, 123, 141
KPF	Fuel propellant density indicator	—	2, 32, 34, 37, 56, 120, 122, 125, 130, 132
KPO	Oxidizer propellant density indicator	—	1, 40, 42, 45, 52, 138, 140, 143, 148, 150
LUP1	—	—	DATA, 28
LUP2	—	—	DATA, 101
LUP3	—	—	DATA, 110
LUP4	—	—	DATA, 116
LUP5	—	—	DATA, 64
LUP6	—	—	DATA, 134
LUP7	—	—	DATA, 152
NEGGG	Negative gas generator control orifice resistance indicator	—	CALL, 160
ORES	Main line control orifice resistance	sec^2/ft^5	/CMAIN/, 9, 80, 81, 84, 85
PARAM	Dummy parameter (PTI(1) or PNS)	—	98, 100, 101, 103, 105

SUBROUTINE H1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PCIE	Thrust chamber injector end pressure	psia	/CMAIN/, 17, 31, 39
PGGFT	Storage array for independent argument, PTI, in Subroutine ESTM	—	DIM, 134
PGGOT	Storage array for independent argument, T, in Subroutine ESTM	—	DIM, 152
PGG	Gas generator injector end pressure	psia	/CMAIN/, 118, 124, 133, 134, 142, 151, 152
PNS	Thrust chamber nozzle stagnation pressure	psia	/CCHAM/, 17, 100, 105, 116
PPDA	Pump discharge total pressure	psia	/CMAIN/, 8, 71, 73, 74, 80, 82, 84, 86
PPD	Pump discharge total pressure	psia	/CPUMP/, 46, 47, 49, 50, 73, 74, 82, 86
PPI	Pump inlet total pressure	psia	/CPUMP/, 71
PRD	Resistance exit total pressure	psia	/CRES/, 33, 36, 41, 44, 46, 47, 49, 50, 80, 84, 119, 121, 126, 129, 131, 133, 134, 137, 139, 144, 147, 149, 151, 152
PRI	Resistance inlet total pressure	psia	/CRES/, 31, 33, 36, 39, 41, 44, 119, 121, 124, 129, 131, 137, 139, 142, 147, 149

SUBROUTINE H1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PTES	Turbine exit static pressure	psia	/CTURB/, 109, 110
PTI	Turbine inlet total pressure	psia	/CTURB/, 18, 98, 103, 118, 134
PXIS	Turbine exhaust inlet static pressure	psia	/CTURBX/, 109, 110
RESFG	Gas generator fuel line control orifice resistance	sec^2/ft^5	/CRESGG/, 4, 127, 128
RESOG	Gas generator oxidizer line control orifice resistance	sec^2/ft^5	/CRESGG/, 4, 145, 146
RGG	Gas generator resistance	sec^2/ft^5	/H1F1J2/, 118
RG	Gas generator gas constant	ft/°R	/CGPROP/, 93, 107, 118
RGT	Turbine gas constant	ft/°R	/CTURB/, 93
RGX	Turbine exhaust gas constant	ft/°R	/CTURBX/, 107
RHPD	Pump discharge propellant density	lb/ft^3	/CPUMP/, 80, 84
RHP	Pump average propellant density	lb/ft^3	/CPUMP/, 71
RHRD	Resistance exit liquid density	lb/ft^3	/CRES/, 80, 84, 126, 144
R	Resistance value	sec^2/ft^5	/CRES/, 4, 81, 85, 126, 127, 144, 145
RTHP	Storage array for independent argument, PARAM, in Subroutine ESTM	—	DIM, 101

SUBROUTINE H1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TN	Nominal gas generator combustion temperature	°F	/CMAIN/, 6
TPIR	Pump inlet propellant temperature	°R	/CMAIN/, 11, 12
TPIS	Pump inlet propellant temperature	°F	/CPUMP/, 11, 12, 30, 38
TPXT	Storage array for independent argument, PTES(1), in Subroutine ESTM	—	DIM, 110
TRDS	Resistance exit liquid temperature	°F	/CRES/, 33, 36, 41, 44, 119, 121, 124, 129, 131, 137, 139, 142, 147, 149
TRIS	Resistance inlet liquid temperature	°F	/CRES/, 30, 33, 36, 38, 41, 44, 119, 121, 124, 129, 131, 137, 139, 142, 147, 149
T	Gas generator combustion temperature	°F	6, 13, 91, 152
TTD	Turbine exit total temperature	°R	/CTURB/, 107
TTDS	Turbine exit static temperature	°R	/CTURB/, 107
TTI	Turbine inlet total temperature	°R	/CTURB/, 91, 118
TXI	Turbine exhaust system inlet total temperature	°R	/CTURBX/, 107

SUBROUTINE H1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TXIS	Turbine exhaust system inlet static temperature	°R	/CTURBX/, 107
WCF	Thrust chamber fuel flow rate	lb/sec	/CCHAM/, 21, 24, 31
WCO	Thrust chamber oxidizer flow rate	lb/sec	/CCHAM/, 22, 23, 39
WC	Thrust chamber total flow rate	lb/sec	/CCHAM/, 157
WEF	Engine fuel flow rate	lb/sec	/CMAIN/, 24, 25, 31, 119, 158
WEO	Engine oxidizer flow rate	lb/sec	/CMAIN/, 23, 25, 39, 137, 158
WGF	Gas generator fuel flow rate	lb/sec	/CMAIN/, 19, 20, 21, 24, 119, 124, 126
WGO	Gas generator oxidizer flow rate	lb/sec	/CMAIN/, 20, 22, 23, 137, 142, 144
WG	Gas generator total flow rate	lb/sec	/CMAIN/, 18, 19, 20, 118
WMIS	Miscellaneous flow rate	lb/sec	/CMAIN/, 21, 22
WP	Pump mass flow rate	lb/sec	/CPUMP/, 21, 22, 33, 41, 80, 84
WRD	Resistance exit mass flow rate	lb/sec	/CRES/, 31, 33, 36, 39, 41, 44, 119, 121, 124, 129, 131, 137, 139, 142, 147, 149

SUBROUTINE H1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WRI	Resistance inlet mass flow rate	lb/sec	/CRES/, 31, 33, 36, 39, 41, 44, 119, 121, 124, 129, 131, 137, 139, 142, 147, 149
WT	Turbine gas flow rate	lb/sec	/CTURB/, 107
WX	Turbine exhaust flow rate	lb/sec	/CTURBX/, 107
XISPC	Thrust chamber specific impulse	sec	/CMAIN/, 157
XISPE	Engine specific impulse	sec	/CMAIN/, 158
XMRCT	Storage array for independent argument, XMRTC, in Subroutine ESTM	—	DIM, 28, 64
XMREN	Nominal engine mixture ratio	—	/H1F1J2/, 27, 28
XMRE	Engine mixture ratio	—	/CMAIN/, 25, 27, 28
XMRG	Gas generator mixture ratio	—	/CGPROP/, 19
XMRTC	Thrust chamber mixture ratio	—	/CCHAM/, 28, 64
XNPP	Pump speed	rpm	/CPUMP/, 55, 60, 64
XNP	Pump speed	rpm	/CPUMP/, 55, 60, 64, 92
XNT	Turbine speed	rpm	/CTURB/, 92
XOMG	Function of GAMG	—	/CGPROP/, 107
XOMX	Function of GAMX	—	/CTURBX/, 107

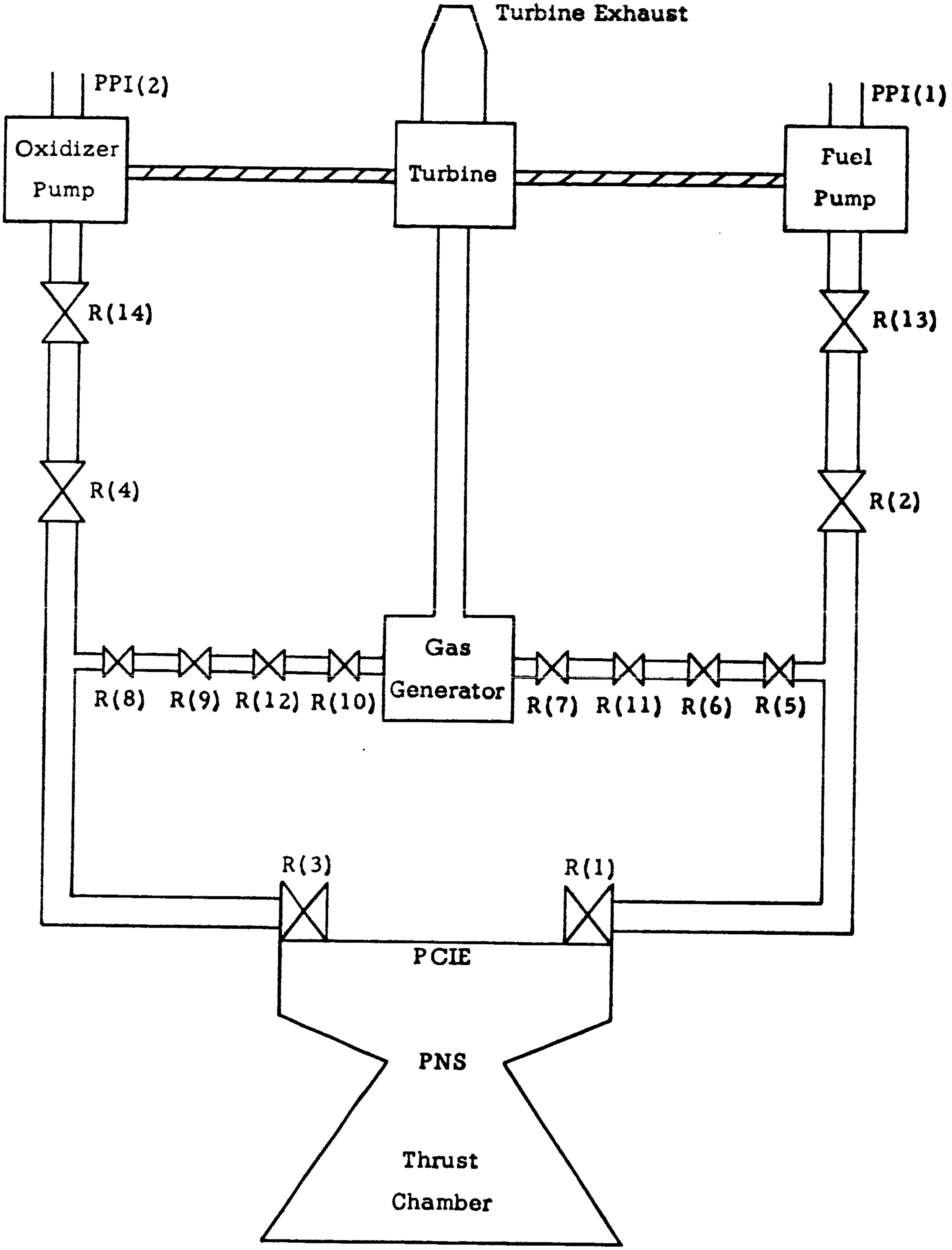


FIGURE 81. H-1 TYPE ENGINE SYSTEM SCHEMATIC

```
$IBFTC H1      REF,FULIST
      SUBROUTINE H1 (NEGGG)
      COMMON /H1F1J2/ RGG,DEL8,DEL9,XMREN
      COMMON /CPUMP/    PPI(2),TPIS(2),XNPSH(2),      PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
C
      COMMON /CGPROP/      C3(6),C4(6),C5(6),C6(6),CSG,GAMG,CPG,XMWG,
     1XMRG,RG,XOMG
C
      COMMON /CRES/ PRI(20),TRIS(20),PVRI(20),RHRI(20),VHRI(20),WRD(20),
     1DRI(20),DEL4(20),TRDS(20),DLTR(20),RHRD(20),RHRA(20),PRD(20),
     2R(20),VHRD(20),DRD(20),PRDS(20),PVRD(20),DEL5(20),WRI(20),PRIS(20)
C
      COMMON /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
C
      COMMON /CTURBX/ TXD,TXI,DELTX,PXDS,PXIS,RX,RGX,TXIS,WX,CSXD,XOMX,
     1GAMX,XMXD,PRXD,PXD,AXD,FX,TXDS,CPCX
C
      COMMON /CCHAM/  INDPR ,CF ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCO
C
      COMMON /CDEN/TRSG(2),SG(2)
      COMMON /CBAL/KBAL /CPA/PA /CRESGG/RESOG,RESFG
      COMMON /CMAIN/WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),TN,
     1ORES(4),WG,WGO,WGF,PCIE,XMRE,PGG,XISPC,XISPE,FE,WEO,WEF,TPIR(2)
C
      DIMENSION RTHP(20),DRTHP(20),TPXT(20),DTPXT(20)
      DIMENSION XMRCT(20),DXMRCT(20),FET(20),DFET(20)
      DIMENSION PGGFT(20),DPGGFT(20),PGGOT(20),DPGGOT(20)
      DATA LUP1/6H XMRE /,LUP2/6H HPT  /,LUP3/6H PTES /,LUP4/6H  FE   /
      DATA LUP5/6HXNP(1)/,LUP6/6H PGGF /,LUP7/6H PGGO /
C
      KPO=INDPR
      KPF=INDPR + 1
      GO TO (4,3),KBAL
    3 CALL SWITCH(6,DRI(7),DRI7,DRD(7),DRD7,DRI(10),DRI10,DRD(10),DRD10,
     1R(11),RESFG,R(12),RESOG)
      GO TO 14
    4 CALL SWITCH (5,DRI7,DRI(7),DRD7,DRD(7),DRI10,DRI(10),DRD10,DRD(10)
     1,T,TN)
   11 DO 12 J=1,2
      PPDA(J)=0.0
   12 ORES(J)=0.0
   14 CALL SWITCH(10,IPBO,0,IPBR,0,IDLT,0,IDLFE,0,IDLMRC,0,IDLPX,0,
     1IDLPGF,0,IDLPGO,0,IGGLUP,0,IPB2,0)
```

FIGURE 82. SYMBOLIC LISTING OF SUBROUTINE H1

```
C           IN GPROP T IS GIVEN IN DEGREES FAHRENHEIT
      TPIS(1) = TPIR(1) - 459.688
      TPIS(2)=TPIR(2)-459.688
   16 CALL GPROP(T)
      IF(IGGLUP)17,17,13
   17 FAUX=FX
      CALL CHAMBR
      PCIE=1.084*PNS
   13 WG=32.174*ATI(1)*PTI(1)/CSG
      WGF=WG/(XMRG+1.0)
      WGO=WG-WGF
      WP(1)=WCF+WGF+WMIS(1)
      WP(2)=WCO+WGO+WMIS(2)
      WEO=WCO+WGO
      WEF=WCF+WGF
      XMRE=WEO/WEF
      GO TO (18,21),KBAL
   18 IF(ABS(XMRE-XMREN)-DEL9)21,21,19
   19 CALL ESTM(IDLMRC,2,XMRCT,DXMRCT,XMRE-XMREN,XMRTC,LUP1)
      GO TO 17
   21 TRIS(1)=TPIS(1)+DLTP(1)
      CALL SWITCH (3,PRI(1),PCIE,WRI(1),WCF,WRD(1),WEF)
      CALL RES(1,1,-1,-1.0,KPF)
      CALL SWITCH (4,TRIS(2),TRDS(1),PRI(2),PRD(1),WRI(2),WRD(1),WRD(2),
     1WP(1))
      CALL RES(2,1,-1,-1.0,KPF)
      GO TO (23,22),KBAL
   22 CALL SWITCH (4,TRIS(13),TRDS(2),PRI(13),PRD(2),WRI(13),WRD(2),
     1WRD(13),WRI(13))
      CALL RES (13,1,-1,-1.0,KPF)
   23 TRIS(3)=TPIS(2)+DLTP(2)
      CALL SWITCH (3,PRI(3),PCIE,WRI(3),WCO,WRD(3),WEO)
      CALL RES(3,2,-1,-1.0,KPO)
      CALL SWITCH (4,TRIS(4),TRDS(3),PRI(4),PRD(3),WRI(4),WRD(3),
     1WRD(4),WP(2))
      CALL RES(4,2,-1,-1.0,KPO)
      GO TO  (25,24),KBAL
   24 CALL SWITCH (4,TRIS(14),TRDS(4),PRI(14),PRD(4),WRI(14),WRD(4),
     1WRD(14),WRI(14))
      CALL RES (14,2,-1,-1.0,KPO)
      PPD(1)=PRD(13)
      PPD(2)=PRD(14)
      GO TO 30
   25 PPD(1)=PRD(2)
      PPD(2)=PRD(4)
      JC = 1
   30 CALL PUMP(2,KPO)
      IF(IPBO-1)36,50,36
   36 IF(IPBR-1)38,40,38
   38 XNPP(1)=XNP(2)
```

FIGURE 82. SYMBOLIC LISTING OF SUBROUTINE H1 (cont.)

```
 40 CALL PUMP(1,KPF)
 42 IF(IPBR-1)50,45,50
 45 GO TO (46,47),KBAL
 46 IPBO=1
    XNPP(2)=XNP(1)
    IPB2 = IPB(2)
    IPB(2) =1
    GO TO 30
 47 CALL ESTM(IDLMRC,2,XMRCT,DXMRCT,XNPP(1 )-XNP(1),XMRTC,LUP5)
    IPBR=0
    GO TO 17
 50 GO TO(51,56),KBAL
 51 DO 60 J=1,2
    IF (H(J)-HPRIM(J))60,48,55
 48 GO TO (60,55),JC
 55 PPDA(J)=PPI(J)+RHP(J)*H(J)/144.0
    GO TO(52,53),JC
 52 IF(ABS(PPDA(J) -PPD(J)) - 1.0)59,59,54
 54 PPD(J) = PPDA(J)
    HPRIME = HPRIM(J)
    JC = 2
    GO TO(40,30),J
 53 HPRIM(J) = HPRIME
 59 GO TO(57,58),J
 57 ORES(1) = 72. * (RHPD(1)+RHRD(2))*(PPDA(1)-PRD(2))/WP(1)**2
    R(13) = ORES(1)
    PPD(1) = PPDA(1)
    GO TO 56
 58 ORES(2)=72.0*(RHPD(2)+RHRD(4))*(PPDA(2)-PRD(4))/WP(2)**2
    R(14) = ORES(2)
    PPD(2) = PPDA(2)
    GO TO 56
 60 CONTINUE
 56 CALL SWITCH(4,IPBR,0,IPBO,0,IDLMRC,0,IPB(2),IPB2 )
    GO TO(62,310),KBAL
 62 TTI(1) = T + 459.688
    XNT(1)=GR(1)*XNP(1)
    CALL SWITCH (4,CPT(1),CPG,RGT(1),RG,GAMT(1),GAMG,CST(1),CSG)
 70 CALL TURBIN(1)
    HPTR=(BHPP(1)+BHPP(2))*ETAGR(1)
    IF(ABS(HPTR-HPT(1))-DEL6)210,210,200
200 GO TO (171,173),KBAL
171 PARAM=PTI(1)
    GO TO 174
173 PARAM=PNS
174 CALL ESTM(IDLT,2,RTHP,DRTHP,HPTR-HPT(1),PARAM,LUP2)
    GO TO (175,176),KBAL
175 PTI(1)=PARAM
    GO TO 13
176 PNS=PARAM
```

FIGURE 82. SYMBOLIC LISTING OF SUBROUTINE H1 (cont.)

```
      GO TO 17
  210 CALL SWITCH(7,IDLT,0,TXI,TTD(1),RGX,RG,TXIS,TTDS(1),WX,WT(1),
     1XOMX,XOMG,GAMX,GAMG)
      CALL TURBEX
      IF(ABS(PXIS-PTES(1))-DEL8)230,230,220
  220 CALL ESTM(IDLPX,2,TPXT,DTPXT,PXIS-PTES(1),PTES(1),LUP3)
      GO TO 70
  230 FE=FC+FX
      IDLPX=0
      GO TO (240,318),KBAL
  240 IF(ABS(FE-FEN)-DEL7)310,310,300
  300 CALL ESTM(IDLFE,2,FET,DFET,FE-FEN,PNS,LUP4)
      GO TO 17
  310 PGG=PTI(1)+RG*TTI(1)*RGG*WG**2/PTI(1)/20736.0
  311 CALL SWITCH(4,TRIS(5),TRDS(1),PRI(5),PRD(1),WRI(5),WEF,WRD(5),WGF)
      CALL RES(5,1,-1,1.0,KPF)
      CALL SWITCH (4,TRIS(6),TRDS(5),PRI(6),PRD(5),WRI(6),WRD(5),WRD(6),
     1WRI(6))
      CALL RES(6,1,1,1.0,KPF)
      GO TO (312,313),KBAL
  312 CALL SWITCH(4,TRIS(7),TRDS(6),PRI(7),PGG,WRI(7),WGF,WRD(7),WRI(7))
      CALL RES(7,1,1,-1.0,KPF)
      R(11)=(PRD(6)-PRD(7))*(RHRD(6)+RHRD(7))*72./WGF**2
      RESFG = R(11)
      IF(RESFG)450,314,314
  313 CALL SWITCH (4,TRIS(11),TRDS(6),PRI(11),PRD(6),WRI(11),WRD(6),
     1WRD(11),WRI(11))
      CALL RES (11,1,1,1.0,KPF)
      CALL SWITCH (6,TRIS(7),TRDS(11),PRI(7),PRD(11),WRI(7),WRD(11),
     1WRD(7),WRI(7),DRI(7),DRD7,DRD(7),DRI7)
      CALL RES (7,1,1,1.0,KPF)
      IF(ABS(PRD(7)-PGG)-DEL4(7))320,320,315
  315 CALL ESTM(IDLPGF,2,PGGFT,DPGGFT,PRD(7)-PGG,PTI,LUP6)
      GO TO 13
  320 IDLPGF=0
  314 CALL SWITCH(4,TRIS(8),TRDS(3),PRI(8),PRD(3),WRI(8),WEO,WRD(8),WGO)
      CALL RES(8,2,-1,1.0,KPO)
      CALL SWITCH (4,TRIS(9),TRDS(8),PRI(9),PRD(8),WRI(9),WRD(8),WRD(9),
     1WRI(9))
      CALL RES(9,2,1,1.0,KPO)
      GO TO (316,317),KBAL
  316 CALL SWITCH(4,TRIS(10),TRDS(9),PRI(10),PGG,WRI(10),WGO,WRD(10),WGO
     1)
      CALL RES(10,2,1,-1.0,KPO)
      R(12)=(PRD(9)-PRD(10))*(RHRD(9)+RHRD(10))*72./WGO**2
      RESOG = R(12)
      IF(RESOG)450,318,318
  317 CALL SWITCH (4,TRIS(12),TRDS(9),PRI(12),PRD(9),WRI(12),WRD(9),
     1WRD(12),WRI(12))
      CALL RES (12,2,1,1.0,KPO)
```

FIGURE 82. SYMBOLIC LISTING OF SUBROUTINE H1 (cont.)

```
      CALL SWITCH (6,TRIS(10),TRDS(12),PRI(10),PRD(12),WRI(10),WRD(12),
     1WRD(10),WRI(10),DRI(10),DRD10,DRD(10),DRI10)
      CALL RES (10,2,1,1.0,KPO)
      IF(ABS(PRD(10)-PGG)-DEL4(10))321,321,322
  322 CALL ESTM (IDLPGO,2,PGGOT,DPGGOT,PRD(10)-PGG,T,LUP7)
      IGGLUP=1
      GO TO 16
  321 IDLPGO=0
      GO TO 62
  318 XISPC=FC/WC
      XISPE=FE/(WEO+WEF)
      RETURN
  450 NEGGG= 1
      WRITE(6,451)
  451 FORMAT(/////20X,47HA GAS GENERATOR ORIFICE RESISTANCE IS NEGATIVE
     1       /////)
  998 RETURN
      END
```

FIGURE 82. SYMBOLIC LISTING OF SUBROUTINE H1 (cont.)

LXV. SUBROUTINE F1

This subroutine provides the logic required to simulate the steady-state performance of an F-1 type engine system. It may be used to perform a balance or a rebalance. The variable KBAL is the balance/rebalance indicator. When KBAL is set equal to 1, a balance is performed. When KBAL is set equal to 2, a rebalance is performed.

Figure 83 is a schematic of the type engine system which may be simulated.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL F1 (NEGGG)

This subroutine uses COMMON blocks CPUMP, CGPROP, CRES, CTURB, CTURBX, CCHAM, H1F1J2, CBAL, CRESGG and CMAIN.

SOLUTION METHOD

Set propellant density indicators:

1. KPO = INDPR
2. KPF = INDPR + 1

Test for balance or rebalance:

```
3.  If KBAL = 1, go to 6
    If KBAL = 2, go to 4
```

Initialize variables for rebalance:

```
4.  CALL SWITCH (6, DRI(10), DRI10, DRD(10), DRD10,
         DRI(12), DRI12, DRD(12), DRD12, R(17),
         RESFG, R(18), RESOG)
5.  Go to 7
```

Initialize variables for balance:

```
6.  CALL SWITCH (11, DRI10, DRI(10), DRD10, DRD(10),
         DRI12, DRI(12), DRD12, DRD(12), PPDA(1), 0,
         PPDA(2), 0, ORES(1), 0, ORES(2), 0, ORES(3),
         0, ORES(4), 0, T, TN)
```

Initialize control indicators:

```
7.  CALL SWITCH(10,IPBO, 0, IPBR, 0, IDLT, 0, IDLFE,
         0, IDLMRC, 0, IDLPX, 0, IDLPGF, 0, IDLPGO,
         0, IGGLUP, 0, IPB2, 0)
```

Convert pump inlet propellant temperatures from degrees Rankine to degrees Fahrenheit:

8. TPIS(1) = TPIR(1) - 459.688

9. TPIS(2) = TPIR(2) - 459.688

Compute gas generator gas properties as a function of combustion temperature:

10. CALL GPROP(T)

Test magnitude of IGGLUP. IGGLUP is set to 1 in a rebalance if a gas generator oxidizer side injector end pressure balance is not accomplished:

11. If IGGLUP $\leq$ 0, go to 12

If IGGLUP $>$ 0, go to 15

Compute thrust chamber performance:

12. FAUX = 0

13. CALL CHAMBR

14. PCIE = (1.146)(PNS)

Compute gas generator flow rates:

15. WG = (32.174)(ATI(1))(PTI(1))/(CSG)

16. WGF = (WG)/(XMRG + 1)

17. WGO = WG - WGF

Compute pump flow rates:

18. WP(1) = WCF + WGF + WMIS(1)

19. WP(2) = WCO + WGO + WMIS(2)

Compute engine flow rates:

20. WEO = WCO + WGO

21. WEF = WCF + WGF

Compute engine mixture ratio:

22. XMRE = (WEO)/(WEF)

If in a balance, compare calculated engine mixture ratio with desired value. If the convergence criterion is not satisfied, re-evaluate thrust chamber mixture ratio:

23. If KBAL = 1, go to 24

    If KBAL = 2, go to 27

24. If $|\text{XMRE} - \text{XMREN}| \le$ DEL9, go to 27

    If $|\text{XMRE} - \text{XMREN}| >$ DEL9, go to 25

25. CALL ESTM (IDLMRC, 2, XMRCT, DXMRCT,

    XMRE - XMREN, XMRTC, LUP1)

26. Go to 12

Main Line Pressure Drop Computations

The pressure drops in the main lines are computed working from injector end pressure up to pump discharge. Subroutine SWITCH is used to initialize variables for Subroutine RES. Subroutine RES performs the actual pressure drop calculations.

Fuel side:

```
27.  TRIS(1) = TPIS(1) + DLTP(1)
28.  CALL SWITCH (3, PRI(1), PCIE, WRI(1), WCF,
          WRD(1), WCF)
29.  CALL RES (1, 1, -1, -1.0, KPF)
30.  CALL SWITCH (7, TRIS(2), TRDS(1), PRI(2), PRD(1),
          TRIS(3), TRDS(1), PRI(3), PRD(1), WRI(2),
          (WRD(1))/(2), WRD(2), WRI(2), WRI(3), WRI(2))
31.  CALL SWITCH (3, WRD(3), WRI(3) + WGF, WDOT,
          WRI(2), IDLW, 0)
32.  CALL RES (2, 1, -1, -1.0, KPF)
33.  CALL RES (3, 1, -1, -1.0, KPF)
34.  CALL SWITCH (4, TRIS(4), TRDS(3), PRI(4), PRD(3),
          WRI(4), WRD(3), WRD(4), WRI(4))
35.  CALL RES (4, 1, -1, -1.0, KPF)
36.  If KBAL = 1, go to 43
     If KBAL = 2, go to 37
37.  CALL SWITCH (8, TRIS(13), TRDS(2), PRI(13), PRD(2),
          WRI(13), WRD(2), WRD(13), WRI(13), TRIS(14),
          TRDS(4), PRI(14), PRD(4), WRI(14), WRD(4),
          WRD(14), WRI(14))
```

```
38. CALL RES (13, 1, -1, -1.0, KPF)

39. CALL RES (14, 1, -1, -1.0, KPF)
```

Compare pressures in each main fuel line at the fuel pump discharge juncture. If the convergence criterion is not satisfied, iterate on the flow rates through these lines:

```
40. If $|PRD(13) - PRD(14)| \le DEL4(13)$, go to 47
    If $|PRD(13) - PRD(14)| > DEL4(13)$, go to 41

41. CALL ESTM (IDLW, 2, WDOTT, DWDOTT,
        PRD(13) - PRD(14), WDOT, LUP8)

42. Go to 45

43. If $|PRD(4) - PRD(2)| \le DEL4(2)$, go to 47
    If $|PRD(4) - PRD(2)| > DEL4(2)$, go to 44

44. CALL ESTM (IDLW, 2, WDOTT, DWDOTT,
        PRD(4) - PRD(2), WDOT, LUP8)

45. CALL SWITCH (4, WRI(2), WDOT, WRI(3), WCF - WDOT,
        WRD(2), WRI(2), WRD(3), WEF - WDOT)

46. Go to 32
```

Oxidizer side:

```
47. TRIS(5) = TPIS(2) + DLTP(2)

48. CALL SWITCH (3, PRI(5), PCIE, WRI(5), WCO,
        WRD(5), WRI(5))
```

```
49.  CALL RES (5, 2, -1, -1.0, KPO)

50.  CALL SWITCH (7, TRIS(6), TRDS(5), PRI(6), PRD(5),
          WRI(6), (WRD(5))/(2), WRD(6), WRI(6), TRIS(7),
          TRDS(5), PRI(7), PRD(5), WRI(7), WRI(6))

51.  CALL SWITCH (3, WRD(7), WRI(7) + WGO, WDOT,
          WRI(6), IDLW, 0)

52.  CALL RES (6, 2, -1, -1.0, KPO)

53.  CALL RES (7, 2, -1, -1.0, KPO)

54.  CALL SWITCH (4, TRIS(8), TRDS(7), PRI(8), PRD(7),
          WRI(8), WRD(7), WRD(8), WRI(8))

55.  CALL RES (8, 2, -1, -1.0, KPO)

56.  If KBAL = 1, go to 63
     If KBAL = 2, go to 57

57.  CALL SWITCH (8, TRIS(15), TRDS(8), PRI(15), PRD(8),
          WRI(15), WRD(8), WRD(15), WRI(15), TRIS(16),
          TRDS(6), PRI(16), PRD(6), WRI(16), WRD(6),
          WRD(16), WRI(16))

58.  CALL RES (15, 2, -1, -1.0, KPO)

59.  CALL RES (16, 2, -1, -1.0, KPO)
```

Compare pressures in each main oxidizer line at the oxidizer pump juncture. If the convergence criterion is not satisfied, iterate on the flow rates through these lines:

```
60.  If $|PRD(15) - PRD(16)| \le DEL4(15)$, go to 67
     If $|PRD(15) - PRD(16)| > DEL4(15)$, go to 61

61.  CALL ESTM (IDLW, 2, WDOTT, DWDOTT,
         PRD(15) - PRD(16), WDOT, LUP9)

62.  Go to 65

63.  If $|PRD(6) - PRD(8)| \le DEL4(6)$, go to 70
     If $|PRD(6) - PRD(8)| > DEL4(6)$, go to 64

64.  CALL ESTM (IDLW, 2, WDOTT, DWDOTT,
         PRD(6) - PRD(8), WDOT, LUP9)

65.  CALL SWITCH (4, WRI(6), WDOT, WRI(7),
         WCO - WDOT, WRD(6), WRI(6),
         WRD(7), WEO - WDOT)

66.  Go to 52

67.  PPD(1) = PRD(13)

68.  PPD(2) = PRD(15)

69.  Go to 73

70.  PPD(1) = PRD(2)
```

71. PPD(2) = PRD(6)

Pump Performance Computations

72. JC = 1

Compute oxidizer pump performance:

73. CALL PUMP (2, KPO)

Test magnitude of IPBO:

74. If IPBO = 1, go to 88

If IPBO ≠ 1, go to 75

Test magnitude of IPBR:

75. If IPBR = 1, go to 77

If IPBR ≠ 1, go to 76

Set desired fuel pump speed equal to oxidizer pump speed:

76. XNPP(1) = XNP(2)

Compute fuel pump performance:

77. CALL PUMP (1, KPF)

Test magnitude of IPBR:

78. If IPBR = 1, go to 79

If IPBR ≠ 1, go to 88

Test for balance or rebalance:

```
79.  If KBAL = 1, go to 80
     If KBAL = 2, go to 85
```

If the computed fuel pump speed is greater than the desired fuel pump speed in a balance, the fuel pump, rather than the oxidizer pump will have to determine pump speed:

```
80.  IPBO = 1
81.  XNPP(2) = XNP(1)
82.  IPB2 = IPB(2)
83.  IPB(2) = 1
84.  Go to 73
```

In a rebalance, iterate on thrust chamber mixture ratio to match pump speeds:

```
85.  CALL ESTM (IDLMRC, 2, XMRCT, DXMRCT,
          XNPP(1) - XNP(1), XMRTC, LUP5)
86.  IPBR = 0
87.  Go to 12
```

Test for balance or rebalance:

```
88.  If KBAL = 1, go to 89
     If KBAL = 2, go to 113
```

Main Line Orifice Requirements

```
89.  Do 112, J = 1,2
```

Compare actual pump head with pump head without main line orifice. For at least one pump these will be equal:

```
90.  If H(J) $\leq$ HPRIM(J), go to 112
     If H(J) $>$ HPRIM(J), go to 91
```

Compute pump discharge pressures based upon actual pump head:

```
91.  PPDA(J) = PPI(J) + (RHP(J))(H(J))/(144)
```

Test value of JC. JC is used for loop control and is initially set to 1 (Statement 72) before the pump computations are performed. The initial pump computations are based upon pump discharge pressures which do not include the effect of a main line orifice. Therefore, the pump discharge densities are not quite right. In order to obtain better values of pump discharge density, and pump performance, the pump computations are performed twice if required. The second time is based upon a more accurate pump discharge pressure:

```
92.  If JC = 1, go to 93
     If JC = 2, go to 98

93.  If $|$PPDA(J) - PPD(J)$|$ $\leq$ 1, go to 99
     If $|$PPDA(J) - PPD(J)$|$ $>$ 1, go to 94

94.  PPD(J) = PPDA(J)

95.  HPRIME = HPRIM(J)
```

```
96.  JC = 2
97.  If J = 1, go to 77
     If J = 2, go to 73
98.  HPRIM(J) = HPRIME
```

Test value of J and determine main line orificing for either fuel or oxidizer side:

```
99.  If J = 1, go to 100
     If J = 2, go to 106
100. ORES(1) = (72)(RHPD(1) + RHRD(2))
         $(PPDA(1) - PRD(2))/(WRD(2)^2)$
101. ORES(2) = (72)(RHPD(1) + RHRD(4))
         $(PPDA(1) - PRD(4))/(WRD(4)^2)$
102. R(13) = ORES(1)
103. R(14) = ORES(2)
104. PPD(1) = PPDA(1)
105. Go to 113
106. ORES(3) = (72)(RHPD(2) + RHRD(6))
         $(PPDA(2) - PRD(6))/(WRD(6)^2)$
107. ORES(4) = (72)(RHPD(2) + RHRD(8))
         $(PPDA(2) - PRD(8))/(WRD(8)^2)$
108. R(15) = ORES(3)
```

```
109.  R(16) = ORES(4)

110.  PPD(2) = PPDA(2)

111.  Go to 113

112.  Continue
```

Reset control parameters:

```
113.  CALL SWITCH (4, IPBR, 0, IPBO, 0, IDLMRC,
           0, IPB(2), IPB2)
```

Test for balance or rebalance:

```
114.  If KBAL = 1, go to 115
      If KBAL = 2, go to 142
```

Turbine Performance Computations

Compute turbine inlet temperature and turbine speed:

```
115.  TTI(1) = T + 459.688

116.  XNT(1) = (GR(1))(XNP(1))
```

Initialize variables for Subroutine TURBINE and determine turbine performance:

```
117.  CALL SWITCH (4, CPT(1), CPG, RGT(1), RG,
           GAMT(1), GAMG, CST(1), CSG)

118.  CALL TURBIN(1)
```

Compute required turbine horsepower:

```
119.  HPTR = (BHPP(1) + BHPP(2))(ETAGR(1))
```

Compare required turbine horsepower with available turbine horsepower. If the convergence criterion is not satisfied, iterate on turbine inlet pressure in a balance, or thrust chamber nozzle stagnation pressure in a rebalance:

```
120.  If $|HPTR - HPT(1)| \leq DEL6$, go to 131
      If $|HPTR - HPT(1)| > DEL6$, go to 121

121.  If KBAL = 1, go to 122
      If KBAL = 2, go to 124

122.  PARAM = PTI(1)

123.  Go to 125

124.  PARAM = PNS

125.  CALL ESTM (IDLT, 2, RTHP, DRTHP,
           HPTR - HPT(1), PARAM, LUP2)

126.  If KBAL = 1, go to 127
      If KBAL = 2, go to 129

127.  PTI(1) = PARAM

128.  Go to 15

129.  PNS = PARAM
```

```
130.  Go to 12
```

Initialize variables for Subroutine TURBEX and determine turbine exhaust system performance:

```
131.  CALL SWITCH (3, IDLT, 0, RGX, RG, WX, WT(1))

132.  CALL TURBEX
```

Compare turbine discharge static pressure with turbine exhaust duct inlet static pressure. If the convergence criterion is not satisfied, iterate on turbine discharge static pressure:

```
133.  If | PXIS - PTES(1) | ≤ DEL8, go to 136

      If | PXIS - PTES(1) | > DEL8, go to 134

134.  CALL ESTM (IDLPX, 2, TPXT, DTPXT,

          PXIS - PTES(1), PTES(1), LUP3)

135.  Go to 118
```

Compute engine thrust. In a balance, compare computed thrust with desired thrust. If the convergence criterion is not satisfied, iterate on thrust chamber nozzle stagnation pressure:

```
136.  FE = FC

137.  IDLPX = 0

138.  If KBAL = 1, go to 139

      If KBAL = 2, go to 177
```

139. If $| FE - FEN | \leq DEL7$, go to 142

If $| FE - FEN | > DEL7$, go to 140

140. CALL ESTM (IDLFE, 2, FET, DFET, FE - FEN,
PNS, LUP4)

141. Go to 12

Compute gas generator injector end pressure working from the turbine inlet:

142. $PGG = PTI(1) + (RG)(TTI(1))(RGG)$
$(WG^2)/((PTI(1))(20736))$

Gas Generator Line Pressure Drop Computations

Fuel side:

143. CALL SWITCH (4, TRIS(9), TRDS(3), PRI(9),
PRDS(3), WRI(9), WGF, WRD(9), WGF)

144. CALL RES (9, 1, -1, 1, KPF)

145. If KBAL = 1, go to 146

If KBAL = 2, go to 151

146. CALL SWITCH (4, TRIS(10), TRDS(9), PRI(10), PGG,
WRI(10), WGF, WRD(10), WGF)

147. CALL RES (10, 1, -1, -1, KPF)

```
148. $R(17) = (PRD(9) - PRD(10))(RHRD(9) + RHRD(10))(72)/(WGF^2)$
149. $RESFG = R(17)$
150. If $RESFG < 0$, go to 180
     If $RESFG \geq 0$, go to 159
151. CALL SWITCH (4, TRIS(17), TRDS(9), PRI(17), PRD(9), WRI(17), WGF, WRD(17), WGF)
152. CALL RES (17, 1, 1, 1, KPF)
153. CALL SWITCH (6, TRIS(10), TRDS(17), PRI(10), PRD(17), WRI(10), WGF, WRD(10), WGF, DRI(10), DRD10, DRD(10), DRI10)
154. CALL RES (10, 1, 1, 1, KPF)
```

Compare the gas generator fuel side injector end pressure, as computed from pressure drop calculations through the gas generator bleed lines, with PGG. If the convergence criterion is not satisfied, iterate on turbine inlet pressure:

```
155. If $| PRD(10) - PGG | \leq DEL4(10)$, go to 158
     If $| PRD(10) - PGG | > DEL4(10)$, go to 156
156. CALL ESTM (IDLPGF, 2, PGGFT, DPGGFT, PRD(10) - PGG, PTI, LUP6)
157. Go to 15
```

Reset gas generator fuel side injector end pressure balance iteration loop counter:

```
158.  IDLPGF = 0
```

Oxidizer side:

```
159.  CALL SWITCH (4, TRIS(11), TRDS(7), PRI(11), PRDS(7),
          WRI(11), WGO, WRD(11), WGO)
160.  CALL RES (11, 2, -1, 1, KPO)
161.  If KBAL = 1, go to 162
      If KBAL = 2, go to 167
162.  CALL SWITCH (4, TRIS(12), TRDS(11), PRI(12), PGG,
          WRI(12), WGO, WRD(12), WGO)
163.  CALL RES (12, 2, 1, -1, KPO)
164.  R(18) = (PRD(11) - PRD(12))(RHRD(11) + RHRD(12))
          (72)/($WGO^2$)
165.  RESOG = R(18)
166.  If RESOG < 0, go to 180
      If RESOG ≥ 0, go to 177
167.  CALL SWITCH (4, TRIS(18), TRDS(11), PRI(18), PRD(11),
          WRI(18), WGO, WRD(18), WGO)
168.  CALL RES (18, 2, 1, 1, KPO)
```

```
169.  CALL SWITCH (6, TRIS(12), TRDS(18), PRI(12),
          PRD(18), WRI(12), WGO, WRD(12), WGO,
          DRI(12), DRD12, DRD(12), DRI12)

170.  CALL RES (12, 2, 1, 1, KPO)
```

Compare the gas generator oxidizer side injector end pressure, as computed from pressure drop calculations through the gas generator bleed lines, with PGG. If the convergence criterion is not satisfied, iterate on gas generator combustion temperature:

```
171.  If | PRD(12) - PGG | ≤ DEL4(12), go to 175
      If | PRD(12) - PGG | > DEL4(12), go to 172

172.  CALL ESTM (IDLPGO, 2, PGGOT, DPGGOT,
          PRD(12) - PGG, T, LUP7)

173.  IGGLUP = 1

174.  Go to 10
```

Reset gas generator oxidizer side injector end pressure balance iteration loop counter:

```
175.  IDLPGO = 0

176.  Go to 115
```

Compute thrust chamber and engine specific impulses:

```
177.  XISPC = (FC)/(WC)
```

178. XISPE = (FE)/(WEO + WEF)

179. Return

Set error indicator and print message if a gas generator orifice is negative:

180. NEGGG = 1

181. Write error message

Return

SUBROUTINE F1 NOMENCLATURE

Symbol	Description	Units	Reference
ATI	Turbine inlet nozzle area	in^2	/CTURB/, 15
BHPP	Pump brake horsepower	bhp	/CPUMP/, 119
CPG	Gas generator gas specific heat	btu/lb-°R	/CGPROP/, 117
CPT	Turbine gas specific heat	btu/lb-°R	/CTURB/, 117
CSG	Gas generator gas characteristic velocity	ft/sec	/CGPROP/, 15, 117
CST	Turbine gas inlet characteristic velocity	ft/sec	/CTURB/, 117
DEL4	Resistance inlet static pressure tolerance	psi	/CRES/, 40, 43, 60, 63, 155, 171
DEL6	Turbine horsepower tolerance	bhp	/CMAIN/, 120
DEL7	Engine thrust tolerance	lb	/CCHAM/, 139
DEL8	Turbine exit static pressure tolerance	psi	/H1F1J2/, 133
DEL9	Engine mixture ratio tolerance	—	/H1F1J2/, 24
DFET	Storage array for dependent argument, FE - FEN, in Subroutine ESTM	—	DIM, 140
DLTP	Pump propellant temperature rise	°F	/CPUMP/, 27, 47

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DPGGFT	Storage array for dependent argument, PRD(10) - PGG, in Subroutine ESTM	—	DIM, 156
DPGGOT	Storage array for dependent argument, PRD(12) - PGG, in Subroutine ESTM	—	DIM, 172
DRD10	DRD(10)	in	4, 6, 153
DRD12	DRD(12)	in	4, 6, 169
DRD	Resistance exit diameter	in	/CRES/, 4, 6, 153, 169
DRI10	DRI(10)	in	4, 6, 153
DRI12	DRI(12)	in	4, 6, 169
DRI	Resistance inlet diameter	in	/CRES/, 4, 6, 153, 169
DRTHP	Storage array for dependent argument, HPTR - HPT(1), in Subroutine ESTM	—	DIM, 125
DTPXT	Storage array for dependent argument, PXIS - PTES(1), in Subroutine ESTM	—	DIM, 134
DWDOTT	Storage array for dependent arguments, PRD(13) - PRD(14), PRD(4) - PRD(2), PRD(15) - PRD(16), or PRD(6) - PRD(8), in Subroutine ESTM	—	DIM, 41, 44, 61, 64

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DXMRCT	Storage array for dependent arguments, XMRE - XMREN or XNPP(1) - XNP(1), in Subroutine ESTM	—	DIM, 25, 85
ETAGR	Turbine to pump gearing efficiency	—	/CMAIN/, 119
FAUX	Auxiliary engine thrust	lb	/CCHAM/, 12
FC	Thrust chamber thrust	lb	/CCHAM/, 136, 177
FEN	Nominal engine thrust	lb	/CCHAM/, 139, 140
FE	Engine thrust	lb	/CMAIN/, 136, 139, 140, 178
FET	Storage array for independent argument, PNS, in Subroutine ESTM	—	DIM, 140
GAMG	Gas generator gas specific heat ratio	—	/CGPROP/, 117
GAMT	Turbine gas specific heat ratio	—	/CTURB/, 117
GR	Turbine to pump gear ratio	—	/CMAIN/, 116
HPRIME	Pump head without main line orificing	ft	95, 98
HPRIM	Pump head without main line orificing	ft	/CPUMP/, 90, 95, 98
HPTR	Required turbine horsepower	bhp	119, 120, 125

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
HPT	Turbine horsepower	bhp	/CTURB/, 120, 125
H	Pump head	ft	/CPUMP/, 90, 91
IDLFE	Engine thrust balance iteration counter	—	7, 140
IDLMRC	Engine mixture ratio balance or pump speed balance iteration counter	—	7, 25, 85, 113
IDLPGF	Gas-generator fuel-side injector-end balance iteration counter	—	7, 156, 158
IDLPGO	Gas-generator oxidizer-side injector-end balance iteration counter	—	7, 172, 175
IDLPX	Turbine discharge static pressure balance iteration counter	—	7, 134, 137
IDLT	Turbine horsepower balance iteration counter	—	7, 125, 131
IDLW	Main oxidizer and fuel line pressure balance iteration counter	—	31, 41, 44, 51, 61, 64
IGGLUP	Gas-generator loop control indicator	—	7, 11, 173
INDPR	Propellant combination indicator	—	/CCHAM/, 1, 2
IPB2	IPB(2)	—	7, 82, 113

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
IPBO	Pump speed balance loop control indicator	—	7, 74, 80, 113
IPBR	Indicator used in pump speed balance loop	—	/CPUMP/, 7, 75, 78, 86, 113
IPB	Pump speed balance indicator	—	/CPUMP/, 82, 83, 113
JC	Pump discharge static pressure balance loop control indicator	—	72, 92, 96
J	Loop control indicator	—	89, 90, 91, 93, 94, 95, 97, 98, 99
KBAL	Engine balance/rebalance indicator	—	/CBAL/, 3, 23, 36, 56, 79, 88, 114, 121, 126, 138, 145, 161
KPF	Fuel propellant density indicator	—	2, 29, 32, 33, 35, 38, 39, 77, 144, 147, 152, 154
KPO	Oxidizer propellant density indicator	—	1, 49, 52, 53, 55, 58, 59, 73, 160, 163, 168, 170
LUP1	—	—	DATA, 25
LUP2	—	—	DATA, 125
LUP3	—	—	DATA, 134
LUP4	—	—	DATA, 140
LUP5	—	—	DATA, 85

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
LUP6	—	—	DATA, 156
LUP7	—	—	DATA, 172
LUP8	—	—	DATA, 41, 44
LUP9	—	—	DATA, 61, 64
NEGGG	Negative gas generator control orifice resistance indicator	—	CALL, 180
ORES	Main line control orifice resistance	sec^2/ft^5	/CMAIN/, 6, 100, 101, 102, 103, 106, 107, 108, 109
PARAM	Dummy parameter (PTI (1) pr PNS)	—	122, 124, 125, 127, 128
PCIE	Thrust chamber injector end pressure	psia	/CMAIN/, 14, 28, 48
PGGFT	Storage array for independent argument, PTI, in Subroutine ESTM	—	DIM, 156
PGGOT	Storage array for independent argument, T, in Subroutine ESTM	—	DIM, 172
PGG	Gas generator injector end pressure	psia	/CMAIN/, 142, 146, 155, 156, 162, 171, 172
PNS	Thrust chamber nozzle stagnation pressure	psia	/CCHAM/, 14, 124, 128, 140
PPDA	Pump discharge total pressure	psia	/CMAIN/, 6, 91, 93, 94, 100, 101, 104, 106, 107, 110

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PPD	Pump discharge total pressure	psia	/CPUMP/, 67, 68, 70, 71, 93, 94, 104, 110
PPI	Pump inlet total pressure	psia	/CPUMP/, 91
PRD	Resistance exit total pressure	psia	/CRES/, 30, 34, 37, 40, 41, 43, 44, 50, 54, 57, 60, 61, 63, 64, 67, 68, 70, 71, 100, 101, 106, 107, 148, 151, 153, 155, 156, 164, 167, 169, 171, 172
PRDS	Resistance exit static pressure	psia	/CRES/, 143, 159
PRI	Resistance inlet total pressure	psia	/CRES/, 28, 30, 34, 37, 48, 50, 54, 57, 143, 146, 151, 153, 159, 162, 167, 169
PTES	Turbine exit static pressure	psia	/CTURB/, 133, 134
PTI	Turbine inlet total pressure	psia	/CTURB/, 15, 122, 127, 142, 156
PXIS	Turbine exhaust inlet static pressure	psia	/CTURBX/, 133, 134
RESFG	Gas generator fuel line control orifice resistance	sec^2/ft^5	/CRESGG/, 4, 149, 150
RESOG	Gas generator oxidizer line control orifice resistance	sec^2/ft^5	/CRESGG/, 4, 165, 166
RGG	Gas generator resistance	sec^2/ft^5	/H1F1J2/, 142

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RG	Gas generator gas constant	ft/°R	/CGPROP/, 117, 131, 142
RGT	Turbine gas constant	ft/°R	/CTURB/, 117
RGX	Turbine exhaust gas constant	ft/°R	/CTURBX/, 131
RHPD	Pump discharge propellant density	lb/ft^3	/CPUMP/, 100, 101, 106, 107
RHP	Pump average propellant density	lb/ft^3	/CPUMP/, 91
RHRD	Resistance exit liquid density	lb/ft^3	/CRES/, 100, 101, 106, 107, 148, 164
R	Resistance value	sec^2/ft^5	/CRES/, 4, 102, 103, 108, 109, 148, 149, 164, 165
RTHP	Storage array for independent argument, PARAM, in Subroutine ESTM	—	DIM, 125
TN	Nominal gas generator combustion temperature	°F	/CMAIN/, 6
TPIR	Pump inlet propellant temperature	°R	/CMAIN/, 8, 9
TPIS	Pump inlet propellant temperature	°F	/CPUMP/, 8, 9, 27, 47
TPXT	Storage array for independent argument, PTES(1), in Subroutine ESTM	—	DIM, 134

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TRDS	Resistance exit liquid temperature	°F	/CRES/, 30, 34, 37, 50, 54, 57, 143, 146, 151, 153, 159, 162, 167, 169
TRIS	Resistance inlet liquid temperature	°F	/CRES/, 27, 30, 34, 37, 47, 50, 54, 57, 143, 146, 151, 153, 159, 162, 167, 169
T	Gas generator combustion temperature	°F	6, 10, 115, 172
TTI	Turbine inlet total temperature	°R	/CTURB/, 115, 142
WCF	Thrust chamber fuel flow rate	lb/sec	/CCHAM/, 18, 21, 28, 45
WCO	Thrust chamber oxidizer flow rate	lb/sec	/CCHAM/, 19, 20, 48, 65
WC	Thrust chamber total flow rate	lb/sec	/CCHAM/, 177
WDOT	WRI(2) or WRI(6)	lb/sec	31, 41, 44, 45, 51, 61, 64, 65
WDOTT	Storage array for independent argument, WDOT, in Subroutine ESTM	—	DIM, 41, 44, 61, 64
WEF	Engine fuel flow rate	lb/sec	/CMAIN/, 21, 22, 45, 178
WEO	Engine oxidizer flow rate	lb/sec	/CMAIN/, 20, 22, 65, 178

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WGF	Gas generator fuel flow rate	lb/sec	/CMAIN/, 16, 17, 18, 21, 31, 143, 146, 148, 151, 153
WGO	Gas generator oxidizer flow rate	lb/sec	/CMAIN/, 17, 19, 20, 51, 159, 162, 164, 167, 169
WG	Gas generator total flow rate	lb/sec	/CMAIN/, 15, 16, 17, 142
WMIS	Miscellaneous flow rate	lb/sec	/CMAIN/, 18, 19
WP	Pump mass flow rate	lb/sec	/CPUMP/, 18, 19
WRD	Resistance exit mass flow rate	lb/sec	/CRES/, 28, 30, 31, 34, 37, 45, 48, 50, 51, 54, 57, 65, 100, 101, 106, 107, 143, 146, 151, 153, 159, 162, 167, 169
WRI	Resistance inlet mass flow rate	lb/sec	/CRES/, 28, 30, 31, 34, 37, 45, 48, 50, 51, 54, 57, 65, 143, 146, 151, 153, 159, 162, 167, 169
WT	Turbine gas flow rate	lb/sec	/CTURB/, 131
WX	Turbine exhaust flow rate	lb/sec	/CTURBX/, 131
XISPC	Thrust chamber specific impulse	sec	/CMAIN/, 177
XISPE	Engine specific impulse	sec	/CMAIN/, 178

SUBROUTINE F1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
XMRCT	Storage array for independent argument, XMRTC, in Subroutine ESTM	—	25, 85
XMREN	Nominal engine mixture ratio	—	/H1F1J2/, 24, 25
XMRE	Engine mixture ratio	—	/CMAIN/, 22, 24, 25
XMRG	Gas generator mixture ratio	—	/CGPROP/, 16
XMRTC	Thrust chamber mixture ratio	—	/CCHAM/, 25, 85
XNPP	Pump speed	rpm	/CPUMP/, 76, 81, 85
XNP	Pump speed	rpm	/CPUMP/, 76, 81, 85, 116
XNT	Turbine speed	rpm	/CTURB/, 116

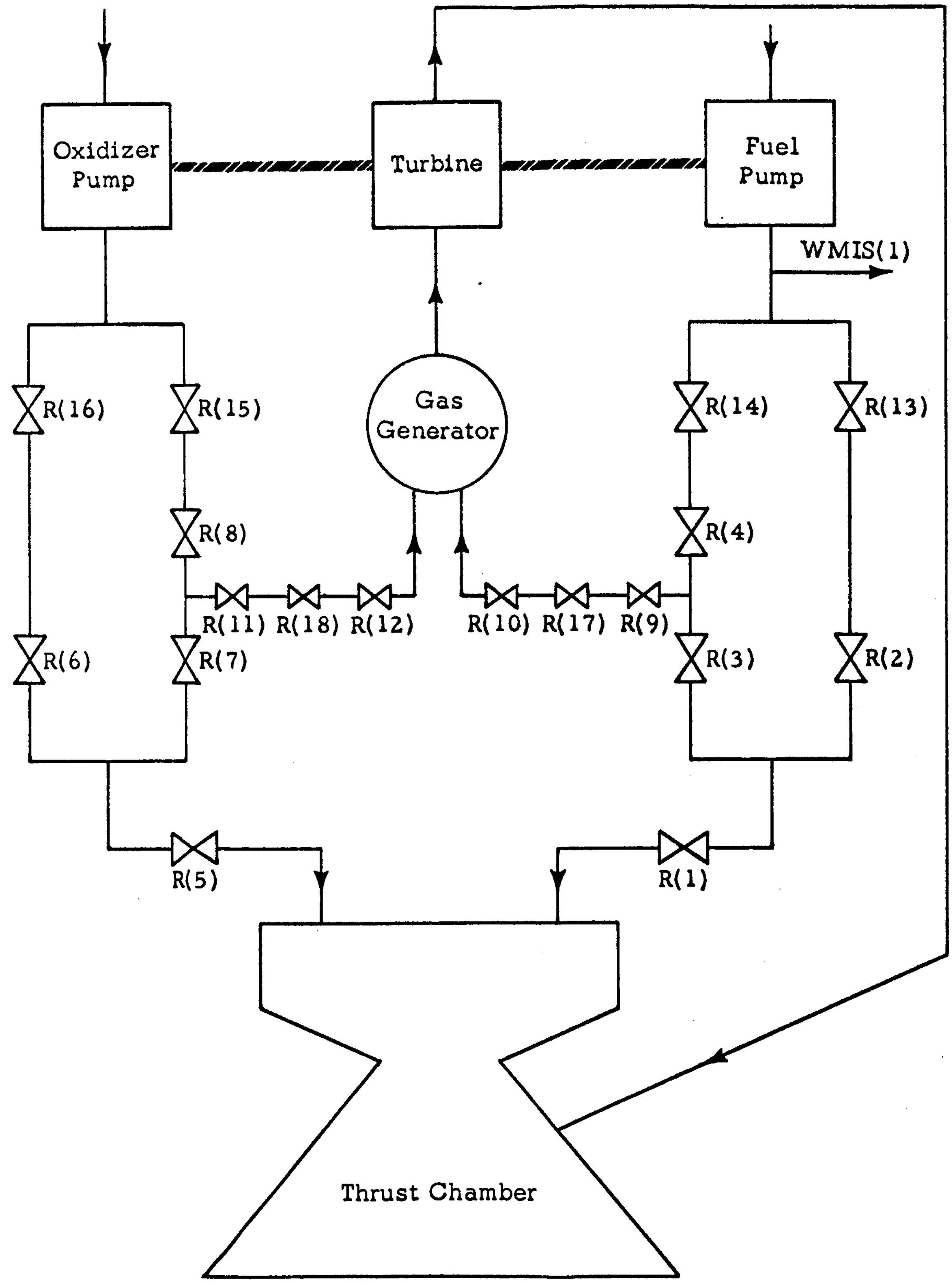


FIGURE 83. P-1 TYPE ENGINE SYSTEM SCHEMATIC

```
$IBFTC F1      REF,FULIST
      SUBROUTINE F1 (NEGGG)
      COMMON /H1F1J2/ RGG,DEL8,DEL9,XMREN
      COMMON /CPUMP/    PPI(2),TPIS(2),XNPSH(2),      PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
C
      COMMON /CGPROP/      C3(6),C4(6),C5(6),C6(6),CSG,GAMG,CPG,XMWG,
     1XMRG,RG,XOMG
C
      COMMON /CRES/ PRI(20),TRIS(20),PVRI(20),RHRI(20),VHRI(20),WRD(20),
     1DRI(20),DEL4(20),TRDS(20),DLTR(20),RHRD(20),RHRA(20),PRD(20),
     2R(20),VHRD(20),DRD(20),PRDS(20),PVRD(20),DEL5(20),WRI(20),PRIS(20)
C
      COMMON /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
C
      COMMON /CTURBX/ TXD,TXI,DELTX,PXDS,PXIS,RX,RGX,TXIS,WX,CSXD,XOMX,
     1GAMX,XMXD, PRXD,PXD,AXD,FX,TXDS,CPCX
C
      COMMON /CCHAM/  INDPR ,CF ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCO
C
      COMMON /CMAIN/WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),TN,
     1ORES(4),WG,WGO,WGF,PCIE,XMRE,PGG,XISPC,XISPE,FE,WEO,WEF,TPIR(2)
      COMMON /CDEN/TRSG(2),SG(2)
      COMMON /CBAL/KBAL /CPA/PA /CRESGG/ RESOG,RESFG
C
      DIMENSION RTHP(20),DRTHP(20),TPXT(20),DTPXT(20)
      DIMENSION XMRCT(20),DXMRCT(20),FET(20),DFET(20)
      DIMENSION PGGFT(20),DPGGFT(20),PGGOT(20),DPGGOT(20)
      DIMENSION WDOTT(20),DWDOTT(20)
      DATA LUP1/6H XMRE /,LUP2/6H HPT  /,LUP3/6H PTES /,LUP4/6H  FE   /
      DATA LUP5/6HXNP(1)/,LUP6/6H PGGF /,LUP7/6H PGGO /
      DATA LUP8/6HF FLOW/,LUP9/6HO FLOW/
C
      KPO=INDPR
      KPF=INDPR+1
      GO TO (4,3),KBAL
    3 CALL SWITCH(6,DRI(10),DRI10,DRD(10),DRD10,DRI(12),DRI12,DRD(12),
     1DRD12,R(17),RESFG,R(18),RESOG)
      GO TO 14
    4 CALL SWITCH(11,DRI10,DRI(10),DRD10,DRD(10),DRI12,DRI(12),DRD12,
     1DRD(12),PPDA(1),.0,PPDA(2),.0,ORES(1),.0,ORES(2),.0,ORES(3),.0,
     2ORES(4),.0,         T,TN)
   14 CALL SWITCH(10,IPBO,0,IPBR,0,IDLT,0,IDLFE,0,IDLMRC,0,IDLPX,0,
     1IDLPGF,0,IDLPGO,0,IGGLUP,0,IPB2,0)
```

FIGURE 84. SYMBOLIC LISTING OF SUBROUTINE F1

```
C           IN GPROP T IS GIVEN IN DEGREES FAHRENHEIT
      TPIS(1)=TPIR(1)-459.688
      TPIS(2)=TPIR(2)-459.688
   16 CALL GPROP(T)
      IF(IGGLUP)17,17,13
   17 FAUX=.0
      CALL CHAMBR
      PCIE=1.146*PNS
   13 WG=32.174*ATI(1)*PTI(1)/CSG
      WGF=WG/(XMRG+1.0)
      WGO=WG-WGF
      WP(1)=WCF+WGF+WMIS(1)
      WP(2)=WCO+WGO+WMIS(2)
      WEO=WCO+WGO
      WEF=WCF+WGF
      XMRE=WEO/WEF
      GO TO (18,21),KBAL
   18 IF(ABS(XMRE-XMREN)-DEL9)21,21,19
   19 CALL ESTM(IDLMRC,2,XMRCT,DXMRCT,XMRE-XMREN,XMRTC,LUP1)
      GO TO 17
   21 TRIS(1)=TPIS(1)+DLTP(1)
      CALL SWITCH(3,PRI(1),PCIE,WRI(1),WCF,WRD(1),WCF)
      CALL RES(1,1,-1,-1.0,KPF)
      CALL SWITCH(7,TRIS(2),TRDS(1),PRI(2),PRD(1),TRIS(3),TRDS(1),PRI(3)
     1,PRD(1),WRI(2),WRD(1)/2.,WRD(2),WRI(2),WRI(3),WRI(2))
      CALL SWITCH (3,WRD(3),WRI(3)+WGF,WDOT,WRI(2),IDLW,0)
  109 CALL RES (2,1,-1,-1.0,KPF)
      CALL RES (3,1,-1,-1.0,KPF)
      CALL SWITCH(4,TRIS(4),TRDS(3),PRI(4),PRD(3),WRI(4),WRD(3),WRD(4),
     1WRI(4))
      CALL RES (4,1,-1,-1.0,KPF)
      GO TO (112,22),KBAL
   22 CALL SWITCH(8,TRIS(13),TRDS(2),PRI(13),PRD(2),WRI(13),WRD(2),
     1WRD(13),WRI(13),TRIS(14),TRDS(4),PRI(14),PRD(4),WRI(14),WRD(4),
     2WRD(14),WRI(14))
      CALL RES (13,1,-1,-1.0,KPF)
      CALL RES (14,1,-1,-1.0,KPF)
      IF (ABS(PRD(13)-PRD(14))-DEL4(13)) 23,23 ,111
  111 CALL ESTM (IDLW,2,WDOTT,DWDOTT,PRD(13)-PRD(14),WDOT ,LUP8)
      GO TO 114
  112 IF (ABS(PRD(4)-PRD(2))-DEL4(2))23,23,113
  113 CALL ESTM (IDLW,2,WDOTT,DWDOTT,PRD(4)-PRD(2),WDOT ,LUP8)
  114 CALL SWITCH (4,WRI(2),WDOT,WRI(3),WCF-WDOT  ,WRD(2),WRI(2),WRD(3),
     1WEF-WDOT)
      GO TO 109
   23 TRIS(5)=TPIS(2)+DLTP(2)
      CALL SWITCH(3,PRI(5),PCIE,WRI(5),WCO,WRD(5),WRI(5))
      CALL RES (5,2,-1,-1.0,KPO)
      CALL SWITCH(7,TRIS(6),TRDS(5),PRI(6),PRD(5),WRI(6),WRD(5)/2.,
     1WRD(6),WRI(6),TRIS(7),TRDS(5),PRI(7),PRD(5),WRI(7),WRI(6))
```

FIGURE 84. SYMBOLIC LISTING OF SUBROUTINE F1 (cont.)

```
      CALL SWITCH (3,WRD(7),WRI(7)+WGO,WDOT,WRI(6),IDLW,0)
  115 CALL RES (6,2,-1,-1.0,KPO)
      CALL RES (7,2,-1,-1.0,KPO)
      CALL SWITCH(4,TRIS(8),TRDS(7),PRI(8),PRD(7),WRI(8),WRD(7),WRD(8),
     1WRI(8))
      CALL RES (8,2,-1,-1.0,KPO)
      GO TO (117, 24),KBAL
   24 CALL SWITCH(8,TRIS(15),TRDS(8),PRI(15),PRD(8),WRI(15),WRD(8),
     1WRD(15),WRI(15),TRIS(16),TRDS(6),PRI(16),PRD(6),WRI(16),WRD(6),
     2WRD(16),WRI(16))
      CALL RES (15,2,-1,-1.0,KPO)
      CALL RES (16,2,-1,-1.0,KPO)
      IF (ABS(PRD(15)-PRD(16))-DEL4(15)) 26, 26,116
  116 CALL ESTM(IDLW,2,WDOTT,DWDOTT,PRD(15)-PRD(16),WDOT ,LUP9)
      GO TO 119
  117 IF (ABS(PRD(6)-PRD(8))-DEL4(6))25 ,25 ,118
  118 CALL ESTM(IDLW,2,WDOTT,DWDOTT,PRD(6)-PRD(8),WDOT,LUP9)
  119 CALL SWITCH( 4,WRI(6),WDOT,WRI(7),WCO-WDOT,WRD(6),WRI(6),WRD(7),
     1WEO-WDOT)
      GO TO 115
   26 PPD(1)=PRD(13)
      PPD(2)=PRD(15)
      GO TO 30
   25 PPD(1)=PRD(2)
      PPD(2)=PRD(6)
      JC=1
   30 CALL PUMP(2,KPO)
      IF(IPBO-1)36,50,36
   36 IF(IPBR-1)38,40,38
   38 XNPP(1)=XNP(2)
   40 CALL PUMP(1,KPF)
   42 IF(IPBR-1)50,45,50
   45 GO TO (46,47),KBAL
   46 IPBO=1
      XNPP(2)=XNP(1)
      IPB2=IPB(2)
      IPB(2)=1
      GO TO 30
   47 CALL ESTM(IDLMRC,2,XMRCT,DXMRCT,XNPP(1 )-XNP(1),XMRTC,LUP5)
      IPBR=0
      GO TO 17
   50 GO TO (51,56),KBAL
   51 DO 60 J=1,2
      IF (H(J)-HPRIM(J))60,60,55
   55 PPDA(J)=PPI(J)+RHP(J)*H(J)/144.0
      GO TO (52,53),JC
   52 IF(ABS(PPDA(J)-PPD(J))-1.0)59,59,54
   54 PPD(J)=PPDA(J)
      HPRIME   =HPRIM(J)
      JC=2
```

FIGURE 84. SYMBOLIC LISTING OF SUBROUTINE F1 (cont.)

```
      GO TO (40,30),J
   53 HPRIM(J)=HPRIME
   59 GO TO (57,58),J
   57 ORES(1)= 72.0*(RHPD(1)+RHRD(2))*(PPDA(1)-PRD(2))/WRD(2)**2
      ORES(2)= 72.0*(RHPD(1)+RHRD(4))*(PPDA(1)-PRD(4))/WRD(4)**2
      R(13)=ORES(1)
      R(14)=ORES(2)
      PPD(1)=PPDA(1)
      GO TO 56
   58 ORES(3)= 72.0*(RHPD(2)+RHRD(6))*(PPDA(2)-PRD(6))/WRD(6)**2
      ORES(4)= 72.0*(RHPD(2)+RHRD(8))*(PPDA(2)-PRD(8))/WRD(8)**2
      R(15)=ORES(3)
      R(16)=ORES(4)
      PPD(2)=PPDA(2)
      GO TO 56
   60 CONTINUE
   56 CALL SWITCH (4,IPBR,0,IPBO,0,IDLMRC,0,IPB(2),IPB2)
      GO TO (62,310),KBAL
   62 TTI(1)=T + 459.688
      XNT(1)=GR(1)*XNP(1)
      CALL SWITCH (4,CPT(1),CPG,RGT(1),RG,GAMT(1),GAMG,CST(1),CSG)
   70 CALL TURBIN(1)
      HPTR=(BHPP(1)+BHPP(2))*ETAGR(1)
      IF(ABS(HPTR-HPT(1))-DEL6)210,210,200
  200 GO TO (171,173),KBAL
  171 PARAM=PTI(1)
      GO TO 174
  173 PARAM=PNS
  174 CALL ESTM(IDLT,2,RTHP,DRTHP,HPTR-HPT(1),PARAM,LUP2)
      GO TO (175,176),KBAL
  175 PTI(1)=PARAM
      GO TO 13
  176 PNS=PARAM
      GO TO 17
  210 CALL SWITCH(3,IDLT,0,RGX,RG,WX,WT(1))
      CALL TURBEX
      IF(ABS(PXIS-PTES(1))-DEL8)230,230,220
  220 CALL ESTM(IDLPX,2,TPXT,DTPXT,PXIS-PTES(1),PTES(1),LUP3)
      GO TO 70
  230 FE=FC
      IDLPX=0
      GO TO (240,318),KBAL
  240 IF(ABS(FE-FEN)-DEL7)310,310,300
  300 CALL ESTM(IDLFE,2,FET,DFET,FE-FEN,PNS,LUP4)
      GO TO 17
  310 PGG=PTI(1)+RG*TTI(1)*RGG*WG**2/PTI(1)/20736.0
  311 CALL SWITCH(4,TRIS(9),TRDS(3),PRI(9),PRDS(3),WRI(9),WGF,WRD(9),WGF
     1)
      CALL RES (9,1,-1,1.0,KPF)
      GO TO (312,313),KBAL
```

FIGURE 84. SYMBOLIC LISTING OF SUBROUTINE F1 (cont.)

```
312 CALL SWITCH(4,TRIS(10),TRDS(9),PRI(10),PGG,WRI(10),WGF,WRD(10),WGF
   1)
    CALL RES (10,1,-1,-1.0,KPF)
    R(17)=(PRD(9)-PRD(10))*(RHRD(9)+RHRD(10))*72./WGF**2
    RESFG=R(17)
    IF (RESFG)450,314,314
313 CALL SWITCH(4,TRIS(17),TRDS(9),PRI(17),PRD(9),WRI(17),WGF,WRD(17),
   1WGF)
    CALL RES (17,1,1,1.0,KPF)
    CALL SWITCH(6,TRIS(10),TRDS(17),PRI(10),PRD(17),WRI(10),WGF,
   1WRD(10),WGF,DRI(10),DRD10,DRD(10),DRI10)
    CALL RES (10,1,1,1.0,KPF)
    IF (ABS(PRD(10)-PGG)-DEL4(10))320,320,315
315 CALL ESTM(IDLPGF,2,PGGFT,DPGGFT,PRD(10)-PGG,PTI,LUP6)
    GO TO 13
320 IDLPGF=0
314 CALL SWITCH(4,TRIS(11),TRDS(7),PRI(11),PRDS(7),WRI(11),WGO,WRD(11)
   1,WGO)
    CALL RES (11,2,-1,1.0,KPO)
    GO TO (316,317),KBAL
316 CALL SWITCH(4,TRIS(12),TRDS(11),PRI(12),PGG,WRI(12),WGO,WRD(12),
   1WGO)
    CALL RES (12,2,1,-1.0,KPO)
    R(18)=(PRD(11)-PRD(12))*(RHRD(11)+RHRD(12))*72./WGO**2
    RESOG=R(18)
    IF (RESOG)450,318,318
317 CALL SWITCH(4,TRIS(18),TRDS(11),PRI(18),PRD(11),WRI(18),WGO,
   1WRD(18),WGO)
    CALL RES (18,2,1,1.0,KPO)
    CALL SWITCH(6,TRIS(12),TRDS(18),PRI(12),PRD(18),WRI(12),WGO,
   1WRD(12),WGO,DRI(12),DRD12,DRD(12),DRI12)
    CALL RES (12,2,1,1.0,KPO)
    IF (ABS(PRD(12)-PGG)-DEL4(12))321,321,322
322 CALL ESTM (IDLPGO,2,PGGOT,DPGGOT,PRD(12)-PGG,T,LUP7)
    IGGLUP=1
    GO TO 16
321 IDLPGO=0
    GO TO 62
318 XISPC=FC/WC
    XISPE=FE/(WEO+WEF)
    RETURN
450 NEGGG=1
    WRITE (6,451)
451 FORMAT(/////20X,47HA GAS GENERATOR ORIFICE RESISTANCE IS NEGATIVE.
   1/////)
998 RETURN
    END
```

FIGURE 84. SYMBOLIC LISTING OF SUBROUTINE F1 (cont.)

LXVI. SUBROUTINE RL10

This subroutine performs the preliminary calculations of the initial values required by the program and provides the necessary logic to control the subroutines which describe the operation of the various RL10-A1 engine components. The following input data is required:

a. Pump inlet conditions

b. Engine mixture ratio

c. Thrust chamber injector and pressure

d. Thrust chamber throat area

e. Balance/rebalance indicator

This subroutine uses COMMON blocks LXRL10, CCHAM, CBAL, CMAIN, MNRL10, CPUMP, CTURB, and PNSPRI.

CALLING SEQUENCE

The subroutine calling sequence is:

```
CALL RL10
```

SOLUTION METHOD

Reset iteration loop counters for Subroutine ESTM.

1. IDLC = 0

2. IDLE = 0

3. IDLF = 0

LOX pump inlet density:

4. CALL UVAR (TPIS(2), 2, RHPI(2))

Initialize thrust chamber mixture ratio:

5. XMRTCL = XMRTC

Determine if balance or rebalance:

6. If KBAL - 1 $\leq$ 0, go to 11

   If KBAL - 1 $>$ 0, go to 7

Engine mixture ratio:

7. XMRTC = 5 + (.017)(PPI(2) - 77.9)

   - (.01)(PPI(1) - 35.59)

   - (.026)(TPIS(2) - 176)

   + (.095)(TPIS(1) - 38.7)

   - (.017)(PCIEL - 302.6361)

8. If XNP(1) $\leq$ 0, go to 9

   If XNP(1) $>$ 0, go to 14

Fuel pump speed:

9. $XNP(1) = 27319 - (15)(PPI(2) - 77.9) - (6)(PPI(1) - 35.49)$
$+ (23)(TPIS(2) - 176) + (80)(TPIS(1) - 38.7)$
$+ (59)(PCIEL - 302.6361)$

10. Go to 14

Ground mixture ratio control valve flow loss coefficient:

11. $CON3 = 4.8509 - (4.75)(XMRTCL - 5)$
$- (.0725)(PCIEL - 302.6361)$

12. If $XNP(1) \leq 0$, go to 13
If $XNP(1) > 0$, go to 14

Fuel pump speed:

13. $XNP(1) = 27319 + (43)(PCIEL - 302.6361)$
$- (800)(XMRTC - 5)$

14. If $Q(2) \leq 0$, go to 15
If $Q(2) > 0$, go to 22

The following equations calculate an initial LOX volume flow rate.

Characteristic velocity efficiency:

15. $ETCS = (1.03694 - (.019841)(XMRTC))(1.01) - .0136$

Thrust chamber nozzle stagnation pressure:

16. PNS = (.97)(PCIEL)

Theoretical characteristic velocity:

17. CALL BIVAR (XMRTC, PNS, 33, CSTHE)

Actual characteristic velocity:

18. CS = (ETCS)(CSTHE)

Chamber total mass flow rate:

19. WC = (PNS)(ATC)(32.174)/(CS)

Oxidizer mass flow rate:

20. WCO = (WC)(XMRTC)/(1 + XMRTC)

Oxidizer pump inlet volume flow rate:

21. Q(2) = (WCO)/(RHPI(2))

Set indicator for correct entry point into Subroutine LOX:

22. IND = 0

Oxidizer system performance:

23. CALL LOX (IND)

Fuel system performance:

```
24.  CALL FUEL (IND, IDLC)
```

Check indicator to see whether Subroutine LOX is to be re-entered:

```
25.  If IND = 0, go to 26
     If IND $\neq$ 0, go to 23
```

Turbine performance:

```
26.  CALL TRBINE
```

Chamber performance:

```
27.  CALL CHMBER (IND, IDLE, IDLF)
```

Check indicator to see whether Subroutine LOX is to be re-entered:

```
28.  If IND < 2, go to 29
     If IND = 2, go to 23
     If IND > 2, go to 22
```

Thrust chamber nozzle stagnation pressure for use in Subroutine INFL:

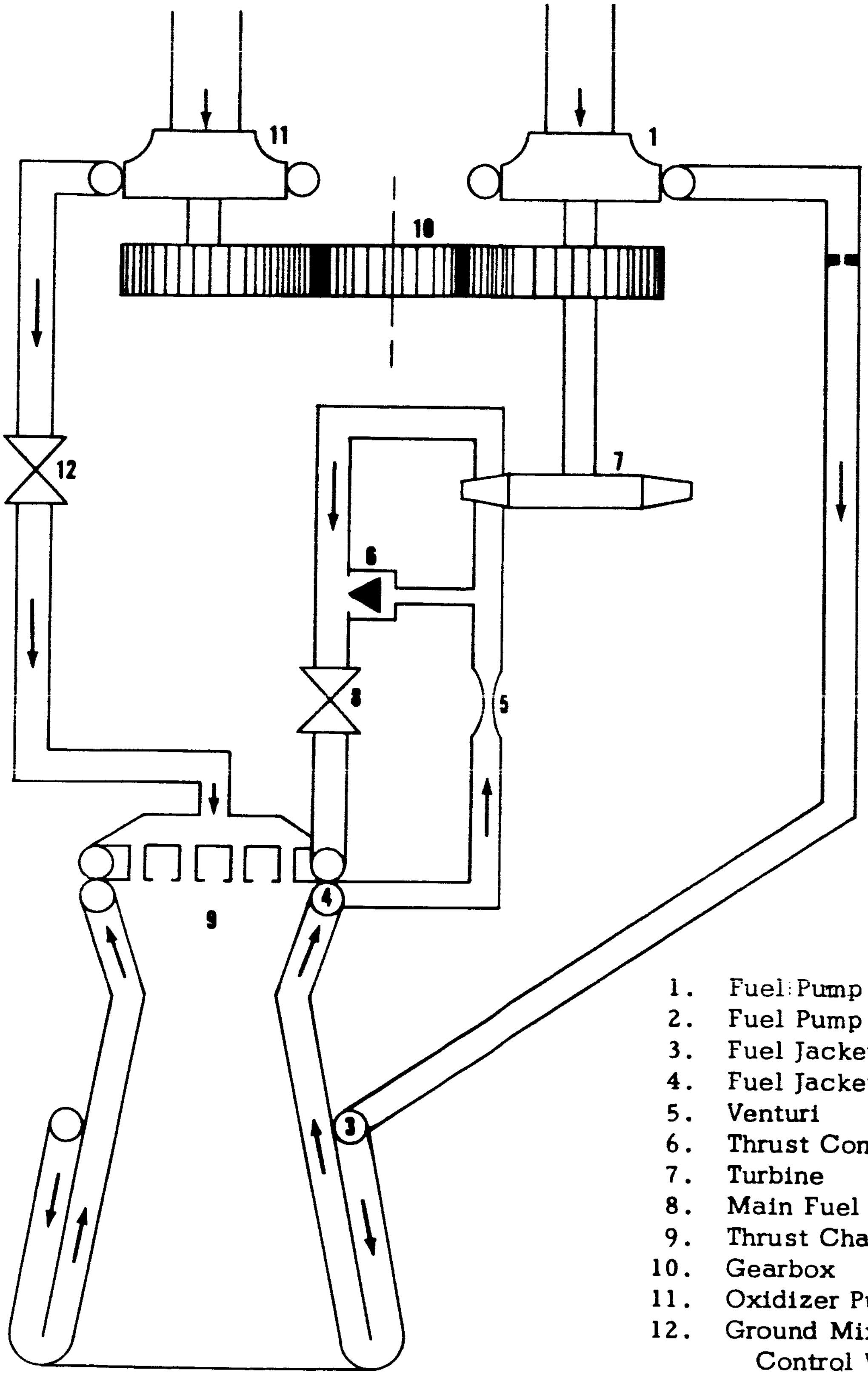
```
29.  PNSPR = (.97)(PCIEL)
     Return
```

SUBROUTINE RL10 NOMENCLATURE

Symbol	Description	Units	Reference
CON3	Ground mixture ratio control valve flow loss coefficient	$\frac{lbf\text{-}sec^2}{lbm\text{-}ft^3\text{-}in^2}$	/LXRL10/, 11
CS	Thrust chamber characteristic velocity	ft/sec	/CCHAM/, 18, 19
CSTHE	Thrust chamber theoretical characteristic velocity	ft/sec	/CTURB/, 17, 18
ETCS	Thrust chamber combustion efficiency	—	/CCHAM/, 15, 18
IDLC	Iteration loop control counter	—	1, 24
IDLE	Iteration loop control counter	—	2, 27
IDLF	Iteration loop control counter	—	3, 27
IND	Indicator to determine correct entry point into Subroutine LOX	—	22, 23, 24, 25, 27, 28
KBAL	Balance/rebalance indicator	—	/CBAL/, 6
PCIEL	Loaded value of thrust chamber injector end pressure	$\frac{lbf}{in^2}$	/MNRL10/, 7, 9, 11, 13, 16, 29

SUBROUTINE RL10 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PNS	Thrust chamber nozzle stagnation pressure	$\frac{lbf}{in^2}$	/CCHAM/, 16, 17, 19
PNSPR	Value of thrust chamber nozzle stagnation pressure for use in Subroutine INFL	$\frac{lbf}{in^2}$	/PNSPRI/, 29
PPI	Pump inlet total pressure	$\frac{lbf}{in^2}$	/CPUMP/, 7, 9
Q	Pump inlet volume flow rate	$\frac{ft^3}{sec}$	/MNRL10/, 14, 21
RHPI	Pump inlet density	$\frac{lbm}{ft^3}$	/CPUMP/, 4, 21
TPIS	Pump inlet static temperature	^oR	/CPUMP/, 4, 7, 9
WC	Thrust chamber total flow rate	$\frac{lbm}{sec}$	/CCHAM/, 19, 20
WCO	Thrust chamber oxidizer flow rate	lbm/sec	/CCHAM/, 20, 21
XMRTC	Thrust chamber mixture ratio	—	/CCHAM/, 5, 7, 11, 13, 15, 17, 20
XMRTCL	Loaded value of thrust chamber mixture ratio	—	/MNRL10/, 5
XNP	Pump speed	$\frac{rev}{min}$	/CPUMP/, 8, 9, 12, 13



1. Fuel Pump
2. Fuel Pump Discharge Orifice
3. Fuel Jacket Inlet Manifold
4. Fuel Jacket Exit Manifold
5. Venturi
6. Thrust Controller
7. Turbine
8. Main Fuel Shutoff Valve
9. Thrust Chamber
10. Gearbox
11. Oxidizer Pump
12. Ground Mixture Ratio
    Control Valve

FIGURE 85. RL10-A1 TYPE ENGINE SYSTEM SCHEMATIC

```
$IBFTC RL10S    FULIST,REF
      SUBRØUTINE RL10
      CØMMØN /PNSPRI/ PNSPR
      CØMMØN /MNRL10/ PCIEL,Q(2),XMRTCL,PII(2)
      CØMMØN /LXRL10/ TØL3,TØL5,CD2,CØN1,CØN2,CØN3,CØN4,CØN5
      CØMMØN /FLRL10/ CØN8,CØN6,TØL4,FLØPT,CØN7,TØJE,WCF1,TØL6,TØL7,
     1CØN10,AVT
      CØMMØN /CHRL10/  TØL8,TØL9,FCL
      CØMMØN /PPRL10/ XRATIØ(2), TØL1, TØL2, XNPN(2)
      CØMMØN /TBRL10/ CØN9,ETGER,CØN11,CØN12,CØN13,CØN14,CØN15,TØL10
     1,HP1,HP2
      CØMMØN /CPUMP/     PPI(2),TPIS(2),XNPSH(2),     PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
      CØMMØN /CTURB/    PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
      CØMMØN /CCHAM/  INDPR ,CF ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCØ
      CØMMØN /CBAL/KBAL /CPA/PA /CRESGG/RESØG,RESFG
      CØMMØN /CMAIN/WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),TN,
     1ØRES(4),WG,WGØ,WGF,PCIE,XMRE,PGG,XISPC,XISPE,FE,WEØ,WEF,TPIR(2)
 999  IDLC = 0
      IDLE = 0
      IDLF = 0
C RL10 PRØGRAM -- FIRST GUESS CALCULATIØNS
      CALL UVAR(TPIS(2),2,RHPI(2) )
      XMRTCL  = XMRTC
      IF(KBAL-1)102,102,101
  101 XMRTC = 5.+.017*(PPI(2)-77.9)-.01*(PPI(1)-35.49)-
     1.026*(TPIS(2)-176.)+.095*(TPIS(1)-38.7)-.017*(PCIEL-302.6361)
      IF(XNP(1))103,103,104
  103 XNP(1)= 27319.-15.*(PPI(2)-77.9) - 6.*(PPI(1)-35.49) +
     123.*(TPIS(2)-176.) + 80.*(TPIS(1)-38.7) + 59.*(PCIEL-302.6361)
      GØ TØ 104
  102 CØN3 = 4.8509 - 4.75*(XMRTC-5.) - .0725*(PCIEL-302.6361)
      IF(XNP(1))105,105,104
  105 XNP(1) = 27319. + 43.*(PCIEL-302.6361) - 800.*(XMRTC-5.)
  104 IF(Q(2))106,106,108
CTHE FØLLØWING EQUATIØNS CALCULATE AN INITIAL LØX VØLUME FLØWRATE
  106 ETCS = (1.03694 - .019841*XMRTC)*1.01 - .0136
      PNS  = .97 * PCIEL
      CALL BIVAR(XMRTC,PNS,33,CSTHE)
      CS   = ETCS * CSTHE
      WC   = (PNS*ATC*32.174)/CS
      WCØ  = (WC * XMRTC)/(1. + XMRTC)
      Q(2) = WCØ / RHPI(2)
  108 IND = 0
```

FIGURE 86. SYMBOLIC LISTING OF SUBROUTINE RL10

```
110 CALL LØX(IND)
    CALL FUEL(IND,IDLC)
    IF(IND)110,111,110
111 CALL TRBINE
115 CALL CHMBER(IND,IDLE,IDLF)
    IF(IND-2)120,110,108
120 CØNTINUE
    PNSPR = .97 * PCIEL
    RETURN
    END
```

FIGURE 86. SYMBOLIC LISTING OF SUBROUTINE RL10 (cont.)

LXVII. SUBROUTINE CHAMBR

This subroutine calculates the operating performance of a thrust chamber, based upon a given chamber pressure and mixture ratio. A thrust balance loop is included to assure that a reasonable chamber pressure is used in the case of an engine balance.

CALLING SEQUENCE

The calling sequence is:

CALL CHAMBR

This subroutine uses COMMON blocks CCHAM, CBAL and CPA.

SOLUTION METHOD

1. IDLTC = 0

Determine theoretical thrust coefficient and characteristic velocity:

2. CALL TCPROP (INDPR, XMRTC, PNS, EPS, CSTHE, CFTHE)

Compute actual characteristic velocity:

3. CS = (CSTHE)(ETCS)

Compute chamber flow rates:

4. WC = (32.174)(PNS)(ATC)/(CS)

5. WCF = (WC)/(1.0 + XMRTC)

6. WCO = WC - WCF

Compute actual thrust coefficient:

7. CF = (CFTHE)(ETCF) - (EPS)(PA)/(PNS)

Compute chamber thrust:

8. FC = (CF)(PNS)(ATC)

Test for balance or rebalance:

9. If KBAL = 1, go to 10

If KBAL = 2, go to 14

In balance only, iterate on chamber pressure to obtain a thrust balance:

10. FEPR = (FC)(FAUX)

11. If $|$ FEPR - FEN $| \leq$ DEL7, go to 14

If $|$ FEPR - FEN $| >$ DEL7, go to 12

12. CALL ESTM (IDLTC, 2, TTC, DTTC, (FEPR - FEN), PNS, LUP1)

13. Go to 2

14. Return

SUBROUTINE CHAMBR NOMENCLATURE

Symbol	Description	Units	Reference
ATC	Throat area	in^2	/CCHAM/, 4, 8
CF	Thrust coefficient	—	/CCHAM/, 7, 8
CFTHE	Theoretical thrust coefficient	—	/CCHAM/, 2, 7
CS	Characteristic velocity	ft/sec	/CCHAM/, 3, 4
CSTHE	Theoretical characteristic velocity	ft/sec	/CCHAM/, 2, 3
DEL7	Engine thrust tolerance	lb	/CCHAM/, 11
DTTC	Storage array for dependent argument in Subroutine ESTM	—	DIM, 12
EPS	Expansion area ratio	—	/CCHAM/, 2, 7
ETCF	Nozzle efficiency	—	/CCHAM/, 7
ETCS	Combustion efficiency	—	/CCHAM/, 3
FAUX	Auxiliary engine thrust	lb	/CCHAM/, 10
FC	Chamber thrust	lb	/CCHAM/, 8, 10
FEN	Nominal engine thrust	lb	/CCHAM/, 11, 12
FEPR	Calculated engine thrust	lb	10, 11, 12
IDLTC	Counter used in Subroutine ESTM	—	1, 12
INDPR	Propellant combination indicator	—	/CCHAM/, 2

SUBROUTINE CHAMBR NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
KBAL	Engine balance/re-balance indicator	—	/CBAL/, 9
LUP1	—	—	DATA, 12
PA	Ambient pressure	lb/in^2	/CPA/, 7
PNS	Nozzle stagnation pressure	lb/in^2	/CCHAM/, 2, 4, 7, 8, 12
TTC	Storage array for independent argument in Subroutine ESTM	—	DIM, 12
WC	Chamber total flow rate	lb/sec	/CCHAM/, 4, 5, 6
WCF	Chamber fuel flow rate	lb/sec	/CCHAM/, 5, 6
WCO	Chamber oxidizer flow rate	lb/sec	/CCHAM/, 6
XMRTC	Chamber mixture ratio	—	/CCHAM/, 2, 5

```
$IBFTC CHAMBR  REF,LIST
      SUBROUTINE CHAMBR
      COMMON /CCHAM/  INDPR ,CF  ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCO
      COMMON /CBAL/KBAL /CPA/PA
      DIMENSION TTC(20),DTTC(20)
      DATA LUP1/6H  FE   /
      IDLTC=0
  105 CALL TCPROP(INDPR,XMRTC,PNS,EPS,CSTHE,CFTHE)
      CS=CSTHE*ETCS
      WC=32.174*PNS*ATC/CS
      WCF=WC/(1.0+XMRTC)
      WCO=WC-WCF
      CF =CFTHE*ETCF-EPS*PA/PNS
      FC=CF*PNS*ATC
      GO TO (106,120),KBAL
  106 FEPR=FC+FAUX
      IF(ABS(FEPR-FEN)-DEL7)120,120,107
  107 CALL ESTM(IDLTC,2,TTC,DTTC,FEPR-FEN,PNS,LUP1)
      GO TO 105
  120 RETURN
      END
```

FIGURE 87. SYMBOLIC LISTING OF SUBROUTINE CHAMBR

LXVIII. SUBROUTINE PUMP

This subroutine computes the operating performance of a pump assuming that the following information are available:

a. Inlet and discharge total pressures

b. Flow rate

c. Inlet propellant temperature

d. Propellant temperature rise across pump

e. Pump performance characteristics

f. Propellant characteristics

Since most liquid propellant rocket engine systems contain two propellant pumps, storage is reserved for two sets of pump performance data.

This subroutine uses COMMON blocks CPUMP, CDEN and CBAL.

CALLING SEQUENCE

The subroutine calling sequence is:

```
CALL PUMP (JP, KP)
```

SOLUTION METHOD

Initialize control variables:

1. IP = JP

2. IIDP = 0

Approximate pump inlet static pressure:

3. PPIS(IP) = PPI(IP)

Initialize iteration loop control counter:

4. IDLP = 0

Compute pump inlet propellant density:

5. CALL PRPDEN (KP, PPIS(IP), TPIS(IP), TRSG(IP), SG(IP), PVPI(IP), RHPI(IP))

Compute pump inlet velocity head:

6. $\mathrm{VHPI(IP)} = (3.627845)(\mathrm{WP(IP)}^2)/((\mathrm{RHPI(IP)})(\mathrm{DPI(IP)}^4))$

Compute pump inlet static pressure:

7. PPISP = PPI(IP) - VHPI(IP)

Compare assumed and calculated values of pump inlet static pressures. If convergence criterion is not satisfied, iterate on PPIS:

8. If $|\mathrm{PPIS(IP)} - \mathrm{PPISP}| \leq \mathrm{DEL1(IP)}$, go to 11

   If $|\mathrm{PPIS(IP)} - \mathrm{PPISP}| > \mathrm{DEL1(IP)}$, go to 9

```
9.  CALL ESTM (IDLP, 2, PUMT, DPUMT, PPIS(IP) - PPISP,
        PPIS(IP), LUP1)

10. Go to 5
```

Compute pump discharge propellant temperature:

```
11. TPDS(IP) = TPIS(IP) + DLTP(IP)
```

Approximate pump discharge static pressure:

```
12. PPDS(IP) = PPD(IP)
```

Initialize iteration loop control counter:

```
13. IDLP = 0
```

Compute pump discharge propellant density:

```
14. CALL PRPDEN (KP, PPDS(IP), TPDS(IP), TRSG(IP),
        SG(IP), PVPD(IP), RHPD(IP))
```

Compute pump discharge velocity head:

```
15. VHPD(IP) = (3.627845)(WP(IP)^2)/((RHPD(IP))(DPD(IP)^4))
```

Compute pump discharge static pressure:

```
16. PPDSP = PPD(IP) - VHPD(IP)
```

Compare assumed and calculated values of pump discharge static pressures. If convergence criterion is not satisfied, iterate on PPDS:

17. If $|\text{PPDS(IP)} - \text{PPDSP}| \leq$ DEL2(IP), go to 20

If $|\text{PPDS(IP)} - \text{PPDSP}| >$ DEL2(IP), go to 18

18. CALL ESTM (IDLP, 2, PUMT, DPUMT, PPDS(IP) - PPDSP, PPDS(IP), LUP2)

19. Go to 14

Compute average propellant density:

20. $\text{RHP(IP)} = (\text{RHPI(IP)} + \text{RHPD(IP)})/(2.0)$

Compute volumetric flow rate:

21. $\text{QP(IP)} = (448.8311)(\text{WP(IP)})/(\text{RHP(IP)})$

Compute pump head:

22. $\text{H(IP)} = (144.0)(\text{PPD(IP)} - \text{PPI(IP)})/(\text{RHP(IP)})$

23. $\text{HPRIM(IP)} = \text{H(IP)}$

Initialize iteration loop control counter:

24. $\text{IDLP} = 0$

Compute pump speed:

25. $\text{XNP(IP)} = \left|(\text{C1(1, IP)})(\text{QP(IP)}) + ((\text{C1(2, IP)})(\text{QP(IP)}^2) + (\text{C1(3, IP)})(\text{H(IP)}))^{.5}\right|$

Test to determine if speed balance should be accomplished:

26. If $IPB(IP) = 0$, go to 38

If $IPB(IP) \neq 0$, go to 27

Test for balance or rebalance:

27. If $KBAL = 1$, go to 28

If $KBAL = 2$, go to 36

Compare desired speed with computed speed:

28. If $XNPP(IP) < XNP(IP)$, go to 29

If $XNPP(IP) = XNP(IP)$, go to 38

If $XNPP(IP) > XNP(IP)$, go to 30

Test magnitude of IIDP. If IIDP has been set to 1, the program is iterating on pump head in an effort to match computed pump speed with desired pump speed.

29. If $IIDP = 1$, go to 31

If $IIDP \neq 1$, go to 34

For the case where the desired speed is greater than the computed speed, iterate on pump head until the convergence criterion is satisfied:

30. If $|XNPP(IP) - XNP(IP)| \leq DEL3(IP)$, go to 38

If $|XNPP(IP) - XNP(IP)| > DEL3(IP)$, go to 31

```
31.  CALL ESTM (IDLP, 2, PUMT, DPUMT, XNPP(IP) - XNP(IP),
          H(IP), LUP3)

32.  IIDP = 1

33.  Go to 25
```

Set indicator to denote that pump speed balance cannot be accomplished:

```
34.  IPBR = 1

35.  Go to 38
```

Compare desired speed with computed speed:

```
36.  If $| XNPP(IP) - XNP(IP) | \leq DEL3(IP)$, go to 38
     If $| XNPP(IP) - XNP(IP) | > DEL3(IP)$, go to 37
```

Set indicator to denote that pump speed balance has not been accomplished:

```
37.  IPBR = 1
     Return
```

Compute pump efficiency:

```
38.  $ETAP(IP) = (C1(4, IP))((QP(IP))/(XNP(IP)))^2$
          $+ (C1(5, IP))((QP(IP))/(XNP(IP))) + C1(6, IP)$
```

Compute net positive suction head:

39. $XNPSH(IP) = (144.0)(PPI(IP) - PVPI(IP))/(RHPI(IP))$

Compute specific speed:

40. $XNS(IP) = (XNP(IP))(QP(IP)^{.5})/(H(IP)^{.75})$

Compute suction specific speed:

41. $XNSS(IP) = (XNP(IP))(QP(IP)^{.5})/(XNPSH(IP)^{.75})$

Compute horsepower:

42. $BHPP(IP) = (WP(IP))(H(IP))/((550.0)(ETAP(IP)))$

Return

SUBROUTINE PUMP NOMENCLATURE

Symbol	Description	Units	Reference
BHPP	Pump brake horsepower	bhp	/CPUMP/, 42
C1	Array of coefficients for pump performance curves	—	/CPUMP/, 25, 38
DEL1	Pump inlet static pressure tolerance	psi	/CPUMP/, 8
DEL2	Pump discharge static pressure tolerance	psi	/CPUMP/, 17
DEL3	Pump speed tolerance	rpm	/CPUMP/, 30, 36
DLTP	Pump propellant temperature rise	°F	/CPUMP/, 11
DPD	Pump discharge flange diameter	in	/CPUMP/, 15
DPI	Pump inlet flange diameter	in	/CPUMP/, 6
DPUMT	Storage array for dependent argument in Subroutine ESTM	—	DIM, 9, 18, 31
ETAP	Pump efficiency	—	/CPUMP/, 38, 42
H	Pump head	ft	/CPUMP/, 22, 23, 25, 31, 40, 42
HPRIM	Pump head without main line orificing	ft	/CPUMP/, 23
IDLP	Iteration loop counter	—	4, 9, 13, 18, 24, 31
IIDP	Indicator used in pump speed balance loop	—	2, 29, 32

SUBROUTINE PUMP NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
IP	JP	—	1, 3, 5, 6, 7, 8, 9, 11, 12, 14, 15, 16, 17, 18, 20, 21, 22, 23, 25, 26, 28, 30, 31, 36, 38, 39, 40, 41, 42
IPB	Pump speed balance indicator 0 = By-pass pump speed balance loop ± = Enter pump speed balance loop	—	/CPUMP/, 26
IPBR	Indicator used in pump speed balance loop	—	/CPUMP/, 34, 37
JP	Index designating storage location of pump performance data. Must be set equal to 1 or 2	—	CALL, 1
KBAL	Engine balance/rebalance indicator	—	/CBAL/, 27
KP	Propellant density indicator	—	CALL, 5, 14
LUP1	—	—	DATA, 9
LUP2	—	—	DATA, 18
LUP3	—	—	DATA, 31
PPD	Pump discharge total pressure	psia	/CPUMP/, 12, 16, 22
PPDS	Pump discharge static pressure	psia	/CPUMP/, 12, 14, 17, 18

SUBROUTINE PUMP NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PPDSP	Pump discharge static pressure	psia	16, 17, 18
PPI	Pump inlet total pressure	psia	/CPUMP/, 3, 7, 22, 39
PPIS	Pump inlet static pressure	psia	/CPUMP/, 3, 5, 8, 9
PPISP	Pump inlet static pressure	psia	7, 8, 9
PUMT	Storage array for independent argument in Subroutine ESTM	—	DIM, 9, 18, 31
PVPD	Pump discharge propellant vapor pressure	psia	/CPUMP/, 14
PVPI	Pump inlet propellant vapor pressure	psia	/CPUMP/, 5, 39
QP	Pump volumetric flow rate	gpm	/CPUMP/, 21, 25, 38, 40, 41
RHP	Pump average propellant density	lb/ft^3	/CPUMP/, 20, 21, 22
RHPD	Pump discharge propellant density	lb/ft^3	/CPUMP/, 14, 15, 20
RHPI	Pump inlet propellant density	lb/ft^3	/CPUMP/, 5, 6, 20, 39
SG	Propellant specific gravity at reference temperature	—	/CDEN/, 5, 14
TPDS	Pump discharge propellant temperature	^oF	/CPUMP/, 11, 14

SUBROUTINE PUMP NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TPIS	Pump inlet propellant temperature	^{o}F	/CPUMP/, 5, 11
TRSG	Propellant reference temperature	^{o}F	/CDEN/, 5, 14
VHPD	Pump discharge velocity head	psia	/CPUMP/, 15, 16
VHPI	Pump inlet velocity head	psia	/CPUMP/, 6, 7
WP	Pump mass flow rate	lb/sec	/CPUMP/, 6, 15, 21, 42
XNP	Calculated pump speed	rpm	/CPUMP/, 25, 28, 30, 31, 36, 38, 40, 41
XNPP	Desired pump speed	rpm	/CPUMP/, 28, 30, 31, 36
XNPSH	Pump net positive suction head	ft	/CPUMP/, 39, 41
XNS	Pump specific speed	—	/CPUMP/, 40
XNSS	Pump suction specific speed	—	/CPUMP/, 41

```
$IBFTC PUMP    FULIST,REF,M94,DECK
      SUBRØUTINE PUMP (JP,KP)
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),     PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
      CØMMØN /CDEN/TRSG(2),SG(2) /CBAL/KBAL
      DIMENSIØN PUMT(20),DPUMT(20)
      DATA LUP1/6HPPIS  /,LUP2/6HPPDS  /,LUP3/6HXNP   /
      IP=JP
      IIDP=0
      PPIS(IP)=PPI(IP)
      IDLP=0
    2 CALL PRPDEN (KP,PPIS(IP),TPIS(IP),TRSG(IP),SG(IP),PVPI(IP),RHPI(IP
     1))
      VHPI(IP)=3.627845*WP(IP)**2/RHPI(IP)/DPI(IP)**4
      PPISP=PPI(IP)-VHPI(IP)
      IF(ABS(PPIS(IP)-PPISP)-DEL1(IP))9,9,8
    8 CALL ESTM(IDLP,2,PUMT,DPUMT,PPIS(IP)-PPISP,PPIS(IP),LUP1)
      GØ TØ 2
    9 TPDS(IP)=TPIS(IP)+DLTP(IP)
      PPDS(IP)=PPD(IP)
      IDLP=0
   11 CALL PRPDEN (KP,PPDS(IP),TPDS(IP),TRSG(IP),SG(IP),PVPD(IP),RHPD(IP
     1))
      VHPD(IP)=3.627845*WP(IP)**2/RHPD(IP)/DPD(IP)**4
      PPDSP=PPD(IP)-VHPD(IP)
      IF(ABS(PPDS(IP)-PPDSP)-DEL2(IP))18,18,17
   17 CALL ESTM(IDLP,2,PUMT,DPUMT,PPDS(IP)-PPDSP,PPDS(IP),LUP2)
      GØ TØ 11
   18 RHP(IP)=(RHPI(IP)+RHPD(IP))/2.0
      QP(IP)=448.8311*WP(IP)/RHP(IP)
      H(IP)=144.0*(PPD(IP)-PPI(IP))/RHP(IP)
      HPRIM(IP)=H(IP)
      IDLP=0
   20 XNP(IP)=ABS(C1(1,IP)*QP(IP)+(C1(2,IP)*QP(IP)**2+C1(3,IP)*H(IP))**.
     A5 )
   29 IF(IPB(IP))21,28,21
   21 GØ TØ(23,32),KBAL
   23 IF(XNPP(IP)-XNP(IP))22,28,24
   22 IF(IIDP-1)25,27,25
   24 IF(ABS(XNPP(IP)-XNP(IP))-DEL3(IP))28,28,27
   27 CALL ESTM(IDLP,2,PUMT,DPUMT,XNPP(IP)-XNP(IP),H(IP),LUP3)
      IIDP=1
      GØ TØ 20
   25 IPBR=1
      GØ TØ 28
   32 IF(ABS(XNPP(IP)-XNP(IP))-DEL3(IP))28,28,33
   33 IPBR=1
```

FIGURE 88. SYMBOLIC LISTING OF SUBROUTINE PUMP

```
      RETURN
   28 ETAP(IP)=C1(4,IP)*(QP(IP)/XNP(IP))**2+C1(5,IP)*(QP(IP)/XNP(IP))+
     1C1(6,IP)
      XNPSH(IP)=144.0*(PPI(IP)-PVPI(IP))/RHPI(IP)
      XNS(IP)=XNP(IP)*QP(IP)**.5/H(IP)**.75
      XNSS(IP)=XNP(IP)*QP(IP)**.5/XNPSH(IP)**.75
      BHPP(IP)=WP(IP)*H(IP)/(550.0*ETAP(IP))
      RETURN
      END
```

FIGURE 88. SYMBOLIC LISTING OF SUBROUTINE PUMP (cont.)

LXIX. SUBROUTINE TURBIN

This subroutine calculates the operating performance of a turbine. Storage has been reserved to allow for two complete sets of turbine performance data.

CALLING SEQUENCE

CALL TURBIN(IT)

Subroutine TURBIN uses COMMON block CTURB.

SOLUTION METHOD

Compute total to static pressure ratio:

1. $\mathrm{PRT(IT)} = \mathrm{(PTI(IT))/(PTES(IT))}$

Compute weight flow rate:

2. $\mathrm{WT(IT)} = \mathrm{(32.174)(ATI(IT))(PTI(IT))/(CST(IT))}$

Compute gas enthalpy change across turbine:

3. $\mathrm{DELHT(IT)} = \mathrm{(CPT(IT))(TTI(IT))}$

$$\left(1 - ((1)/(\mathrm{PRT(IT)}))^{((\mathrm{GAMT(IT)} - 1)/(\mathrm{GAMT(IT)}))}\right)$$

Compute velocity ratio:

4. $$\begin{aligned}\mathrm{UVRAT(IT)} = {} & (2.7576)(10^{-5})(\mathrm{DW(IT)})(\mathrm{XNT(IT)})/ \\ & ((\mathrm{DELHT(IT)})/(\mathrm{XNTST(IT)}))^{.5}\end{aligned}$$

Compute efficiency:

5. $$\begin{aligned}\mathrm{ETAT(IT)} = {} & (\mathrm{C7(1,\ IT)} + (\mathrm{C7(2,\ IT)})(\mathrm{PRT(IT)}) \\ & + (\mathrm{C7(3,\ IT)})(\mathrm{PRT(IT)})^{2})(\mathrm{UVRAT(IT)}) \\ & + (\mathrm{C7(4,\ IT)} + (\mathrm{C7(5,\ IT)})(\mathrm{PRT(IT)}) \\ & + (\mathrm{C7(6,\ IT)})(\mathrm{PRT(IT)})^{2})(\mathrm{UVRAT(IT)})^{2} \\ & + \ \mathrm{C7(7,\ IT)} + (\mathrm{C7(8,\ IT)})(\mathrm{PRT(IT)}) \\ & + (\mathrm{C7(9,\ IT)})(\mathrm{PRT(IT)})^{2}\end{aligned}$$

Compute brake horsepower:

6. $$\mathrm{HPT(IT)} = (1.414836)(\mathrm{WT(IT)})(\mathrm{ETAT(IT)})(\mathrm{DELHT(IT)})$$

Compute exit static pressure:

7. $$\begin{aligned}\mathrm{TTDS(IT)} = {} & (\mathrm{TTI(IT)})(1 - (\mathrm{ETAT(IT)})(1 - ((1)/ \\ & (\mathrm{PRT(IT)}))^{((\mathrm{GAMT(IT)} - 1)/(\mathrm{GAMT(IT)}))})\end{aligned}$$

Compute exit Mach number:

8. $\mathrm{XMTE(IT)} = \mathrm{(WT(IT))((RGT(IT))(TTDS(IT))/((32.174)(GAMT(IT))))^{.5}/((ATE(IT))(PTES(IT)))}$

Compute exit total pressure:

9. $\mathrm{PTE(IT)} = \mathrm{(PTES(IT))(1 + ((GAMT(IT) - 1)/(2))(XMTE(IT))^{2})^{((GAMT(IT))/(GAMT(IT) - 1))}}$

Compute exit total temperature:

10. $\mathrm{TTD(IT)} = \mathrm{(TTDS(IT))((PTE(IT))/(PTES(IT)))^{((GAMT(IT) - 1)/(GAMT(IT)))}}$

Return

SUBROUTINE TURBIN NOMENCLATURE

Symbol	Description	Units	Reference
ATE	Turbine exit flange area	in^2	/CTURB/, 8
ATI	Turbine inlet nozzle area	in^2	/CTURB/, 2
C7	Array of coefficients for turbine efficiency curve	—	/CTURB/, 5
CPT	Turbine gas specific heat	btu/lb-^{O}R	/CTURB/, 3
CST	Turbine gas inlet characteristic velocity	ft/sec	/CTURB/, 2
DELHT	Gas enthalpy change across turbine	btu/lb	/CTURB/, 3, 4, 6
DW	Turbine wheel diameter	in	/CTURB/, 4
ETAT	Turbine efficiency	—	/CTURB/, 5, 6, 7
GAMT	Turbine gas specific heat ratio	—	/CTURB/, 3, 7, 8, 9, 10
HPT	Turbine horsepower	bhp	/CTURB/, 6
IT	Index designating storage location of turbine performance data	—	CALL, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
PRT	Turbine pressure ratio	—	/CTURB/, 1, 3, 5, 7
PTE	Turbine exit total pressure	psia	/CTURB/, 9, 10
PTES	Turbine exit static pressure	psia	/CTURB/, 1, 8, 9, 10

SUBROUTINE TURBIN NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PTI	Turbine inlet total pressure	psia	/CTURB/, 1, 2
RGT	Turbine gas constant	ft/°R	/CTURB/, 8
TTD	Turbine exit total temperature	°R	/CTURB/, 10
TTDS	Turbine exit static temperature	°R	/CTURB/, 7, 8, 10
TTI	Turbine inlet total temperature	°R	/CTURB/, 3, 7
UVRAT	Turbine velocity ratio	—	/CTURB/, 4, 5
WT	Turbine gas flow rate	lb/sec	/CTURB/, 2, 6, 8
XMTE	Turbine exit gas Mach number	—	/CTURB/, 8, 9
XNT	Turbine speed	rpm	/CTURB/, 4
XNTST	Number of turbine pressure stages	—	/CTURB/, 4

```
$IBFTC TURBIN  REF,LIST
      SUBRØUTINE TURBIN (IT)
      CØMMØN /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
      PRT(IT)=PTI(IT)/PTES(IT)
      WT(IT)=32.174*ATI(IT)*PTI(IT)/CST(IT)
      DELHT(IT)=CPT(IT)*TTI(IT)*(1.0-(1.0/PRT(IT)) **((GAMT(IT)-1.0)/
     1GAMT(IT)))
      UVRAT(IT)=2.7576E-5*DW(IT)*XNT(IT)/(DELHT(IT)/XNTST(IT))**.5
      ETAT(IT)=(C7(1,IT)+C7(2,IT)*PRT(IT)+C7(3,IT)*PRT(IT)**2)*UVRAT(IT)
     1+(C7(4,IT)+C7(5,IT)*PRT(IT)+C7(6,IT)*PRT(IT)**2)*UVRAT(IT)**2
     2+C7(7,IT)+C7(8,IT)*PRT(IT)+C7(9,IT)*PRT(IT)**2
      HPT(IT)=1.414836*WT(IT)*ETAT(IT)*DELHT(IT)
      TTDS(IT)=TTI(IT)*(1.0-ETAT(IT)*(1.0-(1.0/PRT(IT))**((GAMT(IT)-1.0)
     1/GAMT(IT))))
      XMTE(IT)=WT(IT)/ATE(IT)/PTES(IT)*(RGT(IT)*TTDS(IT)/32.174/GAMT(IT)
     1)**.5
      PTE(IT)=PTES(IT)*(1.0+(GAMT(IT)-1.0)/2.0*XMTE(IT)**2)**(GAMT(IT)/(
     1GAMT(IT)-1.0))
      TTD(IT)=TTDS(IT)*(PTE(IT)/PTES(IT))**((GAMT(IT)-1.0)/GAMT(IT))
      RETURN
      END
```

FIGURE 89. SYMBOLIC LISTING OF SUBROUTINE TURBIN

LXX. SUBROUTINE TURMAP

This subroutine computes the performance of a turbine by using a turbine map.

CALLING SEQUENCE

The calling sequence is:

CALL TURMAP (IN)

SOLUTION METHOD

Set LOX-fuel side indicator.

1. $\mathrm{IT} = \mathrm{IN}$

2. $\mathrm{PRT(IT)} = \mathrm{(PTI(IT))/(PTES(IT))}$

3. $$\begin{aligned}\mathrm{PP(IT)} &= \mathrm{(C8(1,\ IT) + (C8(2,\ IT))\ (PRT(IT)))} \\ &+ \mathrm{(C8(3,\ IT) + (C8(4,\ IT))\ (PRT(IT)))} \\ &\ \mathrm{(DW(IT))\ (XNT(IT))/((RGT(IT))\ (TTI(IT)))^{.5}}\end{aligned}$$

4. $\mathrm{HPT(IT)} = \mathrm{(PP(IT))\ (PTES(IT))\ (ATI(IT))\ (XNT(IT))\ (DW(IT))}$

5. $$\begin{aligned}\mathrm{DELHT(IT)} &= \mathrm{(CPT(IT))\ (TTI(IT))\ (1.\ -} \\ &\quad \mathrm{((1.)/(PRT(IT)))^{((GAMT(IT)\ -\ 1.)/(GAMT(IT)))})}\end{aligned}$$

6. $\mathrm{ETAT\,(IT)} = (\mathrm{HPT\,(IT)})/((\mathrm{WT\,(IT)})\,(\mathrm{DELHT\,(IT)})\,(1.414836))$

7. $\mathrm{TTDS\,(IT)} = (\mathrm{TTI\,(IT)})\,(1. - (\mathrm{ETAT\,(IT)})\,(1. - ((1.)/(\mathrm{PRT\,(IT)}))^{((\mathrm{GAMT\,(IT)} - 1.)/(\mathrm{GAMT\,(IT)}))}))$

8. $\mathrm{XMTE\,(IT)} = (\mathrm{WT\,(IT)})/((\mathrm{ATE\,(IT)})\,(\mathrm{PTES\,(IT)}))\,((\mathrm{RGT\,(IT)})\,(\mathrm{TTDS\,(IT)})/((\mathrm{GAMT\,(IT)})\,(32.174)))^{.5}$

9. $\mathrm{PTE\,(IT)} = (\mathrm{PTES\,(IT)})\,((1. + ((\mathrm{GAMT\,(IT)} - 1.)/(2.))\,(\mathrm{XMTE\,(IT)})^{2})^{((\mathrm{GAMT\,(IT)})/(\mathrm{GAMT\,(IT)} - 1.))})$

10. $\mathrm{TTD\,(IT)} = (\mathrm{TTDS\,(IT)})\,(((\mathrm{PTE\,(IT)})/(\mathrm{PTES\,(IT)}))^{((\mathrm{GAMT\,(IT)} - 1.)/(\mathrm{GAMT\,(IT)}))})$

11. $\mathrm{UVRAT\,(IT)} = (2.7576)\,(10^{-5})\,(\mathrm{DW\,(IT)})\,(\mathrm{XNT\,(IT)})/((\mathrm{DELHT\,(IT)})/(\mathrm{XNTST\,(IT)}))^{.5}$

Return.

SUBROUTINE TURMAP NOMENCLATURE

Symbol	Description	Units	Reference
ATE	Turbine exit flange area	in^2	/CTURB/, 8
ATI	Turbine inlet nozzle area	in^2	/CTURB/, 4
CPT	Turbine gas specific heat	btu/lbm °R	/CTURB/, 5
C8	Array of coefficients for turbine	—	/CTURMP/, 3
DELHT	Gas enthalpy change across turbine	btu/lbm	/CTURB/, 5, 6, 11
DW	Turbine wheel diameter	in	/CTURB/, 3, 4, 11
ETAT	Turbine efficiency	—	/CTURB/, 6, 7
GAMT	Turbine gas specific heat ratio	—	/CTURB/, 5, 7, 8, 9, 10
HPT	Turbine horsepower	bhp	/CTURB/, 4, 6
IN	Indicator	—	CALL, 1
IT	Indicator (=IN)	—	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11
PP	Turbine power parameter	—	/CTURMP/, 4
PRT	Turbine pressure ratio	—	/CTURB/, 2, 3, 5, 7
PTE	Turbine exit total pressure	lbf/in^2	/CTURB/, 9, 10
PTES	Turbine exit static pressure	lbf/in^2	/CTURB/, 2, 4, 8, 9, 10
PTI	Turbine inlet total pressure	lbf/in^2	/CTURB/, 2

SUBROUTINE TURMAP NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RGT	Turbine gas constant	ft - lbf/lbm - °R	/CTURB/, 3, 8
TTD	Turbine exit total temperature	°R	/CTURB/, 10
TTDS	Turbine exit static temperature	°R	/CTURB/, 7, 8, 10
TTI	Turbine inlet total temperature	°R	/CTURB/, 3, 5, 7
UVRAT	Turbine velocity ratio	—	/CTURB/, 11
WT	Turbine gas mass flow rate	lbm/sec	/CTURB/, 6, 8
XNT	Turbine speed	rpm	/CTURB/, 3, 4, 11
XNTST	Number of turbine pressure stages	—	/CTURB/, 11

```
$IBFTC TURMAP  FULIST,REF
      SUBRØUTINE TURMAP(IN)
C
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),     PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
C
      CØMMØN /CRES/ PRI(20),TRIS(20),PVRI(20),RHRI(20),VHRI(20),WRD(20),
     1DRI(20),DEL4(20),TRDS(20),DLTR(20),RHRD(20),RHRA(20),PRD(20),
     2R(20),VHRD(20),DRD(20),PRDS(20),PVRD(20),DEL5(20),WRI(20),PRIS(20)
C
      CØMMØN /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
C
      CØMMØN /CCHAM/ INDPR(1),CF(1),CFTHE(1),ETCF(1),EPS(1),PNS(1),FC(1)
     1,ATC(1),FAUX(1),FEN(1),DEL7(1),CS(1),CSTHE(1),ETCS(1),WC(1),XMRTC(
     21),WCF(1),WCØ(1)
C
      CØMMØN /CBAL/ KBAL(1)  /CPA/ PA(1) /CRESGG/ RESØG(1),RESFG(1)
C
      CØMMØN /CMAIN/ WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),
     1TN(1),ØRES(4),WG(1),WGØ(1),WGF(1),PCIE(1),XMRE(1),PGG(1),XISPC(1),
     2XISPE(1),FE(1),WEØ(1),WEF(1),TPIR(2)
C
      CØMMØN /H1F1J2/ RGG(1),DEL8(1),DEL9(1),XMREN(1)
C
      CØMMØN /CDEN/TRSG(2),SG(2)
C
      CØMMØN /CGPRØP/      C3(6),C4(6),C5(6),C6(6),CSG,GAMG,CPG,XMWG,
     1XMRG,RG,XØMG
C
      CØMMØN /CTURMP/ PP(2),C8(4,2)
      CØMMØN /CJ2IN/DEL20,DEL21,DEL22,DEL23,DEL24,DEL25,DEL26,CNST1,
     1CNST2,CNST3,CNST4,CNST5,CNST6,PUCØF(6,10),PUSEC(10),PUVA,
     2 EPSM,ABPNI, DMØØ, DMFØ  ,WPRESF, WPRESØ ,DEL27,DEL28,DEL29,CNST7
     3, CNST8, DEL35
      CØMMØN /CJ2ØUT/ RPU,PV,WPUØ,PMAN,PMAN1,TMAN,TAVE,WBP,
     1PBNI,FPL,FPR,XMBPNI,ABPN,DBPN,PTIS(2),TTIS(2)
C
      IT = IN
      PRT(IT) = PTI(IT)/PTES(IT)
      PP(IT)= (C8(1,IT)+C8(2,IT)*PRT(IT))+ (C8(3,IT)+C8(4,IT)*PRT(IT) )
     1* (DW(IT)*XNT(IT)/(RGT(IT)*TTI(IT))**.5)
      HPT(IT)= PP(IT)*PTES(IT)*ATI(IT)*XNT(IT)*DW(IT)
      DELHT(IT)= CPT(IT)*TTI(IT)*(1.-(1./PRT(IT)) **
     1((GAMT(IT)-1.)/GAMT(IT)) )
      ETAT(IT)= HPT(IT)/(WT(IT)*DELHT(IT)*1.414836 )
```

FIGURE 90. SYMBOLIC LISTING OF SUBROUTINE TURMAP

```
 TTDS(IT)=ABS( TTI(IT)*(1.-ETAT(IT)*(1.-(1./PRT(IT)) **
1((GAMT(IT)-1.)/GAMT(IT)) )  )  )
 XMTE(IT)=WT(IT)/(ATE(IT)*PTES(IT))*(RGT(IT)*TTDS(IT)/(GAMT(IT)*
1 32.174) )**.5
 PTE(IT)=PTES(IT)*( 1.+(GAMT(IT)-1.)/2.*XMTE(IT)**2 )**(GAMT(IT)/
1  (GAMT(IT) - 1.  ))
 TTD(IT)=TTDS(IT)*(PTE(IT)/PTES(IT))**((GAMT(IT)-1.)/GAMT(IT))
 UVRAT(IT)=2.7576E-5*DW(IT)*XNT(IT)/(DELHT(IT)/XNTST(IT))**.5
 RETURN
 END
```

FIGURE 90. SYMBOLIC LISTING OF SUBROUTINE TURMAP (cont.)

LXXI. SUBROUTINE TURBEX

This subroutine calculates the operating performance of a rocket engine turbine exhaust system. Either a "straight duct" or a "loop-tube" type system may be simulated.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL TURBEX

This subroutine uses COMMON blocks CTURBX, CTURB, CCHAM and CPA.

SOLUTION METHOD

Determine type of system to be simulated.

1. If CPCX $\leq$ 0, go to 2

   If CPCX $>$ 0, go to 23

Straight Duct

Compute turbine exhaust duct exit temperature and gas characteristic velocity.

2. $TXD = TXI - DELTX$

3. $CSXD = (((32.174)(RGX)(TXD))^{.5})/(XOMX)$

Compute exit critical pressure ratio:

4. $GAMR = (GAMX)/(GAMX - 1)$

5. $PRXDC = ((2)/(GAMX + 1))^{GAMR}$

Compute exit total pressure assuming choked flow:

6. $PXDC = (WX)(CSXD)/((32.174)(AXD))$

Compute exit static pressure based upon 5 and 6:

7. $PXDSC = (PXDC)(PRXDC)$

Compare exit static pressure with ambient pressure:

8. If $PXDSC \leq PA$, go to 9

   If $PXDSC > PA$, go to 16

Unchoked flow; exit static pressure is equal to ambient pressure:

9. $PXDS = PA$

10. $TERM = (32.174)(PXDS)(AXD)/((XOMX)(CSXD))$

Compute exit pressure ratio:

11. $\mathrm{PRXD} = ((((\mathrm{GAMR}^2)(\mathrm{TERM}^4) + (2)(\mathrm{GAMR})((\mathrm{WX})(\mathrm{TERM}))^2)^{.5} - (\mathrm{GAMR})(\mathrm{TERM}^2))/(\mathrm{WX}^2))^{\mathrm{GAMR}}$

Compute exit total pressure:

12. $\mathrm{PXD} = (\mathrm{PXDS})/(\mathrm{PRXD})$

Compute exit static temperature:

13. $\mathrm{TXDS} = (\mathrm{TXD})(\mathrm{PRXD})^{((1)/(\mathrm{GAMR}))}$

Compute exit mach number:

14. $\mathrm{XMXD} = (\mathrm{WX})((\mathrm{RGX})(\mathrm{TXDS})/((32.174)(\mathrm{GAMX})))^{.5}/((\mathrm{AXD})(\mathrm{PXDS}))$

15. Go to 21

Choked flow; exit static pressure is based upon critical pressure ratio:

16. $\mathrm{PXDS} = \mathrm{PXDSC}$

17. $\mathrm{PXD} = \mathrm{PXDC}$

18. $\mathrm{XMXD} = 1$

19. $\mathrm{PRXD} = \mathrm{PRXDC}$

20. $\mathrm{TXDS} = (\mathrm{TXD})(\mathrm{PRXD})^{((1)/(\mathrm{GAMR}))}$

Compute turbine exhaust thrust:

21. $$\begin{aligned} FX = {} & (((GAMR)(AXD)(PXDS))^{2} \\ & + (2)(GAMR)((WX)(XOMX)(CSXD)/(32.174))^{2})^{.5} \\ & + (AXD)(PXDS - PA - (PXDS)(GAMR)) \end{aligned}$$

Compute turbine exhaust system inlet static pressure:

22. $PXIS = PXDS + (RX)(RGX)(TXDS)(WX^{2})/((144)(PXDS))$

Return

Loop-Tube

Compute thrust chamber wall pressure at injection point:

23. $PCX = (CPCX)(PNS)$

Compute turbine exhaust system inlet static pressure:

24. $PXIS = PCX + (RX)(RGX)(TTDS(1))(WX^{2})/((144)(PTES(1)))$

Return

SUBROUTINE TURBEX NOMENCLATURE

Symbol	Description	Units	Reference
AXD	Turbine exhaust discharge area	in^2	/CTURBX/, 6, 10, 14, 21
CPCX	Ratio of thrust chamber wall pressure to nozzle stagnation pressure	—	/CTURBX/, 1, 23
CSXD	Turbine exhaust discharge gas characteristic velocity	ft/sec	/CTURBX/, 3, 6, 10, 21
DELTX	Turbine exhaust system temperature drop	^oR	/CTURBX/, 2
FX	Turbine exhaust thrust	lb	/CTURBX/, 21
GAMR	—	—	4, 5, 11, 13, 20, 21
GAMX	Turbine exhaust gas specific heat ratio	—	/CTURBX/, 4, 5, 14
PA	Ambient pressure	lb/in^2	/CPA/, 8, 9, 21
PCX	Thrust chamber wall pressure	lb/in^2	23, 24
PNS	Thrust chamber nozzle stagnation pressure	lb/in^2	/CCHAM/, 23
PRXD	Turbine exhaust discharge pressure ratio	—	/CTURBX/, 11, 12, 13, 19, 20
PRXDC	Turbine exhaust discharge critical pressure ratio	—	5, 7, 19
PTES	Turbine discharge static pressure	lb/in^2	/CTURB/, 24

SUBROUTINE TURBEX NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PXD	Turbine exhaust discharge total pressure	lb/in^2	/CTURBX/, 12, 17
PXDC	Turbine exhaust discharge total pressure	lb/in^2	6, 7, 17
PXDS	Turbine exhaust discharge static pressure	lb/in^2	/CTURBX/, 9, 10, 12, 14, 16, 21, 22
PXDSC	Turbine exhaust discharge static pressure	lb/in^2	7, 8, 16
PXIS	Turbine exhaust inlet static pressure	lb/in^2	/CTURBX/, 22, 24
RGX	Turbine exhaust gas constant	$ft/^oR$	/CTURBX/, 3, 14, 22, 24
RX	Turbine exhaust system resistance	sec^2/ft^5	/CTURBX/, 22, 24
TERM	—	lb/sec	10, 11
TTDS	Turbine discharge static temperature	^oR	/CTURB/, 24
TXD	Turbine exhaust discharge total temperature	^oR	/CTURBX/, 2, 3, 13, 20
TXDS	Turbine exhaust discharge static temperature	^oR	/CTURBX/, 13, 14, 20, 22
TXI	Turbine exhaust inlet total temperature	^oR	/CTURBX/, 2
WX	Turbine exhaust flow rate	lb/sec	/CTURBX/, 6, 11, 14, 21, 22, 24

SUBROUTINE TURBEX NOMENCLAURES (cont.)

Symbol	Description	Units	Reference
XMXD	Turbine exhaust discharge Mach number	—	/CTURBX/, 14, 18
XOMX	—	—	/CTURBX/, 3, 10, 21

```
$IBFTC TRBX    FULIST,REF
      SUBROUTINE TURBEX
      COMMON /CTURBX/ TXD,TXI,DELTX,PXDS,PXIS,RX,RGX,TXIS,WX,CSXD,XOMX,
     1GAMX,XMXD, PRXD,PXD,AXD,FX,TXDS,CPCX
      COMMON /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
      COMMON /CCHAM/  INDPR ,CF  ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCO
      COMMON /CPA/PA
      IF (CPCX) 10,10,90
   10 TXD=TXI-DELTX
      CSXD=(32.174*RGX*TXD)**.5/XOMX
      GAMR=GAMX/(GAMX-1.0)
      PRXDC=(2.0/(GAMX+1.0))**GAMR
      PXDC=WX*CSXD/32.174/AXD
      PXDSC=PXDC*PRXDC
      IF(PXDSC-PA)50,50,75
   50 PXDS=PA
      TERM=32.174*PXDS*AXD/XOMX/CSXD
      PRXD=(((GAMR**2*TERM**4+2.0*GAMR*(WX*TERM)**2)**.5
     1-GAMR*TERM**2)/WX**2)**GAMR
      PXD=PXDS/PRXD
      TXDS=TXD*PRXD**(1.0/GAMR)
      XMXD=WX/AXD/PXDS*(RGX*TXDS/32.174/GAMX)**.5
      GO TO 80
   75 PXDS=PXDSC
      PXD=PXDC
      XMXD=1.0
      PRXD=PRXDC
      TXDS=TXD*PRXD**(1.0/GAMR)
   80 FX=((GAMR*AXD*PXDS)**2+2.0*GAMR*(WX*XOMX*CSXD/32.174)**2)**.5 +
     1AXD*(PXDS-PA-PXDS*GAMR)
      PXIS=PXDS+RX*RGX*TXDS*WX**2/144.0/PXDS
      RETURN
   90 PCX=CPCX*PNS
      PXIS=PCX+RX*RGX*TTDS(1)*WX**2/144.0/PTES(1)
      RETURN
      END
```

FIGURE 91. SYMBOLIC LISTING OF SUBROUTINE TURBEX

LXXII. SUBROUTINE GPROP

This subroutine calculates the gas generator hot gas properties using fifth-degree polynomial equations.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL GPROP(T)

This subroutine uses COMMON block CGPROP.

SOLUTION METHOD

Compute specific heat ratio:

1. $$GAMG = C3(1) + (C3(2))(T) + (C3(3))(T^2) + (C3(4))(T^3) + (C3(5))(T^4) + (C3(6))(T^5)$$

Compute specific heat:

2. $$CPG = C4(1) + (C4(2))(T) + (C4(3))(T^2) + (C4(4))(T^3) + (C4(5))(T^4) + (C4(6))(T^5)$$

Compute molecular weight:

3. $\mathrm{XMWG} = \mathrm{C5}(1) + (\mathrm{C5}(2))(\mathrm{T}) + (\mathrm{C5}(3))(\mathrm{T}^2) + (\mathrm{C5}(4))(\mathrm{T}^3) + (\mathrm{C5}(5))(\mathrm{T}^4) + (\mathrm{C5}(6))(\mathrm{T}^5)$

Compute mixture ratio:

4. $\mathrm{XMRG} = \mathrm{C6}(1) + (\mathrm{C6}(2))(\mathrm{T}) + (\mathrm{C6}(3))(\mathrm{T}^2) + (\mathrm{C6}(4))(\mathrm{T}^3) + (\mathrm{C6}(5))(\mathrm{T}^4) + (\mathrm{C6}(6))(\mathrm{T}^5)$

Compute gas constant:

5. $\mathrm{RG} = (1545.31)/(\mathrm{XMWG})$

6. $\mathrm{XOMG} = ((\mathrm{GAMG})(((2)/(\mathrm{GAMG}+1))^{((\mathrm{GAMG}+1)/(\mathrm{GAMG}-1))}))^{.5}$

Compute characteristic velocity:

7. $\mathrm{CSG} = (((32.174)(\mathrm{T} + 459.688)(\mathrm{RG}))^{.5})/(\mathrm{XOMG})$

Return

SUBROUTINE GPROP NOMENCLATURE

Symbol	Description	Units	Reference
C3	Array of coefficients for specific heat ratio curve	—	/CGPROP/, 1
C4	Array of coefficients for specific heat curve	—	/CGPROP/, 2
C5	Array of coefficients for molecular weight curve	—	/CGPROP/, 3
C6	Array of coefficients for mixture ratio curve	—	/CGPROP/, 4
CPG	Gas specific heat	btu/lb-$^{\circ}$R	/CGPROP/, 2
CSG	Gas characteristic velocity	ft/sec	/CGPROP/, 7
GAMG	Gas specific heat ratio	—	/CGPROP/, 1, 6
RG	Gas constant	ft/$^{\circ}$R	/CGPROP/, 5, 7
T	Gas temperature	$^{\circ}$F	CALL, 1, 2, 3, 4, 7
XMRG	Misture ratio	—	/CGPROP/, 4
XMWG	Gas molecular weight	—	/CGPROP/, 3, 5
XOMG	—	—	/CGPROP/, 6, 7

```
$IBFTC GPRØP   REF,LIST
      SUBRØUTINE GPRØP (T)
      CØMMØN /CGPRØP/      C3(6),C4(6),C5(6),C6(6),CSG,GAMG,CPG,XMWG,
     1XMRG,RG,XØMG
      GAMG=C3(1)+C3(2)*T+C3(3)*T**2+C3(4)*T**3+C3(5)*T**4+C3(6)*T**5
      CPG=C4(1)+C4(2)*T+C4(3)*T**2+C4(4)*T**3+C4(5)*T**4+C4(6)*T**5
      XMWG=C5(1)+C5(2)*T+C5(3)*T**2+C5(4)*T**3+C5(5)*T**4+C5(6)*T**5
      XMRG=C6(1)+C6(2)*T+C6(3)*T**2+C6(4)*T**3+C6(5)*T**4+C6(6)*T**5
      RG=1545.31/XMWG
      XØMG=(GAMG*((2.0/(GAMG+1.0))**((GAMG+1.0)/(GAMG-1.0))))**.5
      CSG=(32.174*(T+459.688)*RG)**.5/XØMG
      RETURN
      END
```

FIGURE 92. SYMBOLIC LISTING OF SUBROUTINE GPROP

LXXIII. SUBROUTINE RES

This subroutine computes the pressure drop across a resistance through which a liquid flows. The following data are required:

a. Resistance inlet total pressure

b. Liquid flow rate across resistance

c. Resistance inlet liquid temperature

d. Liquid temperature change across resistance

e. Resistance inlet and discharge flow diameters

f. Resistance liquid characteristics

g. Resistance value

h. Flow direction

CALLING SEQUENCE

The subroutine calling sequence is:

CALL RES (JR, IL, IWR, XWD, KP)

This subroutine uses COMMON blocks CRES and CDEN.

SOLUTION METHOD

Initialize control index:

1. IR = JR

Approximate resistance inlet static pressure:

2. PRIS(IR) = PRI(IR)

Initialize iteration control loop counter:

3. IDLR = 0

Compute resistance inlet propellant density:

4. CALL PRPDEN (KP, PRIS(IR), TRIS(IR), TRSG(IL), SG(IL), PVRI(IR), RHRI(IR))

Compute resistance inlet velocity head:

5. $VHRI(IR) = (3.627845)(WRI(IR)^2)/((RHRI(IR))(DRI(IR)^4))$

Compute resistance inlet static pressure:

6. PRISP = PRI(IR) - VHRI(IR)

Compare assumed and computed values of resistance inlet static pressure. If convergence criterion is not satisfied, iterate on PRIS:

7. If $|PRIS(IR) - PRISP| \leq DEL4(IR)$, go to 10

   If $|PRIS(IR) - PRISP| > DEL4(IR)$, go to 8

```
8.  CALL ESTM (IDLR, 2, REST, DREST, PRIS(IR) - PRISP,
        PRIS(IR), LUP1)

9.  Go to 4

Compute resistance discharge temperature:

10. TRDS(IR) = TRIS(IR) + DLTR(IR)

Approximate resistance discharge propellant density:

11. RHRD(IR) = RHRI(IR)

Initialize iteration loop control counter:

12. IDLR = 0

Compute resistance propellant average density:

13. RHRA(IR) = (RHRI(IR) + RHRD(IR))/(2.0)

Determine flow rate upon which pressure drop is to be based:

14. If IWR < 0, go to 15
    If IWR ≥ 0, go to 17

15. WR = WRI(IR)

16. Go to 18

17. WR = WRD(IR)

Compute resistance discharge total pressure:
```

18. PRD(IR) = PRI(IR) - (XWD)(R(IR))(WR^{2})/((144.0)
(RHRA(IR)))

Compute resistance discharge velocity head:

19. VHRD(IR) = (3.627845)(WRD(IR)^{2})/((RHRD(IR))
(DRD(IR)^{4}))

Compute resistance discharge static pressure:

20. PRDS(IR) = PRD(IR) - VHRD(IR)

Determine resistance discharge propellant density:

21. CALL PRPDEN (KP, PRDS(IR), TRDS(IR), TRSG(IL), SG(IL),
PVRD(IR), RHRDP)

Compare assumed and calculated values of resistance discharge propellant density. If convergence criterion is not satisfied, iterate on RHRD:

22. If | RHRD(IR) - RHRDP | ≤ DEL5(IR), go to 25
If | RHRD(IR) - RHRDP | > DEL5(IR), go to 23
23. CALL ESTM (IDLR, 2, REST, DREST, RHRD(IR) - RHRDP,
RHRD(IR), LUP2)
24. Go to 13
25. Return

SUBROUTINE RES NOMENCLATURE

Symbol	Description	Units	Reference
DEL4	Resistance inlet static pressure tolerance	psi	/CRES/, 7
DEL5	Resistance exit density tolerance	lb/ft^3	/CRES/, 22
DLTR	Liquid temperature change across resistance	^oF	/CRES/, 10
DRD	Resistance exit diameter	in	/CRES/, 19
DREST	Storage array for dependent argument in Subroutine ESTM	—	DIM, 8, 23
DRI	Resistance inlet diameter	in	/CRES/, 5
IDLR	Iteration loop counter	—	3, 8, 12, 23
IL	Index designating storage location of liquid reference temperature and specific gravity	—	CALL, 4, 21
IR	JR	—	1, 2, 4, 5, 6, 7, 8, 10, 11, 13, 15, 17, 18, 19, 20, 21, 22, 23
IWR	Resistance flow rate indicator +1 = Compute pressure drop based on exit flow rate -1 = Compute pressure drop based on inlet flow rate	—	CALL, 14

SUBROUTINE RES NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
JR	Index designating storage location of resistance performance data	—	CALL, 1
KP	Propellant density indicator	—	CALL, 4, 21
LUP1	—	—	DATA, 8
LUP2	—	—	DATA, 23
PRD	Resistance exit total pressure	psia	/CRES/, 18, 20
PRDS	Resistance exit static pressure	psia	/CRES/, 20, 21
PRI	Resistance inlet total pressure	psia	/CRES/, 2, 6, 18
PRIS	Resistance inlet static pressure	psia	/CRES/, 2, 4, 7, 8
PRISP	Resistance inlet static pressure	psia	6, 7, 8
PVRD	Resistance exit liquid vapor pressure	psia	/CRES/, 21
PVRI	Resistance inlet liquid vapor pressure	psia	/CRES/, 4
R	Resistance value	sec^2/ft^5	/CRES/, 18
REST	Storage array for independent argument in Subroutine ESTM	—	DIM, 8, 23

SUBROUTINE RES NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RHRA	Average liquid density across resistance	lb/ft^3	/CRES/, 13, 18
RHRD	Resistance exit liquid density	lb/ft^3	/CRES/, 11, 13, 19, 22, 23
RHRDP	Resistance exit liquid density	lb/ft^3	21, 22, 23
RHRI	Resistance inlet liquid density	lb/ft^3	/CRES/, 4, 5, 11, 13
SG	Liquid specific gravity at reference temperature	—	/CDEN/, 4, 21
TRDS	Resistance exit liquid temperature	°F	/CRES/, 10, 21
TRIS	Resistance inlet liquid temperature	°F	/CRES/, 4, 10
TRSG	Liquid reference temperature	°F	/CDEN/, 4, 21
VHRD	Resistance exit velocity head	psia	/CRES/, 19, 20
VHRI	Resistance inlet velocity head	psia	/CRES/, 5, 6
WR	Resistance mass flow rate	lb/sec	15, 17, 18
WRD	Resistance exit mass flow rate	lb/sec	/CRES/, 17, 19
WRI	Resistance inlet mass flow rate	lb/sec	/CRES/, 5, 15

SUBROUTINE RES NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
XWD	Resistance liquid flow direction indicator +1.0 = Upstream conditions known -1.0 = Downstream conditions known	—	CALL, 18

```
$IBFTC RES     REF,LIST
      SUBROUTINE RES (JR,IL,IWR,XWD,KP)
      COMMON /CRES/ PRI(20),TRIS(20),PVRI(20),RHRI(20),VHRI(20),WRD(20),
     1DRI(20),DEL4(20),TRDS(20),DLTR(20),RHRD(20),RHRA(20),PRD(20),
     2R(20),VHRD(20),DRD(20),PRDS(20),PVRD(20),DEL5(20),WRI(20),PRIS(20)
      COMMON /CDEN/TRSG(2),SG(2)
      DIMENSION REST(20),DREST(20)
      DATA LUP1/6HPRIS  /,LUP2/6HRHRD  /
      IR=JR
      PRIS(IR)=PRI(IR)
      IDLR=0
    2 CALL PRPDEN (KP,PRIS(IR),TRIS(IR),TRSG(IL),SG(IL),PVRI(IR),RHRI(IR
     1))
    6 VHRI(IR)=3.627845*WRI(IR)**2/RHRI(IR)/DRI(IR)**4
      PRISP=PRI(IR)-VHRI(IR)
      IF(ABS(PRIS(IR)-PRISP)-DEL4(IR)) 9,9,8
    8 CALL ESTM (IDLR,2,REST,DREST,PRIS(IR)-PRISP,PRIS(IR),LUP1)
      GO TO 2
    9 TRDS(IR)=TRIS(IR)+DLTR(IR)
      RHRD(IR)=RHRI(IR)
      IDLR=0
   11 RHRA(IR)=(RHRI(IR)+RHRD(IR))/2.0
      IF(IWR)12,13,13
   12 WR=WRI(IR)
      GO TO 14
   13 WR=WRD(IR)
   14 PRD(IR)=PRI(IR)-XWD*R(IR)*WR    **2/(144.0*RHRA(IR))
      VHRD(IR)=3.627845*WRD(IR)**2/RHRD(IR)/DRD(IR)**4
      PRDS(IR)=PRD(IR)-VHRD(IR)
      CALL PRPDEN (KP,PRDS(IR),TRDS(IR),TRSG(IL),SG(IL),PVRD(IR),RHRDP)
      IF(ABS(RHRD(IR)-RHRDP)-DEL5(IR)) 21,21,20
   20 CALL ESTM (IDLR,2,REST,DREST,RHRD(IR)-RHRDP,RHRD(IR),LUP2)
      GO TO 11
   21 RETURN
      END
```

FIGURE 93. SYMBOLIC LISTING OF SUBROUTINE RES

LXXIV. SUBROUTINE TCPROP

This subroutine computes the theoretical thrust coefficient and characteristic velocity for liquid oxygen / RP-1 and liquid oxygen / liquid hydrogen propellant combinations. Curve fits are used to obtain the desired quantities.

The liquid oxygen / RP-1 computations are based upon frozen composition, while the liquid oxygen / liquid hydrogen computations are based upon shifting composition.

CALLING SEQUENCE

The calling sequence is:

```
CALL TCPROP (INDP, XMRC, PC, EP, CSTH, CFTH)
```

The variable INDP is used to denote the propellant combination of interest, such that:

INDP = 1 for liquid oxygen / RP-1

INDP = 2 for liquid oxygen / liquid hydrogen

SOLUTION METHOD

Determine propellant combination of interest:

1. If INDP = 1, go to 2

If INDP = 2, go to 55

Liquid Oxygen/RP-1

Set up variables for characteristic velocity calculation:

2. C1 = -8244.2601
3. C2 = 16684.784
4. C3 = -6172.4708
5. C4 = 2733.3791
6. C5 = -547.23793
7. C6 = 42.785606
8. C7 = -3899.8009
9. C8 = 5576.0354
10. C9 = -2738.1352
11. C10 = 578.74455
12. C11 = -36.201772
13. C12 = -1.664795
14. C13 = 700.30026
15. C14 = -1245.0609
16. C15 = 830.64172
17. C16 = -264.34673

18. $C17 = 40.430077$

19. $C18 = -2.3862895$

20. $LNPC = Ln(PC)$

Compute characteristic velocity:

21. $$\begin{aligned} CSTH = {} & (C1 + (C2)(XMRC) + (C3)(XMRC^{2}) + (C4)(XMRC^{3}) \\ & + (C5)(XMRC^{4}) + (C6)(XMRC^{5}) + (C7)(LNPC) \\ & + (C8)(XMRC)(LNPC) + (C9)(XMRC^{2})(LNPC) \\ & + (C10)(XMRC^{3})(LNPC) + (C11)(XMRC^{4})(LNPC) \\ & + (C12)(XMRC^{5})(LNPC) + (C13)(LNPC^{2}) \\ & + (C14)(XMRC)(LNPC^{2}) + (C15)(XMRC^{2})(LNPC^{2}) \\ & + (C16)(XMRC^{3})(LNPC^{2}) + (C17)(XMRC^{4})(LNPC^{2}) \\ & + (C18)(XMRC^{5})(LNPC^{2}))/(XMRC^{1.5}) \end{aligned}$$

Set up variables for thrust coefficient calculation:

22. $A1 = 1.7668722$

23. $A2 = (4.60582)(10^{-1})$

24. $A3 = (-1.6518414)(10^{-1})$

25. $A4 = (-6.6045816)(10^{-2})$

26. $A5 = (3.6782834)(10^{-2})$

27. $A6 = (-7.417031)(10^{-1})$

28. $A7 = 2.3071612$

29. A8 = -5.6948143

30. A9 = 6.4492257

31. A10 = -3.000598

32. A11 = -1.1999626

33. A12 = $(5.8599377)(10^{-1})$

34. A13 = $(-4.327889)(10^{-2})$

35. A14 = $(-1.165337)(10^{-1})$

36. A15 = $(5.3700486)(10^{-1})$

37. A16 = $(1.0577987)(10^{-2})$

38. A17 = .99303665

39. A18 = $(-7.8752103)(10^{-4})$

40. A19 = $(1.9191969)(10^{-3})$

41. A20 = $(-3.5133906)(10^{-3})$

42. A21 = $(1.1578578)(10^{-4})$

43. A22 = $(3.6707999)(10^{-5})$

44. A23 = $(4.3197244)(10^{-4})$

45. A24 = $(4.1092356)(10^{-6})$

46. A25 = $(-5.2376584)(10^{-5})$

47. A26 = $(1.4744636)(10^{-5})$

48. ERC = 1.0 - exp((-3)(XMRC) + 3.9)

49. EPRT = $EP^{.5}$

50. $ERC2 = ERC^2$

51. $ERC3 = ERC^3$

52. $LNRC = \mathrm{Ln}(XMRC)$

Compute thrust coefficient:

53. $$\begin{aligned} CFTH = {} & (A1 + (A2)(LNRC) + (A3)(LNRC^2) + (A4)(LNRC^3) \\ & + (A5)(LNRC^4) + (A6)/(EPRT) + (A7)/(EP) \\ & + (A8)/(EP^{1.5}) + (A9)/(EP^2) + (A10)(EP^{2.5}) \\ & + (A11)(LNRC)/(EPRT) + (A12)(LNRC)/(EP) \\ & + (A13)(LNRC^3)/ EPRT) + (A14)(LNRC^3)/(EP) \\ & + (A15)(LNRC^2)/(EPRT))(A17) + A16 \\ & + (A19)(ERC2)(EPRT)(LNPC) \\ & + (A20)(ERC3)(EPRT)(LNPC) \\ & + (A21)(ERC)(EPRT)(LNPC^2) \\ & + (A22)(ERC2)(EPRT)(LNPC^2) \\ & + (A23)(ERC3)(EPRT)(LNPC^2) \\ & + (A24)(ERC)(EPRT)(LNPC^3) \\ & + (A25)(ERC2)(EPRT)(LNPC^3) \\ & + (A26)(ERC3)(EPRT)(LNPC^3) \\ & + (A18)(ERC)(EPRT)(LNPC) \end{aligned}$$

54. Go to 117

Liquid Oxygen/Liquid Hydrogen

Set up variables for characteristic velocity calculation:

55. $C1 = (-1095.6897)(10^4)$
56. $C2 = (1177.5202)(10^5)$
57. $C3 = (-4888.403)(10^5)$
58. $C4 = (9945.6205)(10^5)$
59. $C5 = (-9962.3336)(10^5)$
60. $C6 = (3935.112)(10^5)$
61. $C7 = (4608.3401)(10^3)$
62. $C8 = (-4698.5585)(10^4)$
63. $C9 = (1902.431)(10^5)$
64. $C10 = (-3817.931)(10^5)$
65. $C11 = (3792.6895)(10^5)$
66. $C12 = (-1489.9681)(10^5)$
67. $C13 = (-4000.8759)(10^2)$
68. $C14 = (4096.6833)(10^3)$
69. $C15 = (-1663.2141)(10^4)$
70. $C16 = (3343.4745)(10^4)$
71. $C17 = (-3324.822)(10^4)$
72. $C18 = (1306.9946)(10^4)$

73. LNPC = Ln (PC)

74. FMU = XMRC

75. FMU12 = $\mathrm{XMRC}^{.5}$

76. FMU32 = $\mathrm{XMRC}^{1.5}$

77. FMU2 = XMRC^{2}

78. FMU52 = $\mathrm{XMRC}^{2.5}$

Compute characteristic velocity:

79. CSTH = (C1 + (C2)/(FMU12) + (C3)/(FMU)
+ (C4)/(FMU32) + (C5)/(FMU2)
+ (C6)/(FMU52) + (LNPC)(C7
+ (C8)/(FMU12) + (C9)/(FMU)
+ (C10)/(FMU32) + (C11)/(FMU2)
+ (C12)/(FMU52))
+ (LNPC^{2})(C13 + (C14)/(FMU12)
+ (C15)/(FMU) + (C16)/(FMU32)
+ (C17)/(FMU2) + (C18)/(FMU52)))/(FMU32)

Set up variables for thrust coefficient calculation:

80. A1 = 1.5283438

81. A2 = 1.0474001

82. A3 = -1.1688996

83. A4 = .6142599
84. A5 = -.104119
85. A6 = -.70660583
86. A7 = .6324424
87. A8 = .21869441
88. A9 = -1.9625851
89. A10 = 1.4593124
90. A11 = -.141305
91. A12 = $(-6.6680344)(10^{-2})$
92. A13 = -.12259043
93. A14 = $(7.9118236)(10^{-2})$
94. A15 = $(9.2647573)(10^{-2})$
95. A16 = $(-5.7726734)(10^{-3})$
96. A17 = 1.0038964
97. A18 = -129.79613
98. A19 = 266.52457
99. A20 = -136.73502
100. A21 = 44.700008
101. A22 = -91.783763
102. A23 = 47.085807

103. $A24 = -3.7654116$

104. $A25 = 7.7314006$

105. $A26 = -3.9661493$

106. $A27 = (1.8912772)(10^{-4})$

107. $A28 = (2.1775238)(10^{-4})$

108. $FLNMU = \mathrm{Ln}(XMRC)$

109. $G = 1.0 - \exp((-3)(XMRC) + 3.0)$

110. $G2 = G^2$

111. $FLNP = LNPC$

112. $E12 = EP^{.5}$

113. $E32 = EP^{1.5}$

114. $E2 = EP^2$

115. $E52 = EP^{2.5}$

Compute thrust coefficient:

116. $$\begin{aligned} CFTH = {} & (A1 + (A2)(FLNMU) + (A3)(FLNMU^2) \\ & + (A4)(FLNMU^3) + (A5)(FLNMU^4) + (A6)/(E12) \\ & + (A7)/(EP) + (A8)/(E32) + (A9)/(E2) + (A10)/(E52) \\ & + (A11)(FLNMU)/(E12) + (A12)(FLNMU)/(EP) \\ & + (A13)(FLNMU^3)/(E12) + (A14)(FLNMU^3)/(EP) \\ & + (A15)(FLNMU^2)/(E12))(A17) + (A16) \end{aligned}$$

```
          + (G)(FLNP)(E12)(A18 + (A19)(G) + (A20)(G2))
          + (G)($FLNP^2$)(E12)(A21 + (A22)(G)+ (A23)(G2))
          + (G)($FLNP^3$)(E12)(A24 + (A25)(G) + (A26)(G2))
          + (FMU)(exp((-PC)/(243.0)))(A27
          + (A28)(FMU)(E12))
117.  Return
```

SUBROUTINE TCPROP NOMENCLATURE

Symbol	Description	Units	Reference
A1	Coefficient of thrust coefficient equation	—	22, 53, 80, 116
A2	"	—	23, 53, 81, 116
A3	"	—	24, 53, 82, 116
A4	"	—	25, 53, 83, 116
A5	"	—	26, 53, 84, 116
A6	"	—	27, 53, 85, 116
A7	"	—	28, 53, 86, 116
A8	"	—	29, 53, 87, 116
A9	"	—	30, 53, 88, 116
A10	"	—	31, 53, 89, 116
A11	"	—	32, 53, 90, 116
A12	"	—	33, 53, 91, 116
A13	"	—	34, 53, 92, 116
A14	"	—	35, 53, 93, 116
A15	"	—	36, 53, 94, 116
A16	"	—	37, 53, 95, 116
A17	"	—	38, 53, 96, 116
A18	"	—	39, 53, 97, 116

SUBROUTINE TCPROP NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
A19	Coefficient of thrust coefficient equation (cont.)	—	40, 53, 98, 116
A20	"	—	41, 53, 99, 116
A21	"	—	42, 53, 100, 116
A22	"	—	43, 53, 101, 106
A23	"	—	44, 53, 102, 116
A24	"	—	45, 53, 103, 116
A25	"	—	46, 53, 104, 116
A26	"	—	47, 53, 105, 116
A27	"	—	106, 116
A28	"	—	107, 116
C1	Coefficient of characteristic velocity equation	—	2, 21, 55, 79
C2	"	—	3, 21, 56, 79
C3	"	—	4, 21, 57, 79
C4	"	—	5, 21, 58, 79
C5	"	—	6, 21, 59, 79
C6	"	—	7, 21, 60, 79
C7	"	—	8, 21, 61, 79

SUBROUTINE TCPROP NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
C8	Coefficient of characteristic velocity equation (cont.)	—	9, 21, 62, 79
C9	"	—	10, 21, 63, 79
C10	"	—	11, 21, 64, 79
C11	"	—	12, 21, 65, 79
C12	"	—	13, 21, 66, 79
C13	"	—	14, 21, 67, 79
C14	"	—	15, 21, 68, 79
C15	"	—	16, 21, 69, 79
C16	"	—	17, 21, 70, 79
C17	"	—	18, 21, 71, 79
C18	"	—	19, 21, 72, 79
CFTH	Thrust coefficient	—	CALL, 53, 116
CSTH	Characteristic velocity	ft/sec	CALL, 21, 79
E2	—	—	114, 116
E12	—	—	112, 116
E32	—	—	113, 116
E52	—	—	115, 116
EP	Expansion area ratio	—	CALL, 49, 53, 112, 113, 114, 115, 116

SUBROUTINE TCPROP NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
EPRT	—	—	49, 53
ERC	—	—	48, 50, 51, 53
ERC2	—	—	50, 53
ERC3	—	—	51, 53
FLNMU	—	—	108, 116
FLNP	—	—	111, 116
FMU	—	—	74, 79, 116
FMU2	—	—	77, 79
FMU12	—	—	75, 79
FMU32	—	—	76, 79
FMU52	—	—	78, 79
G	—	—	109, 110, 116
G2	—	—	110, 116
INDP	Propellant combination indicator	—	CALL, 1
LNPC	—	—	REAL, 20, 21, 53, 73, 79, 111
LNRC	—	—	REAL, 52, 53
PC	Chamber pressure	psia	CALL, 20, 73, 116
XMRC	Chamber mixture ratio	—	CALL, 21, 48, 52, 74, 75, 76, 77, 78, 108, 109

```
$IBFTC TCPROP  REF,LIST
      SUBROUTINE TCPROP (INDP,XMRC,PC, EP ,CSTH,CFTH )
      REAL LNPC,LNRC
      GO TO (10,20),INDP
C         LOX-RP1 CONSTANTS
   10 C1=-8244.2601
      C2=16684.784
      C3=-6172.4708
      C4=2733.3791
      C5=-547.23793
      C6=42.785606
      C7=-3899.8009
      C8=5576.0354
      C9=-2738.1352
      C10=578.74455
      C11=-36.201772
      C12=-1.664795
      C13=700.30026
      C14=-1245.0609
      C15=830.64172
      C16=-264.34673
      C17=40.430077
      C18=-2.3862895
      LNPC=ALOG(PC)
      CSTH = ( C1 + C2 * XMRC +C3 * XMRC**2 +C4 *XMRC**3 +C5 *XMRC**4 +
     1 C6* XMRC**5 +C7 * LNPC +C8* XMRC*LNPC +C9 *XMRC**2 *LNPC +C10 *XM
     2RC **3 * LNPC +C11* XMRC**4 *LNPC+C12 *XMRC**5 *LNPC +C13* LNPC**2
     3+C14*XMRC *LNPC**2 +C15* XMRC**2 *LNPC **2 +C16* XMRC**3 *LNPC**2
     4+C17*XMRC**4* LNPC**2 +C18*XMRC**5* LNPC**2  )/(XMRC **1.5)
      A1=1.7668722
      A2=4.60582E-01
      A3=-1.6518414E-01
      A4=-6.6045816E-02
      A5=3.6782834E-02
      A6=-7.417031E-01
      A7=2.3071612
      A8=-5.6948143
      A9=6.4492257
      A10=-3.000598
      A11=-1.1999626
      A12=5.8599377E-01
      A13=-4.327889E-02
      A14=-1.165337E-01
      A15=5.3700486E-01
      A16=1.0577987E-02
      A17=.99303665
      A18=-7.8752103E-04
      A19=1.9191969E-03
      A20=-3.5133906E-03
      A21=1.1578578E-04
```

FIGURE 94. SYMBOLIC LISTING OF SUBROUTINE TCPROP

```
      A22=3.6707999E-05
      A23=4.3197244E-04
      A24=4.1092356E-06
      A25=-5.2376584E-05
      A26=1.4744636E-05
      ERC= 1.0-EXP(-3.*XMRC +3.9)
      EPRT=SQRT(EP)
      ERC2= ERC**2
      ERC3= ERC**3
      LNRC= ALOG (XMRC)
      CFTH= (A1 +A2* LNRC +A3 *LNRC**2 +A4*LNRC**3 + A5*LNRC**4 + A6/EPR
     1T +A7/EP   +A8/(EP**1.5)  +A9/(EP**2) + A10/(EP**2.5) +A11*LNRC/
     2EPRT + A12 * LNRC/EP  +A13 * LNRC**3 /EPRT  + A14 *LNRC**3/EP + A1
     35 *LNRC **2/EPRT )* A17 +A16 +A19*ERC2 *EPRT *LNPC +A20*ERC3 *EPRT
     4*LNPC +A21*ERC *EPRT* LNPC**2 +A22 *ERC2 * EPRT *LNPC **2 + A23*
     5ERC3 *EPRT *LNPC**2 +A24 *ERC *EPRT*LNPC**3  + A25* ERC2*EPRT*LNPC
     6**3 +A26 * ERC3 *EPRT* LNPC**3+A18*ERC*EPRT*LNPC
      GO TO 60
C       LOX-LH2 CONSTANTS
   20 C1=-1095.6897E+04
      C2=1177.5202E+05
      C3=-4888.403E+05
      C4=9945.6205E+05
      C5=-9962.3336E+05
      C6=3935.112E+05
      C7=4608.3401E+03
      C8=-4698.5585E+04
      C9=1902.431E+05
      C10=-3817.931E+05
      C11=3792.6895E+05
      C12=-1489.9681E+05
      C13=-4000.8759E+02
      C14=4096.6833E+03
      C15=-1663.2141E+04
      C16=3343.4745E+04
      C17=-3324.822E+04
      C18=1306.9946E+04
      LNPC= ALOG (PC)
      FMU=XMRC
      FMU12= SQRT(XMRC)
      FMU32= XMRC**1.5
      FMU2= XMRC**2
      FMU52= XMRC**2.5
      CSTH = (C1+C2/FMU12 +C3/FMU +C4/ FMU32  +C5/FMU2 +C6/FMU52 + LNPC
     1*(C7+C8/FMU12 +C9 /FMU +C10/FMU32+ C11/FMU2 +C12/FMU52) +LNPC**2*
     2(C13 +C14/FMU12 +C15/FMU + C16/FMU32+ C17/FMU2 + C18/FMU52))/FMU32
      A1=1.5283438
      A2=1.0474001
      A3=-1.1688996
      A4=.6142599
```

FIGURE 94. SYMBOLIC LISTING OF SUBROUTINE TCPROP (cont.)

```
      A5=-.104119
      A6=-.70660583
      A7=.6324424
      A8=.21869441
      A9=-1.9625851
      A10=1.4593124
      A11=-.141305
      A12=-6.6680344E-02
      A13=-.12259043
      A14=7.9118236E-02
      A15=9.2647573E-02
      A16=-5.7726734E-03
      A17=1.0038964
      A18=-129.79613
      A19=266.52457
      A20=-136.73502
      A21= 44.700008
      A22=-91.783763
      A23=47.085807
      A24=-3.7654116
      A25=7.7314006
      A26=-3.9661493
      A27=1.8912772E-04
      A28=2.1775238E-04
      FLNMU= ALOG(XMRC)
      G=1.0 -EXP(-3.0* XMRC +3.0)
      G2=G**2
      FLNP= LNPC
      E12=SQRT(EP)
      E32=EP**1.5
      E2=EP*EP
      E52=EP**2.5
      CFTH= (A1 +A2 * FLNMU +A3 *FLNMU**2 +A4*FLNMU**3 +A5*FLNMU**4 +A6/
     1E12 +A7/EP+A8/E32 +A9/E2 +A10/E52+(A11 *FLNMU)/E12 +(A12*FLNMU)/EP
     2+(A13* FLNMU **3)/E12 +(A14*FLNMU**3)/EP+( A15*FLNMU**2)/E12)*A17+
     3A16+ G* FLNP *E12 *(A18+A19 *G + A20 *G2) +G*FLNP**2*E12 *(A21+A22
     4*G +A23*G2)+G*FLNP**3 *E12 *(A24+A25*G +A26*G2) +FMU *EXP(-PC/243.
     50) *(A27 +A28 *FMU *E12 )
   60 RETURN
      END
```

FIGURE 94. SYMBOLIC LISTING OF SUBROUTINE TCPROP (cont.)

LXXV. SUBROUTINE PRPDEN

This subroutine calculates the propellant density and vapor pressure, given the static pressure and temperature. The propellants considered are liquid oxygen, RP1, and liquid hydrogen.

The computational method employed includes the evaluation of a number of polynomial equations. The coefficients of these equations are internally stored in a DATA statement in arrays CLIST1 through CLIST9. Evaluation of the polynomials is accomplished with Subroutine VPOLY.

CALLING SEQUENCE

The calling sequence is:

CALL PRPDEN (K, P, T, ARGA, ARGB, PV, DENS)

SOLUTION METHOD

The following initial data are defined:

1. PSL = 14.696
2. TR = T + 459.688
3. TK = (TR)/(1.8)

The appropriate set of equations is established:

```
4.  If K = 1, go to 5
    If K = 2, go to 11
    If K = 3, go to 18
```

Liquid Oxygen Calculations

```
5.  CALL VPOLY(3, CLIST1, (1.0)/(TK), TEMP1)
6.  PV = (PSL) exp(TEMP1)
7.  CALL VPOLY(3, CLIST2, TK, TEMP1)
8.  CALL VPOLY(3, CLIST3, TK, TEMP2)
9.  TEMP2 = (62.428227)(TEMP2)
10. DENS = $(TEMP2)(1.0 + (10.0)(TEMP1)(P - PV))^{0.1}$
    Return
```

RP1 Calculations

```
11. CALL VPOLY(4, CLIST4, TR, TEMP1)
12. TEMP1 = TEMP1 +(146230.05)/(TR)+(776.60607)(Ln(TR))
13. PV = exp(TEMP1)
14. CALL VPOLY(3, CLIST5, T, TEMP1)
15. TEMP1 = $TEMP1 - (1.1786)(10)^{-10}(P)$
16. TEMP2 = (0.000394)(ARGA - T) + (0.9991)(ARGB)
17. DENS = (62.4266)(TEMP2)(1.0 + (TEMP1) (P))
    Return
```

Liquid Hydrogen Calculations

```
18.  CALL VPOLY(6, CLIST6, (200.0)/(TK), TEMP1)
19.  PV = (PSL) exp ((2.30258)(TEMP1))
20.  CALL VPOLY(4, CLIST7, TK, TEMP2)
21.  CALL VPOLY(4, CLIST8, TK, TEMP3)
22.  CALL VPOLY(5, CLIST9, $(32.976 - TK)^{1/3}$, TEMP4)
23.  TEMP5 = $(TEMP3 + (P)/(PSL))/(TEMP3 + (10.0)^{TEMP1})$
24.  TEMP6 = (1.0 -(TEMP2) Ln(TEMP5))/(TEMP4)
25.  DENS = (125.83517)/(TEMP6)
     Return
```

SUBROUTINE PRPDEN NOMENCLATURE

Symbol	Description	Units	Reference
ARGA	Reference temperature, RP1 option only	°F	CALL, 16
ARGB	Specific gravity at reference temperature, RP1 option only	—	CALL, 16
CLIST1	Polynomial coefficients	—	DIM, DATA, 5
CLIST2	Polynomial coefficients	—	DIM, DATA, 7
CLIST3	Polynomial coefficients	—	DIM, DATA, 8
CLIST4	Polynomial coefficients	—	DIM, DATA, 11
CLIST5	Polynomial coefficients	—	DIM, DATA, 14
CLIST6	Polynomial coefficients	—	DIM, DATA, 18
CLIST7	Polynomial coefficients	—	DIM, DATA, 20
CLIST8	Polynomial coefficients	—	DIM, DATA, 21
CLIST9	Polynomial coefficients	—	DIM, DATA, 22
DENS	Density	lb/ft^3	CALL, 10, 17, 25
K	Propellant type indicator	—	CALL, 4
	K = 1, Liquid oxygen	—	
	K = 2, RP1	—	
	K = 3, Liquid hydrogen	—	
P	Static pressure	lb/in^2	CALL, 10, 15, 17, 23
PSL	Standard sea level atmospheric pressure	lb/in^2	1, 6, 19, 23

SUBROUTINE PRPDEN NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PV	Vapor pressure	lb/in^2	CALL, 6, 10, 13, 19
T	Static temperature	°F	CALL, 2, 14, 16
TEMP1	Temporary storage variables	—	5, 6, 7, 10, 11, 12, 13, 14, 15, 17, 18, 19, 23
TEMP2	Temporary storage variables	—	8, 9, 10, 16, 17, 20, 24
TEMP3	Temporary storage variables	—	21, 23
TEMP4	Temporary storage variables	—	22, 24
TEMP5	Temporary storage variables	—	23, 24
TEMP6	Temporary storage variables	—	24, 25
TK	Temperature, Kelvin	°K	3, 5, 7, 8, 18, 20, 21, 22
TR	Temperature, Rankine	°R	2, 3, 11, 12

```
$IBFTC PRPDEN  FULIST,REF
      SUBROUTINE PRPDEN(K,P,T,ARGA,ARGB,PV,DENS)
C                K = PROPELLANT TYPE
C                    1 = LIQUID OXYGEN
C                    2 = RP1
C                    3 = LIQUID HYDROGEN
C                P = STATIC PRESSURE, LB/IN**2
C                T = STATIC TEMPERATURE, DEGREES FAHRENHEIT
C                ARGA = REFERENCE TEMPERATURE, RP1 OPTION ONLY
C                ARGB = SPECIFIC GRAVITY AT REF. TEMP., RP1 OPTION ONLY
C                PV = VAPOR PRESSURE, LB/IN**2
C                DENS = DENSITY, LB/FT**3
C           PRPDEN CALCULATES PROPELLANT DENSITY AND VAPOR PRESSURE FROM
C                PRESSURE AND TEMPERATURE FOR LOX, RP1, LH2
      DIMENSION  CLIST1(3),CLIST2(3),CLIST3(3),CLIST4(4),CLIST5(3),
     $           CLIST6(6),CLIST7(4),CLIST8(4),CLIST9(5)
      DATA  (CLIST1(I),I=1,3)/5.238279,-7.29534181,-41958.931/,
     $    (CLIST2(I),I=1,3)/8.05790431E-5,-1.8993851E-6,1.2860036E-8/,
     $    (CLIST3(I),I=1,3)/1.414202,-1.033016E-3,-2.23E-5/,
     $    (CLIST4(I),I=1,4)/-4536.4149,-1.4305354,5.7028733E-4,
     $             -1.0132155E-7/,
     $    (CLIST5(I),I=1,3)/3.9389E-6,1.9325E-8,-8.75E-12/,
     $    (CLIST6(I),I=1,6)/4.9259175,-1.1789812,.14833835,-.012457985,
     $             .53989697E-3,-.9468439E-5/,
     $    (CLIST7(I),I=1,4)/.2601774,-.020291145,.94109544E-3,
     $             -.14947327E-4/,
     $    (CLIST8(I),I=1,4)/183.81304,-4.7166549,-.14768532,
     $             .33075926E-2/,
     $    (CLIST9(I),I=1,5)/.01526718,.62675344E-2,.14973511E-2,
     $             -.18306903E-3,-.2069318E-4/
      DATA  PSL/14.696/
      TR = T + 459.688
      TK = TR/1.8
      GO TO (10,20,30), K
C          LIQUID OXYGEN CALCULATIONS
   10 CALL VPOLY(3,CLIST1,1.0/TK,TEMP1)
      PV = PSL * EXP(TEMP1)
      CALL VPOLY(3,CLIST2,TK,TEMP1)
      CALL VPOLY(3,CLIST3,TK,TEMP2)
      TEMP2 = 62.428227 * TEMP2
      DENS = TEMP2 * (1.0 + 10.0 * TEMP1 * (P - PV))**0.1
      RETURN
C          RP1 CALCULATIONS
   20 CALL VPOLY(4,CLIST4,TR,TEMP1)
      TEMP1 = TEMP1 + 146230.05/TR + 776.60607 * ALOG(TR)
      PV = EXP(TEMP1)
      CALL VPOLY(3,CLIST5,T,TEMP1)
      TEMP1 = TEMP1 - 1.1786E-10 * P
      TEMP2 = .000394 * (ARGA - T) + .9991 * ARGB
      DENS = 62.4266 * TEMP2 * (1.0 + TEMP1 * P)
```

FIGURE 95. SYMBOLIC LISTING OF SUBROUTINE PRPDEN

```
      RETURN
C          LIQUID HYDROGEN CALCULATIONS
   30 CALL VPOLY(6,CLIST6,200.0/TK,TEMP1)
      PV = PSL * EXP(2.30258 * TEMP1)
      CALL VPOLY(4,CLIST7,TK,TEMP2)
      CALL VPOLY(4,CLIST8,TK,TEMP3)
      CALL VPOLY(5,CLIST9,(32.976 - TK)**.33333333,TEMP4)
      TEMP5 = (TEMP3 + P/PSL)/(TEMP3 + 10.0**TEMP1)
      TEMP6 = (1.0 - TEMP2 * ALOG(TEMP5))/TEMP4
      DENS = 125.83517/TEMP6
      RETURN
      END
```

FIGURE 95. SYMBOLIC LISTING OF SUBROUTINE PRPDEN (cont.)

LXXVI. SUBROUTINE CHMBER

This subroutine calculates the chamber performance of an RL10-A1 engine.

The following input data is required:

a. Injector end chamber pressure

b. Mixture ratio

c. Thrust chamber throat area

d. Oxidizer mass flow rate

e. Fuel mass flow rate

f. Desired thrust

g. Thrust chamber expansion area ratio

Subroutine CHMBER uses COMMON blocks CCHAM, CMAIN, CHRL10, CBAL, MNRL10, and CMAIN.

CALLING SEQUENCE

The subroutine calling sequence is:

```
CALL CHMBER (IND, IDLE, IDLF)
```

where:

IND = Indicator which determines the correct entry point into Subroutine LOX

IDLE = Iteration loop control counter

IDLF = Iteration loop control counter

SOLUTION METHOD

Thrust chamber nozzle stagnation pressure:

1. PNS = (PCIE)/(1.037)

Theoretical characteristic velocity and thrust coefficient:

2. CALL TCPROP (2, XMRTC, PNS, EPS, CSTHE, CFTHE)

Characteristic velocity efficiency:

3. If XMRTC - 4.3 $\leq$ 0, go to 4

   If XMRTC - 4.3 $>$ 0, go to 6

4. ETCS = .95152

5. Go to 7

6. ETCS = 1.01 - (XMRTC)(.0136)

Actual characteristic velocity:

7. CS = (CSTHE)(ETCS)

Total chamber mass flow rate:

8. WC = (32.174)(PNS)(ATC)/(CS)

Test to determine if total chamber flow rate matches the sum of the flow rates obtained by balancing the LOX and fuel systems:

9. If $|WC - WCO - WCF| - ((TOL8)(WC)) \leq 0$, go to 13
   If $|WC - WCO - WCF| - ((TOL8)(WC)) > 0$, go to 10

New value of fuel pump speed:

10. CALL ESTM (IDLE, 2, TXNP, TTXNP, (WC - WCF - WCO), XNP(1), LUP8)
11. IND = 3
12. Go to 25
13. IDLE = 0

Thrust coefficient efficiency:

14. ETCF = .9996 - (XMRTC)(.0056)

Actual thrust coefficient:

15. CF = (ETCF)(CFTHE)

Thrust:

16. FC = (CF)(PNS)(ATC)

17. If KBAL - 1 $\leq$ 0, go to 18

If KBAL - 1 $>$ 0, go to 22

Test to determine if calculated thrust matches the loaded thrust:

18. If $|FC - FCL| - ((TOL9)(FC)) \leq 0$, go to 22

If $|FC - FCL| - ((TOL9)(FC)) > 0$, go to 19

New value of desired injector end chamber pressure:

19. CALL ESTM (IDLF, 2, TPCIE, TTPCIE, FC - FCL, PCIEL, LUP9)

20. IND = 2

21. Go to 25

22. IDLF = 0

Specific impulse:

23. XISPE = (FC)/(WC)

24. IND = 0

25. Return

SUBROUTINE CHMBER NOMENCLATURE

Symbol	Description	Units	Reference
ATC	Thrust chamber throat area	in^2	/CCHAM/, 8, 16
CF	Thrust chamber thrust coefficient	—	/CCHAM/, 15, 16
CFTHE	Thrust chamber theoretical thrust coefficient	—	/CCHAM/, 2, 15
CS	Thrust chamber characteristic velocity	ft/sec	/CCHAM/, 7, 8
CSTHE	Thrust chamber theoretical characteristic velocity	ft/sec	/CCHAM/, 2, 7
EPS	Thrust chamber expansion area ratio	—	/CCHAM/, 2
ETCF	Thrust chamber nozzle efficiency	—	/CCHAM/, 14, 15
ETCS	Thrust chamber combustion efficiency	—	/CCHAM/, 4, 6, 7
FC	Thrust chamber thrust	lbf	/CCHAM/, 16, 18, 19, 23
FCL	Loaded value of thrust	lbf	/CHRL10/, 18, 19
IDLE	Iteration loop control counter	—	CALL, 10, 13
IDLF	Iteration loop control counter	—	CALL, 19, 22
IND	Indicator to determine correct entry point into Subroutine LOX	—	CALL, 11, 20, 24
KBAL	Balance/rebalance indicator	—	/CBAL/, 17

SUBROUTINE CHMBER NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
LUP8	—	—	DATA, 10
LUP9	—	—	DATA, 19
PCIE	Thrust chamber injector end pressure	lbf/in^2	/CMAIN/, 1
PCIEL	Loaded value of thrust chamber injector end chamber pressure	lbf/in^2	/MNRL10/, 19
PNS	Thrust chamber nozzle stagnation pressure	lbf/in^2	/CCHAM/, 1, 2, 8, 16
TOL8	Chamber flow rate tolerance	—	/CHRL10/, 9
TOL9	Thrust tolerance	—	/CHRL10/, 18
TPCIE	Storage array for independent argument in Subroutine ESTM	—	DIM, 19
TTPCIE	Storage array for dependent argument in Subroutine ESTM	—	DIM, 19
TTXNP	Storage array for dependent argument in Subroutine ESTM	—	DIM, 10
TXNP	Storage array for independent argument in Subroutine ESTM	—	DIM, 10
WC	Thrust chamber total flow rate	lbm/sec	/CCHAM/, 8, 9, 10, 23

SUBROUTINE CHMBER NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WCF	Thrust chamber fuel flow rate	lbm/sec	/CCHAM/, 9, 10
WCO	Thrust chamber oxidizer flow rate	lbm/sec	/CCHAM/, 9, 10
XISPE	Engine specific impulse	$\frac{\text{lbf-sec}}{\text{lbm}}$	/CMAIN/, 23
XMRTC	Thrust chamber mixture ratio	—	/CCHAM/, 2, 3, 6, 14
XNP	Pump speed	rpm	/CPUMP/, 10

```
$IBFTC CHMBER  FULIST,REF
      SUBRØUTINE CHMBER(IND,IDLE,IDLF)
      DIMENSIØN TXNP(20),TTXNP(20),TPCIE(20), TTPCIE(20)
      DATA LUP8/6HXNP(1)/ ,LUP9/6H PCIEL /
      CØMMØN /MNRL10/ PCIEL,Q(2),XMRTCL,PII(2)
      CØMMØN /CHRL10/  TØL8,TØL9,FCL
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),     PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
      CØMMØN /CCHAM/  INDPR ,CF ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCØ
      CØMMØN /CBAL/KBAL /CPA/PA /CRESGG/RESØG,RESFG
      CØMMØN /CMAIN/WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),TN,
     1ØRES(4),WG,WGØ,WGF,PCIE,XMRE,PGG,XISPC,XISPE,FE,WEØ,WEF,TPIR(2)
C        THRUST CHAMBER EQUATIØNS
      PNS = PCIE / 1.037
      CALL TCPRØP(2,XMRTC,PNS,EPS,CSTHE,CFTHE   )
      IF(XMRTC-4.3)301,301,302
  301 ETCS = .95152
      GØ TØ 303
  302 ETCS = 1.01 - XMRTC* .0136
  303 CS = CSTHE * ETCS
      WC = 32.174 * PNS * ATC / CS
      IF(ABS(WC-WCØ-WCF) -(TØL8*WC  ))305,305,304
  304 CALL ESTM(IDLE,2,TXNP,TTXNP,WC-WCF-WCØ,XNP(1),LUP8 )
      IND = 3
      GØ TØ 310
  305 IDLE = 0
      ETCF = .9996 - XMRTC * .0056
      CF = ETCF * CFTHE
      FC = CF * PNS * ATC
      IF(KBAL-1)306,306,308
  306 IF(ABS(FC-FCL) -(TØL9*FC))308,308,307
  307 CALL ESTM(IDLF,2,TPCIE,TTPCIE,FC-FCL,PCIEL,LUP9     )
      IND = 2
      GØ TØ 310
  308 IDLF = 0
      XISPE = FC / WC
      IND =0
  310 RETURN
      END
```

FIGURE 96. SYMBOLIC LISTING OF SUBROUTINE CHMBER

LXXVII. SUBROUTINE PUMP1

This subroutine calculates the fuel and oxidizer pump performance parameters of an RL10-A1 engine. The following input data is required:

a. Actual pump speed

b. Nominal pump speed

c. Pump inlet volume flow rate

d. Pump head

e. Pump inlet conditions

This subroutine uses COMMON blocks CPUMP, PPRL10, and MNRL10.

The subroutine calling sequence is:

CALL PUMP1 (IP)

where:

IP = Indicator denoting fuel or oxidizer pump. Must be set equal to: 1 (fuel pump), or 2 (oxidizer pump).

SOLUTION METHOD

Set LOX-Fuel indicator:

1. IN = IP

Ratio of actual pump speed to nominal pump speed:

2. XRATIO(IN) = (XNP(IN))/(XNPN(IN))

Dimensionless flow coefficient:

3. PHI(IN) = ((Q(IN))/(XNP(IN)))(PCON(IN))

Set ESTM iteration loop control counter:

4. IDL = 0

Do LOX or fuel calculations according to indicator:

5. If IN = 1, go to 25

   If IN = 2, go to 6

LOX Side Calculations

Dimensionless head coefficient:

6. CALL UVAR (PHI(IN), 3, SI(IN))

7. SI(IN) = (SI(IN))(1.01168)

Head:

8. H (IN) = (SI (IN)) (XNP (IN))^{2}/(115600)

Discharge total pressure:

9. PPD (IN) = PPI (IN) + ((H (IN)) (RHPI (IN)))/(144)

Volumetric efficiency:

10. CALL UVAR((XNP (IN)) (.001), 4, ETAPV)

Thermal efficiency:

11. CALL UVAR((Q (IN)/(XRATIO (IN)), 5, ETAPT)

Efficiency:

12. ETAP (IN) = (ETAPV) (ETAPT)

Inlet constant pressure specific heat:

13. CALL UVAR(TPIS (IN), 6, CPI)

Discharge constant pressure specific heat:

14. CALL UVAR(TPIS (IN) + 2, 6, CPD)

Average of inlet and discharge constant pressure specific heats:

15. CPAVE(IN) = (CPI + CPD)/(2)

Temperature rise across pump:

```
16.  TDEL(IN) = (H(IN))/((778.26)(CPAVE(IN)))
```

Discharge static temperature:

```
17.  TPDS = TPIS(IN) + TDEL(IN)
```

Continue with LOX for fuel calculations according to the indicator:

```
18.  If IN = 1,  go to 48
     If IN = 2,  go to 19
```

Discharge constant pressure specific heat:

```
19.  CALL UVAR (TPDS, 6, CPDD)
20.  If | CPD - CPDD | ≤ ((TOL2)(CPD)),  go to 23
     If | CPD - CPDD | > ((TOL2)(CPD)),  go to 21
```

New value of discharge constant pressure specific heat:

```
21.  CALL ESTM (IDL, 2, FCPD, DFCPD, (CPD - CPDD),
          CPD, LUP2)
22.  Go to 15
```

Discharge density:

```
23.  CALL UVAR (TPDS, 2, RHPD(IN))
24.  Return
```

Fuel Side Calculations

25. $XX = ((-.0002)(XNP(IN))) + 6.$

26. If $PHI(IN) \leq .044$, go to 27

If $PHI(IN) > .044$, go to 29

Dimensionless head coefficient:

27. $SI(IN) = ((-1.847)(PHI(IN)) + ((.009)(XX)) + .66755$

28. Go to 30

29. $SI(IN) = ((-4.8611)(PHI(IN))) + ((.009)(XX)) + .80017$

Head:

30. $H(IN) = ((SI(IN))(XNP(IN))^2)/16780)$

Discharge total pressure:

31. $PPD(IN) = PPI(IN) + (((H(IN))(RHPI(IN)))/(144))$

Volumetric efficiency:

32. $ETAPV = ((.00005534)(XNP(IN))) - .6602$

Thermal efficiency:

33. $ETAPT = ((-.0000032)(XNP(IN))) + .096$

34. $XXX = ((.00002298)(XNP(IN))) + .3106$

Efficiency:

35. $ETAP(IN) = -((1 + ((448.86)(Q(IN)) - (670)(XXX))^2 / ((ETAPV)(26267.82)))(.007056))^{1/2} + ETAPT + .63$

Test for correct curve to be used in BIVAR:

36. If TPIS(IN) $\leq$ 60, go to 37

If TPIS(IN) $>$ 60, go to 39

37. N = 8

38. Go to 40

39. N = 9

Inlet constant pressure specific heat:

40. CALL BIVAR(TPIS(IN), (PPI(IN))/(14.7), N, CPI)

Discharge static temperature:

41. TPDS = TPIS(IN) + 15

Test for correct curve to be used in BIVAR:

42. If TPDS $\leq$ 60, go to 43

If TPDS $>$ 60, go to 45

43. N = 8

44. Go to 46

45. N = 9

Discharge constant pressure specific heat:

```
46.  CALL BIVAR (TPDS, (PPD(IN))/(14.7), N, CPD)
47.  Go to 15
```

Test for correct curve to be used in BIVAR:

```
48.  If TPDS ≤ 60, go to 49
     If TPDS > 60, go to 51
49.  N = 8
50.  Go to 52
51.  N = 9
```

Discharge constant pressure specific heat:

```
52.  CALL BIVAR (TPDS, (PPD(IN))/(14.7), N, CPDD)
53.  If | CPD - CPDD | ≤ ((TOL1)(CPD)), go to 56
     If | CPD - CPDD | > ((TOL1)(CPD)), go to 54
```

New value of discharge constant pressure specific heat:

```
54.  CALL ESTM (IDL, 2, FCPD, DFCPD, CPD - CPDD,
          CPD, LUP1)
55.  Go to 15
```

56. XX = Ln(PPD(IN))

Discharge density:

57. CALL BIVAR (XX, TPDS, 1, RHPD(IN))

58. Go to 24

SUBROUTINE PUMP1 NOMENCLATURE

Symbol	Description	Units	Reference
CPAVE	Average of pump inlet and discharge constant pressure specific heats	$\frac{btu}{lbm - {}^oR}$	DIM, 15, 16
CPD	Pump discharge constant pressure specific heat	$\frac{btu}{lbm - {}^oR}$	14, 15, 20, 21, 46, 53, 54
CPDD	Pump discharge constant pressure specific heat	$\frac{btu}{lbm - {}^oR}$	19, 20, 21, 52, 53, 54
CPI	Pump inlet constant pressure specific heat	$\frac{btu}{lbm - {}^oR}$	13, 15, 40
DFCPD	Storage array for dependent argument in Subroutine ESTM	—	DIM, 21, 54
ETAP	Pump efficiency	—	/CPUMP/, 12, 35
ETAPT	Pump thermal efficiency	—	11, 12, 33, 35
ETAPV	Pump volumetric efficiency	—	10, 12, 32, 35
FCPD	Storage array for independent argument in Subroutine ESTM	—	DIM, 21, 54
H	Pump head	$\frac{ft - lbf}{lbm}$	/CPUMP/, 8, 9, 16, 30, 31
IDL	Iteration loop control counter	—	4, 21, 54

SUBROUTINE PUMP1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
IN	Indicator to determine fuel side or LOX side equations	—	1, 2, 3, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 23, 25, 26, 27, 29, 30, 31, 32, 33, 34, 35, 36, 40, 41, 46, 52, 56, 57
IP	Indicator to determine fuel side or LOX side equations	—	CALL, 1
LUP1	—	—	DATA, 54
LUP2	—	—	DATA, 21
N	Set to correct curve number for use in Subroutine BIVAR	—	37, 39, 40, 43, 45, 46, 49, 51, 52
PCON	Constant used in calculating pump dimensionless flow coefficient	$\frac{\text{sec}}{\text{ft}^3 - \text{min}}$	DIM, DATA, 3
PHI	Dimensionless pump flow coefficient	—	3, 6, 26, 27, 29
PPD	Pump discharge total pressure	$\frac{\text{lbf}}{\text{in}^2}$	/CPUMP/, 9, 31, 46, 52, 56
PPI	Pump inlet total pressure	$\frac{\text{lbf}}{\text{in}^2}$	/CPUMP/, 9, 31, 40
Q	Pump inlet volumetric flowrate	$\frac{\text{ft}^3}{\text{sec}}$	/MNRL10/, 3, 11, 35

SUBROUTINE PUMP1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
RHPD	Pump discharge density	$\frac{lbm}{ft^3}$	/CPUMP/, 23, 57
RHPI	Pump inlet density	$\frac{lbm}{ft^3}$	/CPUMP/, 9, 31
SI	Dimensionless head coefficient	—	DIM, 6, 7, 8, 27, 29, 30
TDEL	Temperature rise across pump	^oR	DIM, 16, 17
TOL1	Fuel pump discharge constant pressure specific heat tolerance	—	/PPRL10/, 53
TOL2	Oxidizer pump discharge constant pressure specific heat tolerance	—	/PPRL10/, 20
TPDS	Pump discharge static temperature	^oR	/CPUMP/, 17, 19, 23, 41, 42, 46, 48, 52
TPIS	Pump inlet static temperature	^oR	/CPUMP/, 13, 14, 17, 36, 40, 41
XNP	Pump speed	$\frac{rev}{min}$	/CPUMP/, 2, 3, 8, 10, 25, 30, 32, 33, 34
XNPN	Nominal pump speed	$\frac{rev}{min}$	/PPRL10/, 2
XRATIO	Ratio of actual pump speed to nominal pump speed	—	/PPRL10/, 2, 11

SUBROUTINE PUMP1 NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
XX	Constant used in calculating fuel pump dimensionless head coefficient	—	25, 27, 29, 56, 57
XXX	Constant used in calculating fuel pump efficiency	—	34, 35

```
$IBFTC PMP1S    FULIST,REF
      SUBRØUTINE PUMP1(IP)
C IP = 1 FØR FUEL SIDE , 2 FØR LØX SIDE
      DIMENSIØN PHI(2)
      DIMENSIØN PCØN(2),FCPD(20),DFCPD(20),SI(2),CPAVE(2),TDEL(2)
      DATA PCØN/835.73,2872.5/,LUP2/6HCPD L /,LUP1/6HCPD F /
      CØMMØN /MNRL10/ PCIEL,Q(2),XMRTCL,PII(2)
      CØMMØN /LXRL10/ TØL3,TØL5,CD2,CØN1,CØN2,CØN3,CØN4,CØN5
      CØMMØN /FLRL10/ CØN8,CØN6,TØL4,FLØPT,CØN7,TØJE,WCF1,TØL6,TØL7,
     1CØN10,AVT
      CØMMØN /CHRL10/  TØL8,TØL9,FCL
      CØMMØN /PPRL10/ XRATIØ(2), TØL1, TØL2, XNPN(2)
      CØMMØN /TBRL10/ CØN9,ETGER,CØN11,CØN12,CØN13,CØN14,CØN15,TØL10
     1,HP1,HP2
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),     PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
      CØMMØN /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
      CØMMØN /CCHAM/  INDPR ,CF  ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCØ
      CØMMØN /CMAIN/WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),TN,
     1ØRES(4),WG,WGØ,WGF,PCIE,XMRE,PGG,XISPC,XISPE,FE,WEØ,WEF,TPIR(2)
      IN = IP
      XRATIØ(IN) = XNP(IN)/XNPN(IN)
      PHI(IN) =  (Q(IN)/XNP(IN) ) * PCØN(IN)
      IDL =0
      GØ TØ (2,200),IN
C LØX SIDE EQUATIØNS FØLLØW
  200 CALL UVAR(PHI(IN),3,SI(IN) )
      SI(IN) =  SI(IN) * 1.01168
      H(IN) = SI(IN) * XNP(IN)**2/115600.
      PPD(IN) = PPI(IN) +(H(IN)*RHPI(IN)/144.)
      CALL UVAR(XNP(IN)*.001,4,ETAPV   )
      CALL UVAR (Q(IN)/XRATIØ(IN),5,ETAPT)
      ETAP(IN) = ETAPV*ETAPT
      CALL UVAR(TPIS(IN),6,CPI )
      CALL UVAR(TPIS(IN)+2.,6,CPD )
  210 CPAVE(IN) = (CPI + CPD)/2.0
      TDEL(IN) = H(IN)/(778.26*CPAVE(IN)      )
      TPDS = TPIS(IN) + TDEL(IN)
      GØ TØ (26,220),IN
  220 CALL UVAR(TPDS,6,CPDD )
      IF(ABS(CPD-CPDD) -(TØL2*CPD))240,240,225
  225 CALL ESTM(IDL,2,FCPD,DFCPD,CPD-CPDD,CPD,LUP2)
      GØ TØ 210
  240 CALL UVAR(TPDS,2,RHPD(IN) )
```

FIGURE 97. SYMBOLIC LISTING OF SUBROUTINE PUMP1

```
 1000 RETURN
C FUEL SIDE EQUATIØNS FØLLØW
    2 XX =(-.0002*XNP(IN)) + 6.
      IF(PHI(IN)-.044)21,21,22
   21 SI(IN) = (-1.847*PHI(IN)) + (.009*XX) + .66755
      GØ TØ 23
   22 SI(IN) = (-4.8611*PHI(IN)) + (.009*XX) + .80017
   23 H(IN) = (SI(IN)*XNP(IN)**2)/ 16780.
      PPD(IN) = PPI(IN) + (H(IN)*RHPI(IN)/144.     )
      ETAPV = (.00005534*XNP(IN)) -.6602
      ETAPT = (-.0000032*XNP(IN)) + .096
      XXX   = (.00002298*XNP(IN)) +.3106
      ETAP(IN)=-SQRT((1.+(448.86*Q(IN)-670.*XXX)**2/(ETAPV*26267.82)) *
     1.007056) + ETAPT +.63
      IF(TPIS(IN)-60.)24,24,25
   24 N = 8
      GØ TØ 260
   25 N = 9
 260  CALL BIVAR(TPIS(IN),PPI(IN)/14.7,N,CPI)
      TPDS = TPIS(IN)+15.
      IF(TPDS-60.)27,27,28
   27 N = 8
      GØ TØ 29
   28 N = 9
   29 CALL BIVAR(TPDS,PPD(IN)/14.7,N,CPD )
      GØ TØ 210
   26 IF(TPDS -60.)30,30,31
   30 N = 8
      GØ TØ 33
   31 N = 9
   33 CALL BIVAR(TPDS,PPD(IN)/14.7,N,CPDD)
      IF(ABS(CPD-CPDD) - (TØL1*CPD))35,35,32
   32 CALL ESTM(IDL,2,FCPD,DFCPD,CPD-CPDD,CPD,LUP1)
      GØ TØ 210
   35 XX = ALØG(PPD(IN))
      CALL BIVAR(XX,TPDS,1,RHPD(IN) )
      GØ TØ 1000
      END
```

FIGURE 97. SYMBOLIC LISTING OF SUBROUTINE PUMP1 (cont.)

LXXVIII. SUBROUTINE LOX

This subroutine calculates the liquid oxygen mass flow rate of an RL10-A1 engine. The following input data is required:

a. Chamber pressure

b. LOX pump inlet conditions

c. Flow loss coefficient of each component

This subroutine uses COMMON blocks CPUMP, LXRL10, CCHAM, MNRL10 and CMAIN.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL LOX (IND)

where:

IND = Indicator designating the correct entry point to the subroutine

SOLUTION METHOD

Reset counters for Subroutine ESTM:

1. IDL = 0

2. IDLA = 0

Test for correct entry to subroutine:

3. If IND = 0, go to 4

   If IND = 1, go to 9

   If IND = 2, go to 17

LOX pump speed:

4. XNP(2) = (CON5)(XNP(1))

LOX mass flow rate:

5. WCO = (Q(2))(RHPI(2))

LOX pump inlet static pressure:

6. $PPIS(2) = PPI(2) - ((CON1)(WCO)^2/RHPI(2))$

Calculate LOX pump performance:

7. CALL PUMP1(2)

Pressure drops from LOX pump outlet to injector inlet:

8. $\mathrm{PDEL1} = (\mathrm{CON2})(\mathrm{WCO})^2/(\mathrm{RHPD}(2))$

9. $\mathrm{PDEL2} = (\mathrm{CON3})(\mathrm{WCO})^2/(\mathrm{RHPD}(2))$

LOX injector inlet pressure:

10. $\mathrm{PII}(2) = \mathrm{PPD}(2) - \mathrm{PDEL1} - \mathrm{PDEL2}$

Pressure drop across LOX injector:

11. $\mathrm{PDEL3} = (144)(\mathrm{WCO})^2/((2)(\mathrm{RHPD}(2))(32.174)((\mathrm{CD2})(\mathrm{CON4}))^2)$

12. CALL UVAR (PDEL3 + .06, 7, CDD2)

13. If $|\mathrm{CD2} - \mathrm{CDD2}| - ((\mathrm{TOL3})(\mathrm{CD2})) \leq 0$, go to 15

If $|\mathrm{CD2} - \mathrm{CDD2}| - ((\mathrm{TOL3})(\mathrm{CD2})) > 0$, go to 14

14. CALL ESTM (IDL, 2, TCD2, TCDD2, (CD2 - CDD2), CD2, LUP3)

15. Go to 10

Injector end chamber pressure:

16. $\mathrm{PCIE} = \mathrm{PII}(2) - \mathrm{PDEL3}$

Reset counter used in 14:

17. $\mathrm{IDL} = 0$

Test to determine if calculated chamber pressure matches the loaded value:

```
18. If $|PCIE - PCIEL| - ((TOL5)(PCIE)) \leq 0$, go to 21
    If $|PCIE - PCIEL| - ((TOL5)(PCIE)) > 0$, go to 19
```

New value of LOX pump inlet volume flow rate:

```
19. CALL ESTM (IDLA, 2, TQ, TQY, (PCIE - PCIEL),
                   Q(2), LUP5)
20. Go to 15
21. Return
```

SUBROUTINE LOX NOMENCLATURE

Symbol	Description	Units	Reference
CD2	LOX injector flow loss coefficient	—	/LXRL10/, 11, 13, 14
CDD2	LOX injector flow loss coefficient	—	12, 13, 14
CON1	Constant used in calculating LOX pump inlet static pressure	$\frac{\text{lbf-sec}^2}{\text{lbm-ft}^3\text{-in}^2}$	/LXRL10/, 6
CON2	Flow loss coefficient for line from LOX pump discharge to injector inlet	$\frac{\text{lbf-sec}^2}{\text{lbm-ft}^3\text{-in}^2}$	/LXRL10/, 8
CON3	Ground mixture ratio control value flow loss coefficient	$\frac{\text{lbf-sec}^2}{\text{lbm-ft}^3\text{-in}^2}$	/LXRL10/, 9
CON4	Actual LOX injector area	in^2	/LXRL10/, 11
CON5	Ratio of LOX pump speed to fuel pump speed	—	/LXRL10/, 4
IDL	Iteration loop control counter	—	1, 14, 17
IDLA	Iteration loop control counter	—	2, 19
IND	Indicator to determine correct entry point into Subroutine LOX	—	CALL, 3
LUP3	—	—	DATA, 14
LUP5	—	—	DATA, 19

SUBROUTINE LOX NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PCIE	Injector end chamber pressure	lbf/in^2	/CMAIN/, 16, 18, 19
PCIEL	Loaded value of injector end chamber pressure	lbf/in^2	/MNRL10/, 18, 19
PDEL1	Pressure drop in line from LOX pump discharge to injector inlet	lbf/in^2	8, 10
PDEL2	Pressure drop across ground mixture ratio control value	lbf/in^2	9, 10
PDEL3	Pressure drop across LOX injector	lbf/in^2	11, 12, 16
PII	Injector inlet pressure	lbf/in^2	/MNRL10/, 10, 16
PPD	Pump discharge pressure	lbf/in^2	/CPUMP/, 10
PPI	Pump inlet total pressure	lbf/in^2	/CPUMP/, 6
PPIS	Pump inlet static pressure	lbf/in^2	/CPUMP/, 6
Q	Pump inlet volume flow rate	ft^3/sec	/MNRL10/, 5, 19
RHPD	Pump discharge density	lbm/ft^3	/CPUMP/, 8, 9, 11
RHPI	Pump inlet density	lbm/ft^3	/CPUMP/, 5, 6
TCD2	Storage array for independent argument in Subroutine ESTM	—	DIM, 14
TCDD2	Storage array for dependent argument in Subroutine ESTM	—	DIM, 14

SUBROUTINE LOX NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TOL3	LOX injector flow loss coefficient tolerance	—	/LXRL10/, 13
TOL5	Injector end chamber pressure tolerance	—	/LXRL10/, 18
TQ	Storage array for independent argument in Subroutine ESTM	—	DIM, 19
TQY	Storage array for dependent argument in Subroutine ESTM	—	DIM, 19
WCO	Oxidizer mass flow rate	lbm/sec	/CCHAM/, 6, 8, 9, 11
XNP	Pump speed	rpm	/CPUMP/, 4

```
$IBFTC LØX     FULIST,REF
      SUBRØUTINE LØX(IND)
      DIMENSIØN TCD2(20),TCDD2(20),TQ(20),TQY(20)
      DATA LUP3/6H CD2    /,LUP5/6H PCIE  /
      CØMMØN /MNRL10/ PCIEL,Q(2),XMRTCL,PII(2)
      CØMMØN /LXRL10/ TØL3,TØL5,CD2,CØN1,CØN2,CØN3,CØN4,CØN5
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),     PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
      CØMMØN /CCHAM/  INDPR ,CF ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCØ
      CØMMØN /CMAIN/WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),TN,
     1ØRES(4),WG,WGØ,WGF,PCIE,XMRE,PGG,XISPC,XISPE,FE,WEØ,WEF,TPIR(2)
      IDL  = 0
      IDLA = 0
      IF (IND-1)108,15,110
  108 XNP(2) = CØN5 * XNP(1)
  109 WCØ = Q(2) * RHPI(2)
      PPIS(2) = PPI(2) - (CØN1*WCØ**2/RHPI(2) )
      CALL PUMP1(2)
      PDEL1 = CØN2 * WCØ**2/RHPD(2)
   15 PDEL2 = CØN3 * WCØ**2/RHPD(2)
      PII(2)=PPD(2) - PDEL1 - PDEL2
   10 PDEL3 = 144.*WCØ**2  /(2.*RHPD(2)*32.174*(CD2*CØN4)**2)
      CALL UVAR(PDEL3+.06,7,CDD2 )
      IF(ABS(CD2-CDD2) - (TØL3*CD2))13,13,11
   11 CALL ESTM(IDL,2,TCD2,TCDD2,CD2-CDD2,CD2,LUP3 )
      GØ TØ 10
   13 PCIE = PII(2)-PDEL3
      IDL  = 0
  110 IF(ABS(PCIE-PCIEL) - (TØL5*PCIE))150,150,111
  111 CALL ESTM(IDLA,2,TQ,TQY,PCIE-PCIEL,Q(2),LUP5 )
      GØ TØ 109
  150 RETURN
      END
```

FIGURE 98. SYMBOLIC LISTING OF SUBROUTINE LOX

LXXIX. SUBROUTINE FUEL

This subroutine calculates the hydrogen mass flow rate of an RL10-A1 engine. The following input data is required:

a. Fuel pump inlet conditions

b. Liquid oxygen mass flow rate

c. Engine mixture ratio

d. Injector end chamber pressure

e. Flow loss coefficient of each component

This subroutine uses COMMON blocks CPUMP, MNRL10, CCHAM, CMAIN, CBAL, and FLRL10.

CALLING SEQUENCE

The subroutine calling sequence is:

CALL FUEL (IND, IDLC)

where:

IND = Indicator designating the correct entry point to Subroutine LOX

IDLC = Iteration loop control counter

SOLUTION METHOD

Reset counters used in Subroutine ESTM

1. IDLB = 0

2. IDLD = 0

Fuel pump inlet density:

3. CALL BIVAR (Ln(PPI(1)), TPIS(1), 1, RHPI(1))

4. Q(1) = WCO/((XMRTC)(RHPI(1)))

5. PIS = PPI(1)

6. CALL BIVAR (Ln(PIS), TPIS(1), 1, RHPI(1))

Fuel mass flow rate:

7. WCF = (Q(1))(RHPI(1))

Fuel pump inlet static pressure:

8. $PPIS(1) = PPI(1) - ((CON6)(WCF)^2/RHPI(1))$

9. If $|PIS - PPIS(1)| - ((TOL4)(PIS)) \leq 0$, go to 12

   If $|PIS - PPIS(1)| - ((TOL4)(PIS)) > 0$, go to 10

10. CALL ESTM (IDLD, 2, TCD2, TCDD2, (PIS - PPIS(1)), PIS, LUP4)

11. Go to 6

Calculate fuel pump performance:

```
12.  CALL PUMP1(1)

13.  IDLD = 0
```

Pressure drops from fuel pump outlet to jacket inlet manifold:

```
14.  $PDEL4 = (CON7)(WCF)^2/RHPD(1)$

15.  $PDEL5 = (CON8)(WCF)^2/RHPD(1)$
```

Engine mixture ratio:

```
16.  XMRTC = WCO/WCF
```

Pressure and temperature across the jacket inlet manifold, jacket, and jacket exit manifold:

```
17.  If $(XMRTC - 4.2) \leq 0$, go to 27
     If $(XMRTC - 4.2) > 0$, go to 18

18.  If $(XMRTC - 5.8) \leq 0$, go to 19
     If $(XMRTC - 5.8) > 0$, go to 23
```

Compute constants based on XMRTC:

```
19.  XKON1 = ((XMRTC)(.05438) + .5681)(.97)

20.  XKON2 = ((XMRTC)(.07) + .415)(1.53)

21.  XKON3 = (XMRTC)(.07) + .415

22.  Go to 30
```

```
23. XKON1 = .8583
24. XKON2 = 1.15613
25. XKON3 = .821
26. Go to 30
27. XKON1 = .77891
28. XKON2 = 1.08477
29. XKON3 = .709
30. PDEL6 = $((1.439)(WCF)^2/RHPD(1))(XKON3)$
31. PFJ = PPD(1) - PDEL4 - PDEL5 - PDEL6
32. If (PFJ - 800) $\geq$ 0, go to 38
    If (PFJ - 800) $<$ 0, go to 33
```

Check PCIE to determine which curve to use:

```
33. If ((PCIE)(1.008) - 274) $\leq$ 0, go to 34
    If ((PCIE)(1.008) - 274) $>$ 0, go to 36
34. N = 10
35. Go to 39
36. N = 30
37. Go to 39
38. N = 11
39. CALL BIVAR (PFJ, (PCIE)(1.008), N, TRCJ)
```

```
40.  CALL BIVAR (PFJ, 5, 12, TRCJN)
41.  TRC = (TRCJ)/(TRCJN)
42.  CALL BIVAR (PFJ, XMRTC, 12, TRCJM)
43.  CALL UVAR (1, 29, TRC1)
44.  TJTR = (TRCJM)(TRC)(TRC1)(XKON1)
45.  TOJE = TPDS + TJTR
```

Check PCIE to see which curve to use:

```
46.  If ((PCIE)(1.008)) - 260 ≤ 0, go to 47
     If ((PCIE)(1.008)) - 260 > 0, go to 49
47.  N = 13
48.  Go to 53
49.  If PFJ - 800 ≥ 0, go to 52
     If PFJ - 800 < 0, go to 50
50.  N = 31
51.  Go to 53
52.  N = 14
53.  CALL BIVAR ((PCIE)(1.008), PFJ, N, PJL)
54.  CALL BIVAR (PFJ, 5, 15, PJLNM)
55.  PLJ = (PJL)/(PJLNM)
56.  CALL BIVAR (PFJ, XMRTC, 15, PJL1)
```

57. PDEL7 = (PJL1)(PLJ)(.58)

58. PIJE = PFJ - PDEL7

Check PCIE to determine which curve to use:

59. If ((PCIE)(1.008)) - 260 $\le$ 0, go to 60
    If ((PCIE)(1.008)) - 260 $>$ 0, go to 62

60. N = 16

61. Go to 66

62. If PFJ - 800 $\ge$ 0, go to 65
    If PFJ - 800 $<$ 0, go to 63

63. N = 32

64. Go to 66

65. N = 17

66. CALL BIVAR ((PCIE)(1.008), PFJ, N, PCJ)

67. CALL BIVAR (PFJ, 5, 18, PCJN)

68. PLJE = (PCJ)/(PCJN)

69. CALL BIVAR (PFJ, XMRTC, 18, PCJM)

70. PDEL8 = (PCJM)(PLJE)(XKON2)

71. POJE = PIJE - PDEL8

Pressure drop from jacket exit manifold to venturi inlet:

72. PDEL9 = $(\mathrm{CON10})(\mathrm{WCF})^2(\mathrm{TOJE})/(\mathrm{POJE})$

Fuel total pressure at venturi inlet:

73. POVI = POJE - PDEL9

Fuel mass flow rate at the venturi throat from compressible flow relationships:

74. CALL BIVAR ((TOJE)(.7907), (POVI)(.5075), 19, GAMA1)

75. TSVT = (2)(TOJE)/(GAMA1 + 1)

76. PSVT = $(\mathrm{POVI})((2)/(\mathrm{GAMA1} + 1))^{((\mathrm{GAMA1})/(\mathrm{GAMA1} - 1))}$

77. CALL BIVAR (TSVT, PSVT, 19, GAMA2)

78. CALL UVAR (GAMA2, 20, FLOPT)

79. WCF1 = $(\mathrm{FLOPT})(\mathrm{POVI})(\mathrm{AVT})/(\mathrm{TOJE})^{1/2}$

Test to determine if the mass flow rate of fuel at the venturi throat matches the mass flow rate of fuel at the pump inlet:

80. If $|\mathrm{WCF} - \mathrm{WCF1}| - ((\mathrm{TOL6})(\mathrm{WCF})) \leq 0$, go to 83

    If $|\mathrm{WCF} - \mathrm{WCF1}| - ((\mathrm{TOL6})(\mathrm{WCF})) > 0$, go to 81

New value of fuel pump inlet volume flow rate:

81. CALL ESTM (IDLB, 2, TQQ, TQQ1, WCF - WCF1,

    Q(1), LUP6)

82. Go to 7

83. If KBAL = 1, go to 84

    If KBAL = 2, go to 88

Test to determine if calculated mixture ratio matches the loaded value:

84. If $|XMRTC - XMRTCL| - ((TOL7)(XMRTC)) \leq 0$, go to 88

If $|XMRTC - XMRTCL| - ((TOL7)(XMRTC)) > 0$, go to 85

New value of ground mixture ratio control value flow loss coefficient:

85. CALL ESTM (IDLC, 2, TCON, TTCON, XMRTC - XMRTCL,
CON3, LUP7)

Set counter for correct entry into Subroutine LOX:

86. IND = 1

87. Return

Reset counter used in Subroutine ESTM:

88. IDLC = 0

Set counter so as not to re-enter Subroutine LOX:

89. IND = 0

90. Go to 87

SUBROUTINE FUEL NOMENCLATURE

Symbol	Description	Units	Reference
AVT	Venturi throat area	in^2	79
CON3	Ground mixture ratio control value flow loss coefficient	$\frac{lbf\text{-}sec^2}{lbm\text{-}ft^3\text{-}in^2}$	/LXRL10/, 85
CON6	Constant used in calculating fuel pump inlet static pressure	$\frac{lbf\text{-}sec^2}{lbm\text{-}ft^3\text{-}in^2}$	/FLRL10/, 8
CON7	Flow loss coefficient for line from fuel pump discharge to jacket inlet manifold	$\frac{lbf\text{-}sec^2}{lbm\text{-}ft^3\text{-}in^2}$	/FLRL10/, 14
CON8	Flow loss coefficient for fuel pump discharge orifice	$\frac{lbf\text{-}sec^2}{lbm\text{-}ft^3\text{-}in^2}$	/FLRL10/, 15
CON10	Flow loss coefficient for line from jacket exit manifold to venturi inlet	$\frac{lbf\text{-}sec^2}{lbm\text{-}ft^3\text{-}in^2}$	/FLRL10/, 72
FLOPT	Venturi throat mass flow parameter	$\frac{lbm\text{-}{}^{o}R^{1/2}}{lbf\text{-}sec}$	/FLRL10/, 78, 79
GAMA1	Specific heat ratio of fuel at venturi throat	—	74, 75, 76
GAMA2	Specific heat ratio of fuel at venturi throat	—	77, 78
IDLB	Iteration loop control counter	—	1, 81

SUBROUTINE FUEL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
IDLC	Iteration loop control counter	—	CALL, 85, 88
IDLD	Iteration loop control counter	—	2, 10, 13
IND	Indicator to determine correct entry point into Subroutine LOX	—	CALL, 86, 89
KBAL	Balance/rebalance indicator	—	/CBAL/, 83
LUP4	—	—	DATA, 10
LUP6	—	—	DATA, 81
LUP7	—	—	DATA, 85
N	Set to correct curve number for use in Subroutine BIVAR	—	34, 36, 38, 39, 47, 50, 52, 53, 60, 63, 65, 66
PCIE	Injector end chamber pressure	$\frac{\text{lbf}}{\text{in}^2}$	/CMAIN/, 33, 39, 46, 53, 59, 66
PCJ	Pressure loss factor for jacket exit manifold with chamber pressure and jacket inlet pressure effects	—	66, 68
PLJE	Pressure loss factor for jacket exit manifold	—	68, 70

SUBROUTINE FUEL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PCJM	Pressure loss factor for jacket exit manifold with jacket inlet pressure and mixture ratio effects	—	69, 70
PCJN	Pressure loss factor for jacket exit manifold with jacket inlet pressure effect and nominal mixture ratio	—	67, 68
PDEL4	Line pressure drop from fuel pump discharge to jacket inlet manifold	$\frac{\text{lbf}}{\text{in}^2}$	14, 31
PDEL5	Pressure drop across fuel pump discharge orifice	$\frac{\text{lbf}}{\text{in}^2}$	15, 31
PDEL6	Pressure drop across jacket inlet manifold	$\frac{\text{lbf}}{\text{in}^2}$	30, 31
PDEL7	Pressure drop across tubes in jacket	$\frac{\text{lbf}}{\text{in}^2}$	57, 58
PDEL8	Pressure drop across jacket exit manifold	$\frac{\text{lbf}}{\text{in}^2}$	70, 71
PDEL9	Pressure drop in line from jacket exit manifold to venturi inlet	$\frac{\text{lbf}}{\text{in}^2}$	72, 73
PFJ	Fuel total pressure at jacket inlet manifold exit	$\frac{\text{lbf}}{\text{in}^2}$	31, 32, 39, 40, 42, 49, 53, 54, 56, 58, 62, 67, 69

SUBROUTINE FUEL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PIJE	Total pressure of fuel at inlet to jacket exit manifold	$\frac{lbf}{in^2}$	58, 71
PIS	Pump inlet static pressure	$\frac{lbf}{in^2}$	5, 6, 9, 10
PJL	Jacket pressure loss factor with chamber pressure and jacket inlet pressure effects	—	53, 55
PJL1	Jacket pressure loss factor with mixture ratio and jacket inlet pressure effects	—	56, 57
PJLNM	Jacket pressure loss factor with jacket inlet pressure effect and nominal mixture ratio	—	54, 55
PLJ	Jacket pressure loss factor	—	55, 57
POJE	Total pressure of fuel at exit of jacket exit manifold	$\frac{lbf}{in^2}$	71, 72, 73
POVI	Total pressure of fuel at venturi inlet	$\frac{lbf}{in^2}$	73, 74, 76, 79
PPD	Pump discharge total pressure	$\frac{lbf}{in^2}$	/CPUMP/, 31
PPI	Pump inlet total pressure	lbf/in^2	/CPUMP/, 3, 5, 8
PPIS	Pump inlet static pressure	$\frac{lbf}{in^2}$	/CPUMP/, 8, 9, 10

SUBROUTINE FUEL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PSVT	Static pressure of fuel at venturi throat	$\frac{lbf}{in^2}$	76, 77
Q	Pump inlet volume flow rate	$\frac{ft^3}{sec}$	/MNRL10/, 4, 7, 81
RHPD	Pump discharge density	lbm/ft^3	/CPUMP/, 14, 15, 30
RHPI	Pump inlet density	$\frac{lbm}{ft^3}$	/CPUMP/, 3, 4, 6, 7, 8
TCD2	Storage array for independent argument in Subroutine ESTM	—	DIM, 10
TCDD2	Storage array for dependent argument in Subroutine ESTM	—	DIM, 10
TCON	Storage array for independent argument in Subroutine ESTM	—	DIM, 85
TJTR	Jacket total temperature rise	^oR	44, 45
TOJE	Jacket exit total temperature	^oR	/FLRL10/, 45, 72, 74, 75, 79
TOL4	Fuel pump inlet static pressure tolerance	—	/FLRL10/, 9
TOL6	Fuel mass flow rate tolerance	—	/FLRL10/, 80
TOL7	Mixture ratio tolerance	—	/FLRL10/, 84
TPDS	Pump discharge static temperature	^oR	/CPUMP/, 45

SUBROUTINE FUEL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TPIS	Pump inlet static temperature	$^{\circ}R$	/CPUMP/, 3, 6
TQQ	Storage array for independent argument in Subroutine ESTM	—	DIM, 81
TQQ1	Storage array for dependent argument in Subroutine ESTM	—	DIM, 81
TRC1	Jacket temperature rise factor	—	43, 44
TRC	Jacket temperature rise factor	—	41, 44
TRCJ	Jacket temperature rise factor with chamber pressure and jacket inlet pressure effects	—	39, 41
TRCJM	Jacket temperature rise factor with jacket inlet pressure and mixture ratio effects	—	42, 44
TRCJN	Jacket temperature rise factor with jacket inlet pressure effect and nominal mixture ratio	—	40, 41
TSVT	Static temperature of fuel at venturi throat	$^{\circ}R$	75, 77
TTCON	Storage array for dependent argument in Subroutine ESTM	—	DIM, 85
WCF	Fuel mass flow rate	$\frac{lbm}{sec}$	/CCHAM/, 7, 8, 14, 15, 16, 30, 72, 80, 81

SUBROUTINE FUEL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WCF1	Mass flow rate of fuel at venturi throat	$\frac{lbm}{sec}$	/FLRL10/, 79, 80, 81
WCO	Oxidizer mass flow rate	$\frac{lbm}{sec}$	/CCHAM/, 4, 16
XKON1	Jacket temperature rise factor	—	19, 23, 27, 44
XKON2	Jacket exit manifold pressure loss factor	—	20, 24, 28, 70
XKON3	Jacket pressure loss factor	—	21, 25, 29, 30
XMRTC	Thrust chamber mixture ratio	—	/CCHAM/, 4, 16, 17, 18, 19, 20, 21, 42, 56, 69, 84, 85

```
$IBFTC FUEL    FULIST,REF
      SUBRØUTINE FUEL(IND,IDLC)
      DIMENSIØN TCD2(20),TCDD2(20),TQQ(20),TQQ1(20),TCØN(20),TTCØN(20)
      DATA LUP4/6H  PIS   /  ,LUP6/6H  Q    /, LUP7/6H CØN3  /
      CØMMØN /MNRL10/ PCIEL,Q(2),XMRTCL,PII(2)
      CØMMØN /LXRL10/ TØL3,TØL5,CD2,CØN1,CØN2,CØN3,CØN4,CØN5
      CØMMØN /FLRL10/ CØN8,CØN6,TØL4,FLØPT,CØN7,TØJE,WCF1,TØL6,TØL7,
     1CØN10,AVT
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),     PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
      CØMMØN /CGPRØP/      C3(6),C4(6),C5(6),C6(6),CSG,GAMG,CPG,XMWG,
     1XMRG,RG,XØMG
      CØMMØN /CCHAM/  INDPR ,CF ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCØ
      CØMMØN /CBAL/KBAL /CPA/PA /CRESGG/RESØG,RESFG
      CØMMØN /CMAIN/WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),TN,
     1ØRES(4),WG,WGØ,WGF,PCIE,XMRE,PGG,XISPC,XISPE,FE,WEØ,WEF,TPIR(2)
      IDLB = 0
      IDLD = 0
C  FUEL SIDE EQUATIØNS
  150 CALL BIVAR(ALØG(PPI(1)),TPIS(1),1,RHPI(1)   )
      Q(1) = WCØ / (XMRTC * RHPI(1)   )
      PIS = PPI(1)
  152 CALL BIVAR(ALØG(PIS),TPIS(1),1,RHPI(1)   )
  154 WCF = Q(1) * RHPI(1)
      PPIS(1) = PPI(1) - (CØN6 * WCF**2) / RHPI(1)
      IF(ABS(PIS-PPIS(1)) - (TØL4*PIS))157,157,156
  156 CALL ESTM(IDLD,2,TCD2,TCDD2,PIS-PPIS(1),PIS,LUP4   )
      GØ TØ 152
  157 CALL PUMP1(1)
      IDLD = 0
      PDEL4 = CØN7*WCF**2/RHPD(1)
      PDEL5 = CØN8*WCF**2/RHPD(1)
      XMRTC = WCØ/WCF
C JACKET EQUATIØNS
      IF(XMRTC-4.2)207,207,201
  201 IF(XMRTC-5.8)203,203,206
  203 XKØN1 = (XMRTC*.05438+.5681) * .97
      XKØN2 = (XMRTC*.07+.415) * 1.53
      XKØN3 =  XMRTC*.07+.415
      GØ TØ 208
  206 XKØN1 = .8583
      XKØN2 = 1.15613
      XKØN3 = .821
      GØ TØ 208
  207 XKØN1 = .77891
      XKØN2 = 1.08477
```

FIGURE 99. SYMBOLIC LISTING OF SUBROUTINE FUEL

```
      XKØN3 = .709
208   PDEL6 =(1.439*WCF**2 /RHPD(1)) * XKØN3
      PFJ = PPD(1) - PDEL4 - PDEL5 - PDEL6
      IF(PFJ-800.)210,214,214
210   IF(PCIE*1.008-274.)211,211,213
211   N = 10
      GØ TØ 212
213   N = 30
      GØ TØ 212
214   N = 11
212   CALL BIVAR(PFJ , PCIE*1.008,N, TRCJ)
      CALL BIVAR(PFJ , 5.,12, TRCJN )
      TRC = TRCJ/TRCJN
      CALL BIVAR(PFJ , XMRTC, 12,TRCJM )
      CALL UVAR (1.,29,TRC1 )
      TJTR = TRCJM * TRC * TRC1 *XKØN1
      TØJE = TPDS + TJTR
      IF(PCIE*1.008 - 260.)220,220,222
220   N = 13
      GØ TØ 225
222   IF(PFJ  - 800.)223,224,224
223   N = 31
      GØ TØ 225
224   N = 14
225   CALL BIVAR(PCIE * 1.008, PFJ, N, PJL   )
      CALL BIVAR(PFJ , 5., 15, PJLNM )
      PLJ =  PJL  /PJLNM
      CALL BIVAR( PFJ, XMRTC,15,PJL1   )
      PDEL7 = PJL1 *  PLJ * .58
      PIJE  = PFJ - PDEL7
      IF(PCIE * 1.008 - 260.)230,230,232
230   N = 16
      GØ TØ 235
232   IF(PFJ - 800.)233,234,234
233   N= 32
      GØ TØ 235
234   N = 17
235   CALL BIVAR(PCIE*1.008,PFJ ,N,PCJ )
      CALL BIVAR(PFJ  , 5.,18,PCJN  )
      PLJE  = PCJ  /PCJN
      CALL BIVAR( PFJ  ,XMRTC,18, PCJM   )
      PDEL8 = PCJM * PLJE * XKØN2
      PØJE = PIJE - PDEL8
      PDEL9 = CØN10 * WCF**2 * TØJE / PØJE
      PØVI = PØJE - PDEL9
      CALL BIVAR(TØJE*.7907,PØVI*.5075,19,GAMA1 )
      TSVT =2.*TØJE / (GAMA1+1.)
       PSVT = PØVI* (2./(GAMA1+1.))**(GAMA1/(GAMA1-1.) )
      CALL BIVAR(TSVT,PSVT,19,GAMA2 )
      CALL UVAR(GAMA2,20,FLØPT)
```

FIGURE 99. SYMBOLIC LISTING OF SUBROUTINE FUEL (cont.)

```
      WCF1 = FLØPT*(PØVI*AVT)/SQRT(TØJE)
      IF(ABS(WCF-WCF1) - (TØL6*WCF))160,160,158
  158 CALL ESTM(IDLB,2,TQQ,TQQ1,WCF-WCF1,Q(1),LUP6 )
      GØ TØ 154
  160 IF(KBAL-1)161,161,170
  161 IF(ABS(XMRTC-XMRTCL) - (TØL7*XMRTC))170,170,162
  162 CALL ESTM(IDLC,2,TCØN,TTCØN,XMRTC-XMRTCL,CØN3,LUP7 )
      IND =1
  169 RETURN
  170 IDLC =0
      IND = 0
      GØ TØ 169
      END
```

FIGURE 99. SYMBOLIC LISTING OF SUBROUTINE FUEL (cont.)

LXXX. SUBROUTINE TRBINE

This subroutine calculates the turbine performance of an RL10-A1 engine. The following input data is required:

a. Mass flow rate of fuel
b. Pump heads
c. Pump efficiencies
d. Pump inlet volume flow rates
e. Pump inlet densities
f. Auxilliary horsepower
g. Gearbox efficiency
h. Fuel pump speed
i. Total temperature of fuel at jacket exit
j. Flow loss coefficient of each component

This subroutine uses COMMON blocks TBRL10, CPUMP, CTURB, FLRL10, CMAIN, MNRL10, and CCHAM.

The subroutine calling sequence is:

CALL TRBINE

SOLUTION METHOD

Set ESTM iteration loop control counter:

1. IDLG = 0

Fuel pump horsepower:

2. HP1 = (H(1))(RHPI(1))(Q(1))/((ETAP(1))(550))

Oxidizer pump horsepower:

3. HP2 = (H(2))(RHPI(2))(Q(2))/((ETAP(2))(550))

Turbine horsepower:

4. HPT = HP1 + (HP2 + HPAUX)/(ETGER)

First guess of turbine mass flow rate:

5. WT = (CON9)(WCF)

Actual gas enthalpy change:

6. DELHTA = (.7068)(HPT)/(WT)

Velocity ratio:

7. UVRAT = (.0001199)(XNP(1))/(DELHTA$^{.5}$)

Efficiency:

8. CALL UVAR(Ln(UVRAT), 21, ETAT)

Ideal gas enthalpy change:

9. DELHTI = (DELHTA)/(ETAT)

Inlet to exit pressure ratio:

10. CALL BIVAR((DELHTI)/(TOJE), TOJE, 22, PRT)

Mass flow parameter:

11. CALL UVAR(PRT, 23, FLOPT1)

Mass flow parameter:

12. FLOPT2 = (FLOPT)(FLOPT1)(CON11)

Inlet total pressure:

13. PTI = $(\mathrm{WT})(\mathrm{TOJE}^{.5})/(\mathrm{FLOPT2})$

Exit total pressure:

14. PTE = (PTI)/(PRT)

Exit static pressure:

15. PTES = (CON12)(PTE)

Inlet enthalpy:

16. CALL UVAR(TOJE, 24, HTI)

Exit enthalpy:

17. HTE = HTI - DELHTA

Exit total temperature:

18. CALL UVAR (HTE, 25, TTD)

Enthalpy at turbine exit - bypass line mixing point:

19. HMIX = ((WT)(HTE) + (WCF - WT)(HTI))/(WCF)

Static temperature at turbine exit - bypass line mixing point:

20. CALL UVAR (HMIX, 25, TMIX)

Line pressure drop from turbine exit to fuel injector inlet:

21. PDEL10 = $(\mathrm{CON13})(\mathrm{WCF}^2)(\mathrm{TMIX})/(\mathrm{PTES})$

Pressure drop across main fuel shutoff valve:

22. PDEL11 = $(\mathrm{CON14})(\mathrm{WCF}^2)(\mathrm{TMIX})/(\mathrm{PTES} - \mathrm{PDEL10})$

Fuel injector inlet total pressure:

23. PII(1) = PTES - PDEL10 - PDEL11

Fuel injector inlet constant pressure specific heat:

24. CALL BIVAR (TMIX, (PII(1))/(14.7), 26, CPFII)

Fuel injector inlet total temperature:

25. TFII = TMIX + (20.9135)/((WCF)(CPFII))

Specific heat ratio at fuel injector inlet:

26. CALL UVAR (TFII, 27, GAMA3)

27. If (PII(1))/(PCIE) $\leq$ 1, go to 28

    If (PII(1))/(PCIE) $>$ 1, go to 30

28. CALL UVAR (1, 28, XCD1)

29. Go to 31

30. CALL UVAR ((PII(1))/(PCIE), 28, XCD1)

Fuel injector flow coefficient:

31. CD1 = (XCD1)(1.172) + .13

Mass flow parameter:

32. A = (WCF)($\text{TFII}^{.5}$)/((CON15)(CD1)(PII(1)))

Mass flow parameter:

```
33.  $B = (((64.348)(GAMA3))/((766.52)(GAMA3 - 1)))^{.5}((PCIE)/(PII(1)))^{((1)/(GAMA3))}(1 - ((PCIE)/(PII(1)))^{((GAMA3 - 1)/(GAMA3))})^{.5}$

34.  If $|A - B| \leq (TOL10)(A)$,  go to 37
     If $|A - B| > (TOL10)(A)$,  go to 35

35.  CALL ESTM (IDLG, 2, TWT, TTWT, (A - B), WT, LUP10)

36.  Go to 6

37.  Return
```

SUBROUTINE TRBINE NOMENCLATURE

Symbol	Description	Units	Reference
A	Left-hand member of mass flow parameter equation	$\frac{lbm - {}^{o}R^{1/2}}{lbf - sec}$	32, 34, 35
B	Right-hand member of mass flow parameter equation	$\frac{lbm - {}^{o}R^{1/2}}{lbf - sec}$	33, 34, 35
CD1	Fuel injector flow coefficient	—	31, 32
CON9	First guess of ratio of turbine mass flow rate to total fuel mass flow rate	—	/TBRL10/, 5
CON11	Constant used in computing mass flow parameter	—	/TBRL10/, 12
CON12	Ratio of turbine exit static pressure to turbine exit total pressure	—	/TBRL10/, 15
CON13	Flow loss coefficient for line from turbine exit to injector inlet	$\frac{lbf^2 - sec^2}{in^4 - lbm^2 - {}^{o}R}$	/TBRL10/, 21
CON14	Flow loss coefficient for main fuel shut-off valve	$\frac{lbf^2 - sec^2}{in^2 - lbm^2 - {}^{o}R}$	/TBRL10/, 22
CON15	Fuel injector area	in^2	/TBRL10/, 32
CPFII	Fuel injector inlet constant pressure specific heat	$\frac{btu}{lbm - {}^{o}R}$	24, 25

SUBROUTINE TRBINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DELHTA	Actual gas enthalpy change across turbine	$\frac{btu}{lbm}$	6, 7, 9, 17
DELHTI	Ideal gas enthalpy change across turbine	$\frac{btu}{lbm}$	9, 10
ETAP	Pump efficiency	—	/CPUMP/, 2, 3
ETAT	Turbine efficiency	—	/CTURB/, 8, 9
ETGER	Gear box efficiency	—	/TBRL10/, 4
FLOPT	Venturi throat mass flow parameter	$\frac{lbm - {}^{o}R^{1/2}}{lbf - sec}$	/FLRL10/, 12
FLOPT1	Mass flow parameter	$\frac{lbm - {}^{o}R^{1/2} - in^{2}}{lbf - sec}$	11, 12
FLOPT2	Mass flow parameter	$\frac{lbm - {}^{o}R^{1/2} - in^{2}}{lbf - sec}$	12, 13
GAMA3	Specific heat ratio at fuel injector inlet	—	26, 33
H	Pump head	$\frac{ft - lbf}{lbm}$	/CPUMP/, 2, 3
HMIX	Enthalpy at turbine exit - bypass line mixing point	$\frac{btu}{lbm}$	19, 20
HP1	Fuel pump horsepower	—	/TBRL10/, 2, 4

SUBROUTINE TRBINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
HP2	Oxidizer pump horsepower	—	/TBRL10/, 3, 4
HPAUX	Auxilliary drive horsepower	—	/CMAIN/, 4
HPT	Turbine horsepower	—	/CTURB/, 4, 6
HTE	Turbine exit enthalpy	$\frac{btu}{lbm}$	17, 18, 19
HTI	Turbine inlet enthalpy	$\frac{btu}{lbm}$	16, 17, 19
IDLG	Iteration loop control counter	—	1, 35
LUP10	—	—	DATA, 35
PCIE	Injector end chamber pressure	$\frac{lbf}{in^2}$	/CMAIN/, 27, 30, 33
PDEL10	Line pressure drop from turbine exit to fuel injector inlet	$\frac{lbf}{in^2}$	21, 22, 23
PDEL11	Pressure drop across main fuel shutoff valve	$\frac{lbf}{in^2}$	22, 23
PII	Injector inlet total pressure	$\frac{lbf}{in^2}$	/MNRL10/, 23, 24, 27, 30, 32, 33
PRT	Turbine inlet to exit pressure ratio	—	/CTURB/, 10, 11, 14
PTE	Turbine exit total pressure	$\frac{lbf}{in^2}$	/CTURB/, 14 15

SUBROUTINE TRBINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PTES	Turbine exit static pressure	$\frac{lbf}{in^2}$	/CTURB/, 15, 21, 22, 23
PTI	Turbine inlet total pressure	$\frac{lbf}{in^2}$	/CTURB/, 13, 14
Q	Pump inlet volume flow rate	$\frac{ft^3}{sec}$	/MNRL10/, 2, 3
RHPI	Pump inlet density	$\frac{lbm}{ft^3}$	/CPUMP/, 2, 3
TFII	Fuel injector inlet total temperature	°R	25, 26, 32
TMIX	Static temperature at turbine exit - bypass line mixing point	°R	20, 21, 22, 24, 25
TOJE	Jacket exit total temperature	°R	/FLRL10/, 10, 13, 16
TOL10	Mass flow parameter tolerance	—	/TBRL10/, 34
TTD	Turbine exit total temperature	°R	/CTURB/, 18
TTWT	Storage array for dependent argument in Subroutine ESTM	—	DIM 35
TWT	Storage array for independent argument in Subroutine ESTM	—	DIM, 35

SUBROUTINE TRBINE NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
UVRAT	Turbine velocity ratio	—	/CTURB/, 7, 8
WCF	Fuel mass flow rate	$\frac{\text{lbm}}{\text{sec}}$	/CCHAM/, 5, 19, 21, 22, 25, 32
WT	Turbine mass flow rate	$\frac{\text{lbm}}{\text{sec}}$	/CTURB/, DATA, 5, 6, 13, 19, 35
XCD1	Value used in computing fuel injector flow coefficient	—	28, 30, 31
XNP	Pump speed	$\frac{\text{rev}}{\text{min}}$	/CPUMP/, 7

```
$IBFTC TRBINE  FULIST,REF
      SUBRØUTINE TRBINE
      DIMENSIØN TWT(20), TTWT(20)
      DATA LUP10 /6H  WT   /
      CØMMØN /MNRL10/ PCIEL,Q(2),XMRTCL,PII(2)
      CØMMØN /FLRL10/ CØN8,CØN6,TØL4,FLØPT,CØN7,TØJE,WCF1,TØL6,TØL7,
     1CØN10,AVT
      CØMMØN /TBRL10/ CØN9,ETGER,CØN11,CØN12,CØN13,CØN14,CØN15,TØL10
     1,HP1,HP2
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),      PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
      CØMMØN /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
      CØMMØN /CCHAM/  INDPR ,CF ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCØ
      CØMMØN /CMAIN/WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),TN,
     1ØRES(4),WG,WGØ,WGF,PCIE,XMRE,PGG,XISPC,XISPE,FE,WEØ,WEF,TPIR(2)
      IDLG = 0
      HP1 = H(1)*RHPI(1)*Q(1) /(ETAP(1)*550. )
      HP2 = H(2)*RHPI(2)*Q(2) /(ETAP(2)*550. )
      HPT = HP1 + (HP2+HPAUX) / ETGER
      WT = CØN9 * WCF
  500 DELHTA = .7068*HPT / WT
      UVRAT = .0001199*XNP(1) / SQRT(DELHTA)
      CALL UVAR(ALØG(UVRAT),21,ETAT)
      DELHTI = DELHTA / ETAT
      CALL BIVAR(DELHTI / TØJE , TØJE,22,PRT)
      CALL UVAR(PRT,23,FLØPT1 )
      FLØPT2 = FLØPT * FLØPT1 * CØN11
      PTI = (WT  * SQRT(TØJE))/FLØPT2
      PTE = PTI / PRT
      PTES = CØN12 * PTE
      CALL UVAR(TØJE,24,HTI)
      HTE = HTI - DELHTA
      CALL UVAR(HTE,25,TTD)
      HMIX = (WT*HTE + (WCF-WT)*HTI) / WCF
      CALL UVAR(HMIX,25,TMIX)
      PDEL10 = CØN13 * WCF**2 * TMIX / PTES
      PDEL11 = CØN14 * WCF**2 * TMIX / (PTES - PDEL10)
      PII(1) = PTES- PDEL10 - PDEL11
      CALL BIVAR(TMIX,PII(1)/14.7,26,CPFII )
      TFII = TMIX + 20.9135 / (WCF * CPFII )
      CALL UVAR(TFII,27,GAMA3 )
      IF(PII(1)/PCIE - 1.)501,501,502
  501 CALL UVAR(1.,28,XCD1)
      GØ TØ 503
```

FIGURE 100. SYMBOLIC LISTING OF SUBROUTINE TRBINE

```
502 CALL UVAR(PII(1)/PCIE,28,XCD1 )
503 CD1 = XCD1 * 1.172 + .13
    A = (WCF * SQRT(TFII)) /(CØN15 * CD1 * PII(1)  )
    B = ((64.348*GAMA3)/(766.52*(GAMA3 -1.)) )**.5 * (PCIE/PII(1)) **
   1(1./GAMA3)     * (1.-(PCIE/PII(1))**((GAMA3-1.)/GAMA3))**.5
    IF(ABS(A-B) -(TØL10*A))505,505,504
504 CALL ESTM(IDLG,2,TWT,TTWT,A-B,WT,LUP10 )
    GØ TØ 500
505 RETURN
    END
```

FIGURE 100. SYMBOLIC LISTING OF SUBROUTINE TRBINE (cont.)

LXXXI. SUBROUTINES BIVAR, UVAR AND PTEST

Subroutines BIVAR and UVAR are bivariate and univariate curve-fit routines written in the MAP assembly language. Subroutine PTEST is called from these subroutines if the independent variable is out of range. Subroutine PTEST is written in FORTRAN IV.

These routines are referenced in Marshall Space Flight Center document IN-P & VE-P-62-16.

CALLING SEQUENCE

```
CALL BIVAR (X, Y, N, Z)

CALL UVAR (X, N, Z)

CALL PTEST (TEST, NTEST)
```

where:

X = independent variable

Y = independent variable

N = table number for interpolation

Z = dependent variable

TEST = array containing the independent variable(s) and the limit(s) for the variable(s).

NTEST = Array containing the curve number and an error signal.

```
$IBMAP BIVAUV  FULIST,REF
       ENTRY   BIVAR
       ENTRY   UVAR
*9PØLY      7090 UNIVARIATE AND BIVARIATE PØLYNØMIAL EVALUATUØN
 BIVAR SXA     PEXIT,4
       SXA     PEXIT+1,2
       AXT     6,2
       SXA     PEXIT+2,2
       CLA*    3,4
       STØ     CØMMØN            X-VALUE
       STØ     CØMMØN+1
       CLA*    4,4
       STØ     CØMMØN+2              Y-VALUE
       CLA*    5,4
       STØ     CØMMØN+4              CURVE NUMBER
       PAC     ,2
       CLA     CURVES-1,2
       PAC     ,4
       AXT     0,2
       CLA     CØMMØN
       CAS     1,4               X-MAX FØR ENTIRE CURVE
       TXI     BIVR1,2,2
       TRA     *+1
       CAS     0,4               X-MIN
       TRA     BIVR2
       TRA     BIVR2
       CLA     0,4
       TXI     *+2,2,1
BIVR1  CLA     1,4
       STØ     CØMMØN+1              X-LIMIT
BIVR2  CLA     CØMMØN+2
       CAS     3,4               Y-MAX
       TXI     BIVR3,2,6
       TRA     *+1
       CAS     2,4               Y-MIN
       TRA     BIVR4
       TRA     BIVR4
       CLA     2,4
       TXI     *+2,2,3
BIVR3  CLA     3,4
       STØ     CØMMØN+3              Y-LIMIT
BIVR4  TXL     BIVR5,2,0
       SXA     CØMMØN+5,2            ERRØR SIGNAL
       SXA    BIVR5-1,4
       CALL    PTEST(TEST,NTEST)
       AXT    **,4
BIVR5  CLA     4,4               X-MAX FØR SECTIØN
       CAS     CØMMØN+1
       TRA     BIVR7
       TRA     BIVR7
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR

```
       CLA    5,4              DEGREE FØR SECTIØN
       ADD    ØNE
       PAX    0,2
       LBT
       TXI    *+2,2,1
       ADD    ØNE
       SXD    BIVR6,2
       ARS    1
       PAX    0,2
BIVR6  TIX    *+1,4,**
       TIX    *-1,2,1
       TIX    BIVR5,4,6
BIVR7  LDQ    CØMMØN
       FMP    6,4              ALPHA-X
       FAD    7,4              BETA-X
       STØ    CØMMØN
       LDQ    CØMMØN+2
       FMP    8,4              ALPHA-Y
       FAD    9,4              BETA-Y
       STØ    CØMMØN+2
       CLA    5,4
       ADD    ØNE
       ALS    18
       STD    BIVRC
       AXT    1,2
       STZ    CØMMØN+3
       TRA    BIVRC
BIVR8  SXD    BIVRB,2
       CLA    10,4
       TRA    BIVRA
BIVR9  XCA
       FMP    CØMMØN+2
       FAD    10,4
BIVRA  TIX    *+1,4,1
       TIX    BIVR9,2,1
       STØ    CØMMØN+1
       LDQ    CØMMØN+3
       FMP    CØMMØN
       FAD    CØMMØN+1
BIVRB  TXI    *+1,2,**
       STØ    CØMMØN+3
BIVRC  TNX    BIVR8,2,**
PEXIT  AXT    **,4
       AXT    **,2
       STØ*   **,4
       TRA    1,4
UVAR   SXA    PEXIT+1,2
       AXT    5,2
       SXA    PEXIT+2,2
       CLA*   3,4
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
        STO    COMMON              X-VALUE
        STO    COMMON+1
        CLA*   4,4
        STO    COMMON+4              CURVE NUMBER
        PAC    ,2
        CLA    CURVES-1,2
        SXA    PEXIT,4
        PAC    ,4
        AXT    0,2
        CLA    COMMON
        CAS    1,4                 X-MAX FOR ENTIRE CURVE
        TXI    UVAR1,2,2
        TRA    *+1
        CAS    0,4                 X-MIN
        TRA    UVAR2
        TRA    UVAR2
        CLA    0,4
        TXI    *+2,2,1
UVAR1   CLA    1,4
        STO    COMMON+1              X-LIMIT
        SXA    COMMON+5,2            ERROR SIGNAL
        SXA    UVAR2-1,4
        CALL   PTEST(TEST,NTEST)
        AXT    **,4
UVAR2   CLA    2,4                 X-MAX FOR SECTION
        CAS    COMMON+1
        TRA    UVAR3
        TRA    UVAR3
        CLA    3,4                 DEGREE OF SECTION
        PAX    0,2
        TXI    *+1,2,5
        SXD    *+1,2
        TIX    UVAR2,4,**
UVAR3   LDQ    COMMON
        FMP    4,4                 ALPHA
        FAD    5,4                 BETA
        STO    COMMON+1
        CLA    3,4                 DEGREE
        PAX    0,2
        CLA    6,4
UVAR4   XCA
        FMP    COMMON+1
        FAD    7,4
        TIX    *+1,4,1
        TIX    UVAR4,2,1
        TRA    PEXIT
ONE     DEC    1
*3682CV             CURVES ASSEMBLY
CURVES  PZE NCV1              F-1802  DENSITY = F(LN(P) , TEMP)
        PZE NCV2              F-1801  DENSITY = F(TEMP.)
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
     PZE NCV3            F-5685       SI = F(PHI)  LOX FLOW PARAM.
     PZE NCV4            F-4881      KNO = F(.001NPO) EFFICIENCY CV
     PZE NCV5            F-4878 NPO/KNO  = F(QOP1/NPO/NDES)
     PZE NCV6            DF-15608 CPO = F(TOP1)
     PZE NCV7            F-7030     CDO1 = F(DPO1)
     PZE NCV11           F-4891-A   CPF1 = F(TEMP,P/14.7) TFTK L60.
     PZE NCV12           F-4891-B   CPF2 = F(TEMP,P/14.7) TFTK GR60
     PZE NCV14           F-5092-A    DTC1= F(PJI,PCS(K76)) PJI L800
     PZE NCV15           F-5092-B    DTC1= F(PJI,PCS(K76)) PJIGR800
     PZE NCV16           F-4975      DTC = F(PJI,RM)
     PZE NCV17           F-4977-A    DPJ1= F(PJI,PCS(K76))PCS()L260
     PZE NCV18           F-4977-B    DPJ1= F(PJI,PCS(K76))PCS()G260
     PZE NCV19           F-4974      DPJ2= F(PJI,RM)
     PZE NCV20           F-4979-A   DPJM1= F(PJI,PCS(K76)PCS()L 260
     PZE NCV21           F-4979-B   DPJM1= F(PJI,PCS(K76)PCS()G 260
     PZE NCV22           F-4976     DPJM2= F(PJI,K77) K77 LESS 260
     PZE NCV23           F-2765    GMTJE = F(.833TJE,.52773PJE)
     PZE NCV24           F-4971     WTPV = F(GMTJE)
     PZE NCV25           F-4874 TURB.EFF.= F(LN(NUT*K78))
     PZE NCV26           F-4875   POO/P2 = F(DHT/TJE,TJE)
     PZE NCV27           F-4876   WTJPP  = F(POO/P2)
     PZE NCV28           F-4877 ENTHL. HV= F(TEMP.)
     PZE NCV29           F-4877REV TEMP. = F(ENTHALPY)
     PZE NCV30               TD-25061  CPFI = F(TEMP,P114.7)
     PZE NCV31           F-4885     GMFI = F(TEMP)
     PZE NCV32           F-4884     CDFI = F(PFI/PCS)
     PZE NCV39               F-7285  DTK80 = F(1,1K80)
     PZE NCV55             F-5092-C COVER GAP IN CURVE 14
     PZE NCV56             F-4977-C COVER GAP IN CURVE 18
     PZE NCV57             F-4979-C COVER GAP IN CURVE 21
     PZE NCV46               F-10875   CSTP = F(RM,PT)
NCV1 DEC 2.0,7.3,35.,60.     F-1802    BIVAR
     DEC 4.2  END POINT OF FIRST SECTION =X1MAX
     DEC 5    DEGREE OF FIRST SECTION =5
     OCT 200721350565,602550564272,175507534121,602746314626
     OCT 571433766223,566645661551,170771214711,171631207704
     OCT 567562263325,171704336472,570570514344,572513360524
     OCT 570427073650,171672113111,576566003206,170767467445
     OCT 177656216367,177506350351,175544552536,200412547055
     OCT 177624170737,600573063526,601417654343,601445235536
     OCT 202746600064
     DEC 5.8  END POINT OF SECOND SECTION =X2MAX
     DEC 5    DEGREE OF SECOND SECTION =5
     OCT 201500000001,603620000000,175507534121,602746314626
     OCT 570411213321,566466043633,170570305751,570557174300
     OCT 567477612544,172451634652,170455070526,567707346263
     OCT 571601202745,172775135745,576400252541,172602416734
     OCT 177422224100,176631237254,176404323104,176667045724
     OCT 175547501725,577540377257,577607371702,600552222740
     OCT 203401254130
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
     DEC 7.3  END POINT OF LAST SECTION =X3MAX
     DEC 5    DEGREE OF THIRD SECTION =5
     OCT 201525252525,604427356735,175507534121,602746314626
     OCT 573552161244,571743636627,172451617763,571521441014
     OCT 571742701124,174506632415,172620416235,573520417224
     OCT 572705466671,174443743553,170500001410,174500126512
     OCT 175430011276,176401644606,177411206325,174401617402
     OCT 573772523351,575434545714,574535154262,577666066165
     OCT 203432050672
     BSS 20
NCV2 DEC 140.,280.             F1801 UVAR
     DEC 260.                  X1MAX
     DEC 5                     DEG.
     OCT 173421042104,602652525252                                F1801 04
     OCT 602415375600,601454140500,201627431560,601475223760      F1801 05
     OCT 604652562455,206777452050                                F1801 06
     DEC 280.                  X2MAX
     DEC 5                     DEG.
     OCT 175631463146,605660000000
     OCT 175734416000,575436026000,600561772400,602472674540      F1801 05
     OCT 603743473072,206520234472                                F1801 06
     BSS 20
NCV3 DEC .03,.145          F-5685 UVAR
     DEC .145         X1MAX
     DEC 4            DEGREE
     OCT 205426205441,601605441307
     OCT 573640776465,574417703274,574720735300,174510162333
     OCT 200544377175
     BSS 20
NCV4 DEC 0.,15.        F-4881  UVAR
     DEC 4.           X1MAX
     DEC 5            DEG.
     OCT 200400000000,601400000000
     OCT 173507532300,573631464640,174507535320,575772702160
     OCT 177474121664,200410753410
     DEC 15.          X2MAX
     DEC 5            DEG
     OCT 176564272135,601672135056                                IBM15B04
     OCT 172665517400,573417271500,573552424000,575566466324      IBM15B05
     OCT 176402735575,200755266631                                IBM15B06
     BSS 20
NCV5 DEC 0.,.725      F-4878  UVAR
     DEC .425         X1MAX
     DEC 5            DEG.
     OCT 203455132265,601400000000                                IBM12A04
     OCT 173440307300,574533264240,172623156340,575543567104      IBM12A05
     OCT 176702600110,177604264237                                IBM12A06
     DEC .725         X2MAX
     DEC 5            DEG.
     OCT 203652525253,602752525252                                IBM12B04
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
      ØCT 574643552550,5754 21171714,171575746500,574414265432  IBM12B05
      ØCT 575763450473,177673153367                             IBM12B06
      BSS 20
NCV6  DEC 150.,250.,250.,5        DF-15608
      ØCT 173507534121,603400000000
      ØCT 171761760000,173576022666,174445641100,174546470662
      ØCT 174657463150,177666013502
NCV7  DEC 0.,102.5             F-7030  UVAR
      DEC 20.                  X1MAX
      DEC 5                    DEG.
      ØCT 176421042104,601652525252
      ØCT 167530210000,570462461400,170475202400,571532730000
      ØCT 172400440740,200632517765
      DEC 50.                  X2MAX
      DEC 5                    DEG
      ØCT 175421042104,602452525252
      ØCT 163511340000,166416136000,166416333000,572461651740
      ØCT 573537642734,200626275454
      DEC 102.5                X3MAX
      DEC 5                    DEG.
      ØCT 174470047004,602563636363
      ØCT 167643226000,170636076400,571676213600,171701466200
      ØCT 573565601410,200563566257
      BSS 20
NCV11 DEC 26.,60.,1.,115.         F-4891-A  BIVAR
      DEC 46.                     X1MAX
      DEC 5                       DEG.
      ØCT 175631463146,602714631462,173421042104,600725252524
      ØCT 175431366431,572621616276,172500000731,175661511704
      ØCT 575616544375,575661102277,575401661546,176645443164
      ØCT 576432046062,174672770201,173401673566,576476624616
      ØCT 176722313774,576650462340,177741323342,177460373012
      ØCT 565471004156,600433444766,176433754003,575775457326
      ØCT 202407106527
      DEC 60.                     X2MAX
      DEC 5                       DEG.
      ØCT 176444444444,603744444442,173421042104,600725252524
      ØCT 173663122365,575413622450,175406137041,177562323265
      ØCT 577447174457,574530611263,600724654122,201434423551
      ØCT 576443567570,575632713237,200505562014,601642167351
      ØCT 201474432001,600424050530,200400754465,200752016515
      ØCT 177751606574,602464212673,201527661005,600741502351
      ØCT 202606723753
      BSS 20
NCV12 DEC 60.,70.,25.,115.        F-4891-B  BIVAR
      DEC 65.                     X1MAX
      DEC 5                       DEG.
      ØCT 177631463146,605617777777,173554055405,601616161614
      ØCT 174770606207,174463110723,170711115600,572453535625
      ØCT 564505031457,575527263624,574535555552,173635306375
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
      ØCT 573603332550,571603371507,200623214530,600477642273
      ØCT 577547732042,175523603216,176634763117,602470254015
      ØCT 201760320405,201623765573,577617617566,601544076426
      ØCT 202671767376
      DEC 70.                     X2MAX
      DEC 5                       DEG.
      ØCT 177631463146,605657777777,173554055405,601616161614
      ØCT 174770605530,574763777575,173666061035,176515757554
      ØCT 575542222164,575732001423,576615555611,175745701262
      ØCT 176442606021,574654051737,200755207724,600712035204
      ØCT 600437240620,177427754374,176705624226,602713672264
      ØCT 202715453575,201634172662,601442345204,601566354176
      ØCT 202745351233
      BSS 20
NCV14 DEC 600.,800.,149.,274.        F-5092 BIVAR
      DEC 800.                       X1MAX
      DEC 5
      ØCT 172507534121,603677777776,173406111564,602661115644         00
      ØCT 173414646464,174433540711,576605424250,175766310222         01
      ØCT 576657047602,177553702330,177407355337,175505262066         02
      ØCT 600423433120,602401105017,176735615414,575646554616         03
      ØCT 575433312640,201777217774,203616720334,202505166616         04
      ØCT 601405446401,603402743521,203424205750,605563353350         05
      ØCT 211512642457                                                06
      BSS 20
NCV15 DEC 800.,1500.,149.,399.     F-5092-B BIVAR
      DEC 1500.                    X1MAX
      DEC 5                        DEG.
      ØCT 170566373471,602644444444,172406111564,602430446721   -12-6000
      ØCT 200500121172,576777152431,600763701570,201455156004   -12-6001
      ØCT 201723620543,201712116005,601702076614,601652665112   -12-6002
      ØCT 602501170465,603545755212,200452537722,177407301731   -12-6003
      ØCT 576514146417,200511631603,603606571314,201542777766   -12-6004
      ØCT 203436707661,602650130710,203446345603,605777507510   -12-6005
      ØCT 211476017555                                          -12-6006
      BSS 20
NCV16 DEC 700.,1500.,3.,8.         F-4975  BIVAR
      DEC 1100.                    X1MAX
      DEC 5                        DEG.
      ØCT 171507534121,603437777777,177631463146,602431463146    SN-DJ00
      ØCT 200612625364,200731237020,577706413506,200465125333    SN-DJ01
      ØCT 600562126714,177631401742,200710770466,573531057533    SN-DJ02
      ØCT 577461107250,602766472077,201600622564,202404041574    SN-DJ03
      ØCT 603467070467,601400445121,203444172255,603523717364    SN-DJ04
      ØCT 603644405357,205763357630,606573673773,206751171454    SN-DJ05
      ØCT 211506516376                                           SN-DJ06
      DEC 1500.                    X2MAX
      DEC 5                        DEG.
      ØCT 171507534121,603637777776,177631463146,602431463146         00
      ØCT 577424051134,574535463405,574645540232,177403724173         01
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
      ØCT 577635661631,200413476143,601702574277,576546562041           02
      ØCT 200752107130,601461261312,600404476530,602516465274           03
      ØCT 601472526723,202662777476,603473632305,602510334540           04
      ØCT 603701602404,205724516030,606632752111,206771015377           05
      ØCT 211504516026                                                  06
      BSS 20
NCV17 DEC 150.,260.,600.,1300.      F-4977-A BIVAR
      DEC 260.                      X1MAX
      DEC 5                         DEG.
      ØCT 173451710112,602735056426,170566373471,602533333333     FERSØN00
      ØCT 176470110546,600716273073,200404674111,203502515771     FERSØN01
      ØCT 602646070051,577535641441,604436562241,203552614342     FERSØN02
      ØCT 201672517721,202557542040,203702510643,603770050512     FERSØN03
      ØCT 201551604757,604614666231,206412364641,603746201774     FERSØN04
      ØCT 203566313704,203525031264,203734565044,605775322563     FERSØN05
      ØCT 207431617733                                            FERSØN06
      BSS 20
NCV18 DEC 260.,400.,800.,1500.      F-4977-8 BIVAR
      DEC 350.                      X1MAX
      DEC 5                         DEG.
      ØCT 173554055405,603661616160,170566373471,602644444444     ØN    00
      ØCT 201515665161,574652143217,200533430673,175507174774     ØN    01
      ØCT 600636525516,601432452063,602567440515,202606641117     ØN    02
      ØCT 602431351324,202416731664,203677137266,604446606376     ØN    03
      ØCT 203510371444,604617761770,206400413253,602656567451     ØN    04
      ØCT 203627713123,603552303520,204775704474,606526277755     ØN    05
      ØCT 207711325665                                            ØN    06
      DEC 400.                      X2MAX
      DEC 5                         DEG.
      ØCT 174507534121,604737777776,170566373471,602644444444     ØN    00
      ØCT 177561625055,200637631266,601400405055,602556212735     ØN    01
      ØCT 202422141130,600511313415,603430422355,203455575266     ØN    02
      ØCT 600542771545,201574247210,205563354231,605631464023     ØN    03
      ØCT 203557441407,604520257214,205567060410,605571354322     ØN    04
      ØCT 206423171550,604575130017,205615271700,607447555004     ØN    05
      ØCT 210527365644                                            ØN    06
NCV19 DEC 700.,1500.,3.,8.           F-4974 BIVAR
      DEC 1100.                      X1MAX
      DEC 5                          DEG.
      ØCT 171507534121,603437777777,177631463146,602431463146           00
      ØCT 602767153455,604534544762,203547645202,604576467470           01
      ØCT 204711217003,577674523520,202563603040,204455753036           02
      ØCT 604531006322,204532757542,205501657372,603675440747           03
      ØCT 605526514705,205604530274,606467144061,205620243642           04
      ØCT 605402423171,605673302077,205527221574,606605071743           05
      ØCT 210414417201                                                  06
      DEC 1500.                      X2MAX
      DEC 5                          DEG
      ØCT 171507534121,603637777776,177631463146,602431463146     N     00
      ØCT 202404412024,201456407544,600445227201,602435466407     N     01
      ØCT 202537452447,602451315515,601643201273,200676446202     N     02
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
      ØCT 602603545016,203415654327,601577161561,602444312213  N       03
      ØCT 203577507355,202665670307,605403574267,203701165772  N       04
      ØCT 202716515014,604474120776,200410630064,605707366501  N       05
      ØCT 207533253534                                         N       06
      BSS 20
NCV20 DEC 150.,260.,600.,1300.      F-4979-A
      DEC 260.                      X1MAX
      DEC 5                         DEG.
      ØCT 173451710112,602735056426,170566373471,602533333333
      ØCT 177424240162,576631645032,177504511751,177764117753  FERSØNO1
      ØCT 600431634043,577514170720,601704724301,201616653203  FERSØNO2
      ØCT 576455102323,200444702716,200771175507,601753643606  FERSØNO3
      ØCT 202456526503,602755174266,204424462601,600551216622  FERSØNO4
      ØCT 201450624426,600565074650,202503773046,603720055235  FERSØNO5
      ØCT 205417213506                                         FERSØNO6
      BSS 20
NCV21 DEC 260.,400.,800.,1500.      F-4979 BIVAR
      DEC 350.                      X1MAX
      DEC 5                         DEG.
      ØCT 173554055405,603661616160,170566373471,602644444444  FERSØNO0
      ØCT 200507024621,175637771735,574542643303,200436214767  FERSØNO1
      ØCT 600471332276,600517234503,601472516236,201534402137  FERSØNO2
      ØCT 600470132715,200611252020,200772143251,602476452662  FERSØNO3
      ØCT 202401600511,602637204157,204426540763,600445720357  FERSØNO4
      ØCT 201607631273,602433122540,202771653353,604530654373  FERSØNO5
      ØCT 205721263406                                         FERSØNO6
      DEC 375.                      X2MAX
      DEC 5                         DEG.
      ØCT 175507534121,605717777776,170566373471,602644444444          00
      ØCT 575566777555,575533160156,176613652133,200457004623          01
      ØCT 577531037445,172543427557,601532614233,200660052510          02
      ØCT 200417036652,577415413723,202675714650,602725007170          03
      ØCT 577614251073,600564153214,202617461335,600431433140          04
      ØCT 203561430354,604424545444,203733461727,605405736545          05
      ØCT 206514623626                                                 06
      DEC 400.                      X3MAX
      DEC 5                         DEG.
      ØCT 175507534121,605757777776,170566373471,602644444444          00
      ØCT 172515241432,575477343155,174640527737,577436207573          01
      ØCT 176770740353,174636643427,601552173243,200766725412          02
      ØCT 200474271721,576527637505,204666170027,604555412253          03
      ØCT 603461004367,201476067310,202732405441,605435033767          04
      ØCT 205555604227,601551553771,202506741765,605526004131          05
      ØCT 206603416461                                                 06
      BSS 20
NCV22 DEC      700.,1500.,3.,8.       F-4976  3 SEC.
      DEC      750.                   X1 MAX.
      DEC      10                     DEG.
      ØCT 174507534121,605717777776,177631463146,602431463146          00
      ØCT 605424600647,204632475371,603543332203,203411751317          01
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
ØCT 174434662447,206436501141,603453762361,203567266571         02
ØCT 605635737414,204536100007,602471253770,576714163663         03
ØCT 603576423717,201757507235,605607676513,201747754776         04
ØCT 202440255610,203460041360,604573647131,205414735704         05
ØCT 603622431140,203452236543,602442531617,603425000150         06
ØCT 202611737653,203517011775,602672252404,203631531447         07
ØCT 602675507771,603603537517,203600032612,203460172341         08
ØCT 603502602431,203477346510,602575771167,200631576021         09
ØCT 605666550122,203652275033,206612662266,604637624710         10
ØCT 605615066573,203623117525,202573631212,600436153253         11
ØCT 577444034575,604740537707,203610740522,205701207061         12
ØCT 603652574001,605466427765,202475734074,204420536636         13
ØCT 603564641263,202577321253,602516057505,611536142123         14
ØCT 206447733125,212615255572,607571250003,612465647001         15
ØCT 207574737362,210565120747,606534177563,176554150100         16
ØCT 605611016714,206564115066                                   17
DEC     950.                    X2 MAX.
DEC     10                      DEG.
ØCT 172507534121,604417777777,177631463146,602431463146
ØCT 201513531717,176501661552,601407661657,601526255172
ØCT 200617607306,602460356372,201765560471,201530005176
ØCT 601702514014,201507433436,602424671656,176515605053
ØCT 202770310701,600746515135,201410062246,577437505357
ØCT 576572761703,602510574016,602412326426,202422760433
ØCT 176523072410,602624350636,200404426013,203767103171
ØCT 600764037414,603460443135,576654031666,177547066666
ØCT 601740401425,201614333370,201573070177,600516436475
ØCT 201660304343,600516721323,576434006307,600503653106
ØCT 205633135364,604425362743,606570217351,205422571466
ØCT 205537213254,604454245117,571773574660,576643500026
ØCT 200761224705,604717405120,604645756677,206463532624
ØCT 205570403367,606442113654,604534124541,204670371166
ØCT 601615507043,202662307711,603602421063,610433150337
ØCT 207441644012,211432545222,610451617104,610537051775
ØCT 207617430366,205776455000,605617437603,203774315134
ØCT 605425173234,206442276737
DEC     1500.                   X3 MAX
DEC     10
ØCT 170734500167,603435056427,177631463146,602431463146         00
ØCT 205426746555,601776276754,604460767404,603452502560         01
ØCT 203414440671,606426561032,203660372506,576653501704         02
ØCT 601715426223,205477020221,603661633443,177754003263         03
ØCT 205402133063,604453764570,205515443614,176776741612         04
ØCT 202564633640,604617254232,601701112011,204401637366         05
ØCT 604665447243,602646764113,601670420031,204776421700         06
ØCT 201532645047,605457103213,203664732030,602563070632         07
ØCT 602413740113,177444502620,202771472365,603461110440         08
ØCT 203434106061,202551576322,603427155516,202606271555         09
ØCT 203705033076,601716522206,604465232231,203507161766         10
ØCT 602723037746,602746311271,204431343620,602630653462         11
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
       ØCT 201416024206,604426705170,601706271262,205455213767             12
       ØCT 202733650656,604733425502,200650410660,202700261167             13
       ØCT 602565315510,202757574755,603622256215,610543667740             14
       ØCT 205670502403,211557654017,606711471123,610753321342             15
       ØCT 206457316415,206741342075,604424463122,602512511317             16
       ØCT 604451365116,205564006641                                       17
NCV23  DEC 100.,1100.,147.,1470.         NCV23 F-2765 BIVAR
       DEC 400.                          X1MAX
       DEC 6                             DEG.
       ØCT 171664720155,601652525252,167614223036,601470707070
       ØCT 175641443452,573647007357,574525602651,570606241736
       ØCT 175456053720,574411412545,170761517423,172412460776
       ØCT 575474162410,575776024220,566665726566,171503677760
       ØCT 171672121336,175412757224,176636036124,400000000000
       ØCT 164615723145,570741667045,571724011272,575504104731
       ØCT 576721676622,155562222220,552411276127,165755417352
       ØCT 571567516211,570472726166,174677042141,201621234227
       DEC 1100.                         X2MAX
       DEC 6                             DEG.
       ØCT 170566373471,602422222222,167614223036,601470707070             00
       ØCT 570636527064,164730051260,564445744151,566563406164             01
       ØCT 167633271772,172740334561,170767214060,166603115354             02
       ØCT 571744101163,573470123570,564472776255,567654413071             03
       ØCT 167533744424,171506002077,172461630124,150604631462             04
       ØCT 165563476671,570527033546,570411027356,166634314313             05
       ØCT 567657463150,153636560506,551657777777,564537573027             06
       ØCT 166720132676,166517564042,566426602015,201547330457             07
       BSS 20
NCV24  DEC 1.2,2.                        F-4971
       DEC 2.                            X1MAX
       DEC 3                             DEG.
       ØCT 202477777776,602777777775
       ØCT 165637751000,567635230000,172620425022,176454547473
NCV25  DEC -4.5945,1.9375                F-4874 UVAR
       DEC -2.297                        X1MAX
       DEC 5                             DEG.
       ØCT 200674557306,202600000000                                   SEC-1 04
       ØCT 175430165544,174435077614,573701420052,174511404000         SEC-1 05
       ØCT 174607261041,172701502446                                   SEC-1 06
       DEC -1.049                        X2MAX
       DEC 5                             DEG.
       ØCT 201630544531,202526417025                                   SEC-2 04
       ØCT 573433757200,172530316300,573571703640,575423615202         SEC-2 05
       ØCT 176744515301,177670715366                                   SEC-2 06
       DEC 0.                            X3MAX
       DEC 5                             DEG.
       ØCT 201747547203,201400000000                                   SEC-3 04
       ØCT 571424424600,172757504040,174552147520,575461477340         SEC-3 05
       ØCT 576706216217,177712421062                                   SEC-3 06
       DEC 1.9375                        X4MAX
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
      DEC 5                        DEG
      ØCT 201407073275,601400000000                            SEC-4 04
      ØCT 572444631631,173662515350,574436255427,174536573136  SEC-4 05
      ØCT 574753077417,174431267623                            SEC-4 06
      BSS 20
NCV26 DEC 0.,1.,200.,400.          F-4875  BIVAR
      DEC 1.                       X1MAX
      DEC 5                        DEG
      ØCT 202400000000,601400000000,172406111564,602431463145        00
      ØCT 176450354223,575634762464,176537057263,170430203445        01
      ØCT 575545565342,175542647372,567616734505,171420275244        02
      ØCT 174441440403,177555624551,171431543213,570542206444        03
      ØCT 571571536433,571431033552,201407257523,170551545653        04
      ØCT 170514467347,571467560575,571706574047,573745557472        05
      ØCT 201673044275                                               06
      BSS 20
NCV27 DEC 1.,1.95                  F-4876  UVAR
      DEC 1.2                      X1MAX
      DEC 5                        DEG.
      ØCT 204500000005,604540000005                            N9S1  04
      ØCT 172465235320,572561446640,170733376620,573406745460  N9S1  05
      ØCT 174531003361,175523334544                            N9S1  06
      DEC 1.95                     X2MAX
      DEC 5                        DEG.
      ØCT 202525252525,603414631462                            N9S2  04
      ØCT 171622330300,567731666600,570466207000,572533067064  N9S2  05
      ØCT 172655460776,176436242341                            N9S2  06
      BSS 20
NCV28 DEC 100.,650.                F-4877  UVAR
      DEC 650.                     X1MAX
      DEC 4                        DEG.
      ØCT 170734500167,601535056427                            IBM10R04
      ØCT 603555516000,606426526140,207474574500,212700370045  IBM10R05
      ØCT 213473275353                                         IBM10R06
NCV29 DEC 470.,2195.               F-4877 REVERSED UVAR
      DEC 2195.                    X1MAX
      DEC 4                        DEG.
      ØCT 167457736603,601613400573                            IBM10R04
      ØCT 602602423400,204715743600,605464122060,211404426512  IBM10R05
      ØCT 211614462612                                         IBM10R06
NCV30 DEC 160.,800.,0.,100.        TD NØ. BIVAR
      DEC 290.                     X1MAX
      DEC 5                        DEG
      ØCT 172770077007,602673047302,173507534121,600777777774  N 1   00
      ØCT 571405446744,173455645313,173643336422,174455620325  N 1   01
      ØCT 574415056147,575456047272,574434656664,574611405715  N 1   02
      ØCT 175445170750,575740021234,574430642754,570550043301  N 1   03
      ØCT 174607271114,575573122627,177601272140,176522367564  N 1   04
      ØCT 571553322076,576741060173,171745077232,176754211671  N 1   05
      ØCT 202747046715                                         N 1   06
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
      DEC 400.                          X2MAX
      DEC 5                             DEG
      ØCT 173451710112,603621350563,173507534121,600777777774  ØN 2  00
      ØCT 171424612441,566516171022,570401247001,573507376276  ØN 2  01
      ØCT 571473342732,172770534614,570751373407,171643125321  ØN 2  02
      ØCT 171712410743,573775317504,570710666720,172543224455  ØN 2  03
      ØCT 173505037077,573674765122,575777762316,175564241373  ØN 2  04
      ØCT 570551261517,576472005227,570607451423,176431474365  ØN 2  05
      ØCT 202770757573                                         ØN 2  06
      DEC 800.                          X3MAX
      DEC 5                             DEG
      ØCT 171507534121,602577777776,173507534121,600777777774        00
      ØCT 172410701102,571461707517,172577223760,170527375352        01
      ØCT 170477612124,574674537256,572441763503,166553773253        02
      ØCT 173554320421,175631744433,172432535342,173404011571        03
      ØCT 572602301751,574561100063,575731653526,173451676161        04
      ØCT 571535537726,573762510364,171556132013,173763627630        05
      ØCT 202703702721                                               06
NCV31 DEC 90.,460.                      F-4885
      DEC 460.                          X1MAX
      DEC 5                             DEG
      ØCT 171542177235,601574424603                            CRV19 04
      ØCT 576550251320,176417477750,175517146700,174550450350  CRV19 05
      ØCT 576544760116,201601233115                            CRV19 06
NCV32 DEC 1.,1.9                   F--4884  UVAR
      DEC 1.9                      X1MAX
      DEC 5                        DEG.
      ØCT 202434343435,602634343434                            CRV18 04
      ØCT 571524254400,567643442000,172611206540,572762251600  CRV18 05
      ØCT 175423331134,200522522257                            CRV18 06
NCV39 DEC .3,1.9                        F-7285 UVAR 1 SEC.
      DEC 1.9
      DEC 5
      ØCT 201500000000,601537777777                              S-MAT.
      ØCT 576523276000,177525376553,577467756500,177577066551    S-MAT.
      ØCT 600512441706,200725137110                              S-MAT.
NCV46 DEC 2.,20.,25.,1000.              F- 10875 BIVAR - ØLD 91 4 SEC.
      DEC 3.65                          X1 MAX
      DEC 5                             DEG.
      ØCT 201466233132,602666233130,170414673152,601415101506  10875100
      ØCT 205644346531,204474223171,202714546430,203632171051  10875101
      ØCT 201675553132,606674013346,604416322330,201726007527  10875102
      ØCT 201565330252,607514217347,604610440734,201420327430  10875103
      ØCT 202665473164,203670601015,207750313232,204537600327  10875104
      ØCT 603723346272,603776366452,202601522770,202763205716  10875105
      ØCT 215763436474                                         10875106
      DEC 6.                            X2 MAX
      DEC 5                             DEG.
      ØCT 200663575230,603406635756,170414673152,601415101506  10875200
      ØCT 604521441204,204446373634,204441023411,603656232362  10875201
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
      ØCT 603521444051,205463270041,606454141212,204746121021  10875202
      ØCT 203766312135,607423442552,606625306222,206601407770  10875203
      ØCT 204743627471,205401675017,610711177130,207623632625  10875204
      ØCT 607523151337,606556721006,205502110442,206573644321  10875205
      ØCT 215752246577                                         10875206
      DEC 11.                      X3 MAX
      DEC 5                        DEG.
      ØCT 177631463146,602663146314,170414673152,601415101506  10875300
      ØCT 605512452743,201651551673,603401321713,604644000511  10875301
      ØCT 205455601603,206604617515,604441450213,203607765555  10875302
      ØCT 603634305375,207420712145,205430565640,202755274451  10875303
      ØCT 600554227151,605736446467,612466415571,207541172576  10875304
      ØCT 607643750560,602766725672,204564430077,206770776247  10875305
      ØCT 215657655700                                         10875306
      DEC 20.                      X4 MAX
      DEC 5                        DEG.
      ØCT 176707070707,602670707070,170414673152,601415101506  10875400
      ØCT 577725436127,601463445310,203425326626,604767200351  10875401
      ØCT 204627150474,605404355714,204522031067,603440224725  10875402
      ØCT 202531127230,207610614666,203461751222,606404051125  10875403
      ØCT 206420276731,604754175757,612430175605,206757732642  10875404
      ØCT 607444165627,605456775625,205421544753,206425734514  10875405
      ØCT 215544275105                                         10875406
NCV55 DEC 600.,800.,274.,400.,800.,5                 F-5092
      ØCT 172507534121,603677777776,173404040404,603526262626  092   00
      ØCT 200477216741,576532775471,576712221221,176462467400  092   01
      ØCT 575631624576,600514674246,600451346503,174433254512  092   02
      ØCT 201476620010,601607664444,200451161052,177500770767  092   03
      ØCT 600457326050,201571036224,204474225220,200674346322  092   04
      ØCT 200764003360,176475744131,575463753000,605510611360  092   05
      ØCT 211440512714                                         092   06
NCV56 DEC 260.,380.,600.,1100.,320.,5                F-4977   SEC 1
      ØCT 174421042104,604465252525,171406111564,602663146312    SEC100
      ØCT 177510245337,201517400414,574766541574,577411555025    SEC101
      ØCT 577450165074,575603157104,603570224243,204465161550    SEC102
      ØCT 603647777600,202527274065,204616556644,605567276363    SEC103
      ØCT 205506470607,605525023423,206413441256,604763203042    SEC104
      ØCT 205576637777,604652147075,205552343532,606763034731    SEC105
      ØCT 210464655776                                           SEC106
      DEC 380.,5                                     F-4977   SEC 2
      ØCT 174421042104,604565252524,171406111564,602663146312    SEC200
      ØCT 201737412367,201663732605,175544064743,603477644464    SEC201
      ØCT 601474631520,201767314455,204473017177,200656317257    SEC202
      ØCT 605572661660,204627130666,202776026665,605614003433    SEC203
      ØCT 206750123774,607403473627,206662721737,605620071073    SEC204
      ØCT 206473144732,606772127632,207663645402,610423366342    SEC205
      ØCT 210727564774                                           SEC206
NCV57 DEC 260.,380.,600.,1100.,320.,5                F-4979   SEC 1
      ØCT 174421042104,604465252525,171406111564,602663146312  9 SEC100
      ØCT 176467024045,174766544474,555642721350,176716666664  9 SEC101
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
       ØCT 575666666644,577465462735,577634545723,200756543425  9 SEC102
       ØCT 601521723353,200754257647,201604311022,602607212106  9 SEC103
       ØCT 202744644476,603554737375,204447276044,602410703043  9 SEC104
       ØCT 202720222300,603403275117,203763521621,605407030752  9 SEC105
       ØCT 206467155061                                         9 SEC106
       DEC 380.,5                                     F-4979   SEC 2
       ØCT 174421042104,604565252524,171406111564,602663146312  9 SEC200
       ØCT 177722436345,574525063334,177437776414,602424333347  9 SEC201
       ØCT 174612761731,200717146473,203634545447,602414614763  9 SEC202
       ØCT 603774132616,203402240774,602652255553,203475353530  9 SEC203
       ØCT 204535615555,605514156767,205406762343,603657430263  9 SEC204
       ØCT 202507303026,602451627244,205556327036,606461125652  9 SEC205
       ØCT 206763730531                                         9 SEC206
CØMMØN BSS     6
 TEST  SYN     CØMMØN
 NTEST SYN     TEST+4
       END
```

FIGURE 101. SYMBOLIC LISTING OF SUBROUTINES BIVAR AND UVAR (cont.)

```
$IBFTC PTESTS  FULIST,REF
      SUBRØUTINE PTEST (TEST,NTEST)
      DIMENSIØN TEST(4), NTEST(2)
C3682PT PTEST SUBRØUTINE TØ TAKE LIMIT ØN ØUT ØF RANGE CURVES
      I=NTEST(1)
      K=NTEST(2)
      GØ TØ (1,1,2,3,3,2,3,3),K
1     TEST(1)=TEST(2)
      GØ TØ 10
2     TEST(3)=TEST(4)
      GØ TØ 10
3     TEST(1)=TEST(2)
      TEST(3)=TEST(4)
10    RETURN
      END
```

FIGURE 102. SYMBOLIC LISTING OF SUBROUTINE PTEST

LXXXII. SUBROUTINE ESTM

This subroutine calculates the value of X which yields Y = 0 for a given function Y = Y(X). Lagrange's interpolation formula is used for first-through-seventh degree interpolation or extrapolation.

CALLING SEQUENCE

The calling sequence is:

CALL ESTM (IDL, IN, XT, YT, YV, XV, LUPNAM).

SOLUTION METHOD

```
1.  IDL = IDL + 1
2.  If IDL ≤ 20, go to 3
    If IDL > 20, go to 20
3.  XT(IDL) = XV
4.  YT(IDL) = YV
5.  If IDL ≤ 1, go to 6
    If IDL > 1, go to 7
```

```
6.  XV = (1.04)(XV)
    Return
7.  SUM = 0
8.  If IDL ≤ (IN + 1), go to 9
    If IDL > (IN + 1), go to 10
9.  IM = 1
    Go to 11
10. IM = IDL - IN
11. Do 18, I = IM, IDL
12. PROD = XT(I)
13. Do 17, J = IM, IDL
14. A = YT(I) - YT(J)
15. If A ≠ 0, go to 16
    If A = 0, go to 17
16. B = (-YT(J)/(A)
17. PROD = (PROD)(B)
18. SUM = SUM + PROD
19. XV = SUM
    Return
20. ERROR PRINT
21. CALL CNTROL(1.0)
```

SUBROUTINE ESTM NOMENCLATURE

Symbol	Description	Units	Reference
A	—	—	14, 15, 16
B	—	—	16, 17
I	—	—	11, 12, 14
IDL	Indicator denoting number of times routine has been entered	—	CALL, 1, 2, 3, 4, 5, 8, 10, 11, 13
IM	—	—	9, 10, 11, 13
IN	Desired degree of fit (1 - 7)	—	CALL, 8, 10
J	—	—	13, 14, 16
LUPNAM	Hollerith word, of not more than six characters, which identifies iteration loop in case of failure to converge. Must be specified as an alphameric word in a DATA statement in the routine which calls ESTM	—	CALL
PROD	—	—	12, 17, 18
SUM	—	—	7, 18, 19
XT	Symbolic name of the array of X's	—	CALL, DIM, 3, 12
XV	X argument	—	CALL, 3, 6, 19
YT	Symbolic name of the array of Y's	—	CALL, DIM, 4, 14, 16
YV	Y argument	—	CALL, 4

```
$IBFTC ESTM    REF,LIST
      SUBROUTINE ESTM  (IDL,IN,XT,YT,YV,XV,LUPNAM)
      DIMENSION XT(20),YT(20)
    2 IDL=IDL+1
      IF(IDL-20)5,5,60
    5 XT(IDL)=XV
      YT(IDL)=YV
      IF(IDL-1)10,10,20
   10 XV=1.04*XV
      RETURN
   20 SUM=0.0
      IF(IDL-(IN+1))30,30,40
   30 IM=1
      GO TO 50
   40 IM=IDL-IN
   50 DO 3 I=IM,IDL
      PROD=XT(I)
      DO12 J=IM,IDL
      A=YT(I)-YT(J)
      IF(A)11,12,11
   11 B=(-YT(J))/A
      PROD=PROD*B
   12 CONTINUE
    3 SUM=SUM+PROD
      XV=SUM
      RETURN
   60 WRITE(6,70)LUPNAM,(I,XT(I),YT(I),I=1,20)
   70 FORMAT(1H1,//24X,20HITERATION FAILED IN A6,1X,4HLOOP//13X,52HTHE I
     1NDEPENDENT AND DEPENDENT VARIABLE TABLES FOLLOW//19X,1HI,13X,2HXT,
     218X,2HYT//20(I20,2E20.8/))
      CALL CNTRØL (1.0)
      STØP
      END
```

FIGURE 103. SYMBOLIC LISTING OF SUBROUTINE ESTM

LXXXIII. SUBROUTINE SWITCH

This subroutine will set one variable equal to another variable.

CALLING SEQUENCE

The subroutine calling sequence is:

```
CALL SWITCH (N, A, B, C, D, E, F, J, L .......)
```

where

N = number of "sets" of parameters

A, C, E, J .... are the variables to be set equal to B, D, F, L ....

$$\left.\begin{array}{l} A, B \\ C, D \\ E, F \\ J, L \end{array}\right\} \text{ are the parameter "sets"}$$

There is no limit to the number of "sets" which may be used in a CALL statement.

```
$IBMAP SWTCH   FULIST,REF
       ENTRY   SWITCH
SWITCH CLA*    3,4
       SXA     SXR,4
       SXA     SXR+1,1
       PAX     ,1
       CLA*    5,4
       STO*    4,4
       TXI     *+1,4,-2
       TIX     *-3,1,1
 SXR   AXT     **,4
       AXT     **,1
       TRA     1,4
       END
```

FIGURE 104. SYMBOLIC LISTING OF SUBROUTINE SWITCH

LXXXIV. SUBROUTINE PU

The purpose of this subroutine is to find the propellant utilization (PU) valve flow loss coefficient as a function of the PU valve angle. The curve may be segmented into as many as ten parts each of which is represented by a fifth degree polynomial curve fit.

CALLING SEQUENCE

The calling sequence is:

CALL PU (A, C, X, Y)

SOLUTION METHOD

Set PU valve flow loss coefficient to zero.

1. Y = 0

Search for correct curve segment depending upon the value of the PU valve angle.

2. Do 5 I = 1, 10

3. K = I

4. If $X \leq C(I)$, go to 6

If $X > C(I)$, go to 5

5. Continue

Compute PU valve flow loss coefficient.

6. $Y = A(1, K) + (A(2,K))(X) + (A(3, K))(X^2)$

$+ (A(4, K))(X^3) + (A(5, K))(X^4) + (A(6, K))(X^5)$

7. Return

SUBROUTINE PU NOMENCLATURE

Symbol	Description	Units	Reference
A	Array of polynomial coefficients in ascending order	—	CALL, DIM, 6
C	Array of curve segment limits	—	CALL, DIM, 4
I	Section locater	—	2, 3, 4
K	Locates coefficients for a given section	—	3, 6
X	PU valve angle	degrees	CALL, 4, 6
Y	PU valve flow loss coefficient	$\frac{sec^2}{ft^3-in^2}$	CALL, 1, 6

```
$IBFTC SØLUT   REF,LIST
      SUBRØUTINE PU(A,C,X,Y)
      DIMENSIØN A(6,10) ,C(10)
      Y=0
C     SEARCH TØ FIND WHICH SECTIØN X LIES IN
      DØ 20I=1,10
      K=I
      IF (X .LE. C(I)) GØ TØ 30
   20 CØNTINUE
C     CALCULATE DEPENDENT VARIABLE
   30 Y=A(1,K) +A(2,K)*X+A(3,K)*X**2+A(4,K)*X**3+A(5,K)*X**4+A(6,K)*X**5
      RETURN
      END
```

FIGURE 105. SYMBOLIC LISTING OF SUBROUTINE PU

LXXXV. SUBROUTINE EPRINT

The purpose of this subroutine is to print the input data to the engine balance/rebalance programs (Block I) and to print the output from these programs.

CALLING SEQUENCE

The calling sequence is:

CALL EPRINT (J, JA)

SOLUTION METHOD

Determine whether to print input or output data.

1. If JA = 0, go to 28

   If JA ≠ 0, go to 2

Determine whether to print balance or rebalance heading.

2. If KBAL = 1, go to 10

   If KBAL = 2, go to 3

Determine heading to be used on output data.

```
3.  If J = 1, go to 4
    If J = 2, go to 6
    If J = 3, go to 8
    If J = 4, go to 21
4.  WRITE  rebalance case heading for engine H-1
5.  Go to 17
6.  WRITE  rebalance case heading for engine F-1
7.  Go to 17
8.  WRITE  rebalance case heading for engine J-2
9.  Go to 61
```

Determine heading to be used on output data.

```
10. If J = 1, go to 11
    If J = 2, go to 13
    If J = 3, go to 15
    If J = 4, go to 21
11. WRITE  balance case heading for engine H-1
12. Go to 17
13. WRITE  balance case heading for engine F-1
14. Go to 17
15. WRITE  balance case heading for engine J-2
16. Go to 61
```

H-1 and F-1 Engine Output Data Printout

```
17.  WRITE  FE, XISPE, WEO, WEF, XMRE, FC, PNS, PCIE,
            XMRTC, WC, WCO, WCF, CS, CSTHE, CF,
            CFTHE

18.  WRITE  WP, QP, XNP, H, PPD, ETAP, BHPP, XNPSH,
            PPIS, RHPI, WT, PTI, PTES, TTI, CST, CPT,
            GAMT, RGT, XNT, UVRAT, ETAT, HPT

19.  WRITE  PGG, WG, WGO, WGF, ORES, XMRG, RESOG,
            RESFG

20.  Return
```

RL-10 Engine Output Data Printout and Heading

```
21.  If KBAL = 1, go to 24

     If KBAL = 2, go to 22

22.  WRITE  rebalance case heading

23.  Go to 25

24.  WRITE  balance case heading

25.  WRITE  KBAL, FC, PPI, XMRTC, Q, PTI(1), XNP,
            RHPI, TPIS, UVRAT(1), ETAT(1)

26.  WRITE  XISPE, HP2, HP1, PCIE, CS, WC, WCO, WCF,
            CSTHE, CF, ETCF, CFTHE, TOJE, ETCS, H,
            PPD, ETAP, PPIS, CON3, PNS, WT, HPT(1),
            PNSPR, FC

27.  Return
```

Determine heading for printout of input data.

```
28.  If J = 1, go to 29
     If J = 2, go to 31
     If J = 3, go to 33
     If J = 4, go to 50
29.  WRITE  H-1 engine input data heading
30.  Go to 34
31.  WRITE  F-1 engine input data heading
32.  Go to 34
33.  WRITE  J-2 engine input data heading
```

Determine whether to print out H-1, F-1, J-2 or RL-10 data.

```
34.  If J = 1, go to 35
     If J = 2, go to 35
     If J = 3, go to 55
     If J = 4, go to 50
```

<u>H-1 and F-1 Input Data Print</u>

```
35.  WRITE  KBAL, INDPR, FEN, DEL7, XMREN, DEL9, PA,
            TN, PPI, TPIR, TRSG, SG
36.  WRITE  C7, DPI, WMIS, C3, C4, C5, C6
37.  WRITE  DRI, DRD, DEL4, DEL5, DLTR, R
```

Determine heading for second page of input printout.

```
38.  If J = 1, go to 39
     If J = 2, go to 41
     If J = 3, go to 64
     If J = 4, go to 64
39.  WRITE  H-1 engine heading
40.  Go to 42
41.  WRITE  F-1 engine heading
42.  WRITE  DPD, DEL1, DEL2, DEL3, DLTP, RGG, RX, C1
43.  WRITE  PTI, PTES, ATI, DW, IPB, GR, XNTST, ETAGR,
            DEL6, ATE
44.  If J = 1, go to 45
     If J = 2, go to 48
     If J = 3, go to 64
     If J = 4, go to 64
45.  WRITE  DEL8, ETCF, EPS, PNS, CON, ETCS, AXD,
            DELTX
46.  WRITE  ATC, CPCX, FX, XMRTC, HPAUX
47.  Return
48.  WRITE  DEL8, ETCF, EPS, PNS, CON, ETCS, CPCX,
            XMRTC, ATC
49.  Return
```

RL-10 Engine Input Data Printout and Heading

```
50.  WRITE  RL-10 engine heading
51.  WRITE  KBAL, FCL, XMRTC, ATC, PPI, TPIS
52.  WRITE  XNPN, XNP, RHPI, Q, HPAUX, PCIEL, TOL1
53.  WRITE  TOL2, TOL3, TOL4, TOL5, TOL6, TOL7, TOL8, TOL9, TOL10, CON1, CON2, CON3, CON4, CON5, CON6, CON7, CON8, CON9, CON10, CON11, CON12, CON13, CON14, CON15, CD2, CON, AVT, ETGER, EPS
54.  Return
```

J-2 Engine Input Data Printout

```
55.  WRITE  KBAL, INDPR, FEN, DEL7, XMREN, PA, TTI, PUVA, PPI, TPIS, TRSG, SG
56.  WRITE  C1, C3, C4, C5, C6, C8, PUCOF, PUSEC, DEL1, DEL2, FAUX, ETCF
57.  WRITE  DEL4, DEL5, DRI
58.  WRITE  second page heading for J-2 engine
59.  WRITE  DRD, DLTR, R, DW, ATE, DPI, ATI, DLTP, DPD, IPB, DEL3, PTI, CNST1, DEL20, DEL21, DEL22, XMRTC, PNS, EPS, ETCS, ATC, CNST2, CNST3, DMOO, DMFO, WPRESF, WPRESO, HPAUX, CNST4, CNST5, CNST6, CNST7, CNST8, DEL23, DEL24, DEL25, DEL26, DEL27, DEL28, DEL29, DEL35, ABPNI, EPSM, CON
```

```
60.  Return
```

J-2 Engine Output Data Printout

```
61.  WRITE  FE, XISPE, WEO, WEF, XMRE, FC, PNS, PCIE, XMRTC, WCO, WCF, CS, CSTHE, CF, CFTHE, WC
62.  WRITE  WP, QP, XNP, H, PPD, ETAP, BHPP, XNPSH, PPIS, RHPI, WT, PTI, PTES, TTI, CPT, GAMT, RGT, XNT, UVRAT, ETAT, HPT, PRT
63.  WRITE  PGG, WG, WGO, WGF, DBPN, XMRG, RESOG, RESFG, CSG, WPUO, WBP
64.  Return
```

SUBROUTINE EPRINT NOMENCLATURE

Symbol	Description	Units	Reference
ATC	Thrust chamber throat area	in^2	/CCHAM/, 46, 48, 51, 59
ATE	Turbine exit flange area	in^2	/CTURB/, 43, 59
ATI	Turbine inlet nozzle area	in^2	/CTURB/, 43, 59
AVT	Venturi throat area	in^2	/FLRL10/, 53
AXD	Turbine exhaust discharge area	in^2	/CTURBX/, 45
BHPP	Pump brake horsepower	bhp	/CPUMP/, 18, 62
C1	Array of coefficients for pump performance	—	/CPUMP/, 42, 56
C3	Array of coefficients for specific heat ratio curve	—	/CPUMP/, 36, 56
C4	Array of coefficients for specific heat curve	—	/CGROP/, 36, 56
C5	Array of coefficients for molecular weight curve	—	/CGROP/, 36, 56
C6	Array of coefficients for gas generator mixture ratio curve	—	/CGROP/, 36, 56
C7	Array of coefficients for turbine efficiency curve	—	/CTURB/, 36
C8	Array of coefficients for turbine performance map	—	/CJ2IN/, 56
CD2	Lox injector flow loss coefficient	—	/LXRL10/, 53

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
CF	Thrust chamber thrust coefficient	—	/CCHAM/, 17, 26, 61
CFTHE	Thrust chamber theoretical thrust coefficient	—	/CCHAM/, 17, 26, 61
CNST1	Ratio of thrust chamber injector end pressure to thrust chamber nozzle stagnation pressure	lb/in^2	/CJ2IN/, 59
CNST2	Constant used in calculating pressure drop from fuel injector face to fuel pressurization bleed line	sec/in^2	/CJ2IN/, 59
CNST3	Constant used in calculating pressure drop from fuel pressurization bleed line to thrust chamber manifold inlet	sec/in^2	/CJ2IN/, 59
CNST4	First guess of fuel turbine exit static pressure	lb/in^2	/CJ2IN/, 59
CNST5	First guess of oxidizer turbine exit static pressure	lb/in^2	/CJ2IN/, 59
CNST6	First guess of by-pass nozzle inlet Mach number	—	/CJ2IN/, 59
CNST7	First guess of by-pass nozzle mass flow rate	lb/sec	/CJ2IN/, 59
CNST8	First guess of oxidizer turbine inlet total pressure	lb/in^2	/CJ2IN/, 59

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
CON	First guess of ratio of turbine mass flow rate to total fuel mass flow rate	—	/INCON/, 45, 48, 53, 59
CON1	Constant used in calculating lox pump inlet static pressure	$\frac{sec^2}{ft^3 - in^2}$	/LXRL10/, 53
CON2	Flow loss coefficient for line from lox pump discharge to injector inlet	$\frac{sec^2}{ft^3 - in^2}$	/LXRL10/, 53
CON3	Ground mixture ratio control valve flow loss coefficient	$\frac{sec^2}{ft^3 - in^2}$	/LXRL10/, 26, 53
CON4	Actual lox injector area	in^2	/LXRL10/, 53
CON5	Ratio of lox pump speed to fuel pump speed	—	/LXRL10/, 53
CON6	Constant used in calculating fuel pump inlet static pressure	$\frac{sec^2}{ft^3 - in^2}$	/FLRL10/, 53
CON7	Flow loss coefficient for line from fuel pump discharge to jacket inlet manifold	$\frac{sec^2}{ft^3 - in^2}$	/FLRL10/, 53
CON8	Flow loss coefficient for fuel pump discharge orifice	$\frac{sec^2}{ft^3 - in^2}$	/FLRL10/, 53
CON9	First guess of ratio of turbine mass flow rate to total fuel mass flow rate	—	/TBRL10/, 53

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
CON10	Flow loss coefficient for line from jacket exit manifold to venturi inlet	$\frac{sec^2}{ft^3 - in^2}$	/FLRL10/, 53
CON11	Constant used in computing mass flow rate	—	/TBRL10/, 53
CON12	Ratio of turbine exit static pressure to turbine exit total pressure	—	/TBRL10/, 53
CON13	Flow loss coefficient for line from turbine exit to injector inlet	$\frac{sec^2}{in^4 - {}^\circ R}$	/TBRL10/, 53
CON14	Flow loss coefficient for main fuel shut-off valve	$\frac{-sec^2}{in^2 - {}^\circ R}$	/TBRL10/, 53
CON15	Fuel injector area	in^2	/TBRL10/, 53
CPCX	Ratio of thrust chamber wall pressure to nozzle stagnation pressure	—	/CTURBX/, 46, 48
CPT	Turbine gas specific heat	$\frac{btu}{lb - {}^\circ R}$	/CTURB/, 18, 62
CSG	Gas characteristic velocity	ft/sec	/CGROP/, 63
CS	Thrust chamber characteristic velocity	ft/sec	/CCHAM/, 17, 26, 61
CSTHE	Thrust chamber theoretical characteristic velocity	ft/sec	/CCHAM/, 17, 26, 61
CST	Turbine gas inlet characteristic velocity	ft/sec	/CTURB/, 18

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DBPN	By-pass nozzle diameter	in	/CJ2OUT/, 63
DEL1	Pump inlet static pressure tolerance	psi	/CPUMP/, 42, 56
DEL2	Pump discharge static pressure tolerance	psi	/CPUMP/, 42, 56
DEL3	Pump speed tolerance	rpm	/CPUMP/, 42, 59
DEL4	Resistance inlet static pressure tolerance	psi	/CRES/, 37, 57
DEL5	Resistance exit density tolerance	lb/ft^3	/CRES/, 37, 57
DEL6	Turbine horsepower tolerance	bhp	/CMAIN/, 43
DEL7	Engine thrust tolerance	lb	/CCHAM/, 35, 55
DEL8	Turbine exit static pressure tolerance	psi	/H1F1J2/, 45, 48
DEL9	Engine mixture ratio tolerance	—	/H1F1J2/, 37
DEL20	Mixture ratio tolerance	—	/CJ2IN/, 59
DEL21	Oxidizer pump discharge static pressure tolerance	—	/CJ2IN/, 59
DEL22	Oxidizer pump inlet static pressure tolerance	—	/CJ2IN/, 59
DEL23	Fuel turbine horsepower tolerance	—	/CJ2IN/, 59

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DEL24	Oxidizer turbine horsepower tolerance	—	/CJ2IN/, 59
DEL25	Turbine exhaust manifold total pressure tolerance	—	/CJ2IN/, 59
DEL26	By-pass nozzle inlet mass flowrate parameter tolerance	—	/CJ2IN/, 59
DEL27	Fuel turbine inlet total pressure tolerance	—	/CJ2IN/, 59
DEL28	A^* equation tolerance	—	/CJ2IN/, 59
DEL29	Oxidizer turbine inlet total pressure tolerance	—	/CJ2IN/, 59
DEL35	Oxidizer turbine mass flowrate tolerance	—	/CJ2IN/, 59
DELTX	Turbine exhaust system temperature drop	°R	/CTURBX/, 45
DLTP	Pump propellant temperature rise	°F	/CPUMP/, 42, 59
DLTR	Liquid temperature change across resistance	°F	/CRES/, 37, 59
DMFO	Main fuel orifice diameter	in	/CJ2IN/, 59
DMOO	Main oxidizer orifice diameter	in	/CJ2IN/, 59
DPD	Pump discharge flange diameter	in	/CPUMP/, 42, 59

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DPI	Pump inlet flange diameter	in	/CPUMP/, 36, 59
DRD	Resistance exit diameter	in	/CRES/, 37, 59
DRI	Resistance inlet diameter	in	/CRES/, 37, 57
DW	Turbine wheel diameter	in	/CTURB/, 43, 59
EPSM	Thrust chamber nozzle expansion ratio at turbine exhaust manifold	—	/CJ2IN/, 59
EPS	Thrust chamber expansion area ratio	—	/CCHAM/, 45, 48, 53, 59
ETAGR	Turbine-to-pump gearing efficiency	—	/CMAIN/, 43
ETAP	Pump efficiency	—	/CPUMP/, 18, 26, 62
ETAT	Turbine efficiency	—	/CTURB/, 18, 25, 62
ETCF	Thrust chamber nozzle efficiency	—	/CCHAM/, 26, 45, 48, 56
ETCS	Thrust chamber combustion efficiency	—	/CCHAM/, 26, 45, 48, 59
ETGER	Gear box efficiency	—	/TBRL10/, 53
FAUX	Auxiliary engine thrust	lb	/CCHAM/, 56
FCL	Loaded value of thrust	lb	/CHRL10/, 51
FC	Thrust chamber thrust	lb	/CCHAM/, 17, 25, 26, 61
FEN	Nominal engine thrust	lb	/CCHAM/, 35, 55

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
FE	Engine thrust	lb	/CMAIN/, 17, 61
FX	Turbine exhaust thrust	lb	/CTURBX/, 46
GAMT	Turbine gas specific heat	btu/lb - °R	/CTURB/, 18, 62
GR	Turbine-to-pump gear ratio	—	/CMAIN/, 43
HP1	Fuel pump horsepower	bhp	/TBRL10/, 26
HP2	Oxidizer pump horsepower	bhp	/TBRL10/, 26
HPAUX	Auxiliary drive horsepower	bhp	/CMAIN/, 46, 52, 59
HPT	Turbine horsepower	bhp	/CTURB/, 18, 26, 62
H	Pump head	ft	/CPUMP/, 18, 26, 62
INDPR	Propellant combination indicator	—	/CCHAM/, 35, 55
IPB	Pump speed balance indicator	—	/CPUMP/, 43, 59
J	Indicator to determine engine data printout J = 1; H-1 engine J = 2; F-1 engine J = 3; J-2 engine J = 4; RL-10 engine	—	CALL, 3, 10, 28, 34, 38, 44
JA	Indicator to determine printout of input or output JA = 0; Input JA = 1; Output	—	CALL, 1

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
KBAL	Engine balance/rebalance indicator	—	/CBAL/, 2, 21, 25, 35, 51, 55
ORES	Main line control orifice resistance	sec^2/ft^5	/CMAIN/, 19
PA	Ambient pressure	psia	/CPA/, 35, 55
PCIE	Thrust chamber injector end pressure	psia	/CMAIN/, 17, 26, 61
PCIEL	Thrust chamber injector end pressure	psia	/MNRL10/, 52
PGG	Gas generator injector end pressure	psia	/CMAIN/, 19, 63
PNSPR	Value of thrust chamber nozzle stagnation pressure for use in Subroutine INFL	psia	/PNSPRI/, 26
PNS	Thrust chamber nozzle stagnation pressure	psia	/CCHAM/, 17, 26, 45, 48, 59, 61
PPD	Pump discharge total pressure	psia	/CPUMP/, 18, 26, 62
PPI	Pump inlet total pressure	psia	/CPUMP/, 25, 35, 51, 55
PPIS	Pump inlet static pressure	psia	/CPUMP/, 18, 26, 62
PRT	Turbine pressure ratio	—	/CTURB/, 62
PTES	Turbine exit static pressure	psia	/CTURB/, 18, 43, 62
PTI	Turbine inlet total pressure	psia	/CTURB/, 18, 25, 43, 59, 62

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PUCOF	Array of coefficients for PU valve flow loss coefficient curve	—	/CJ2IN/, 56
PUSEC	Array of PU valve flow loss coefficient curve segment limits	—	/CJ2IN/, 56
PUVA	Propellant utilization valve angle	deg	/CJ2IN/, 55
Q	Pump inlet volume flow rate	ft^3/sec	/MNRL10/, 25, 52
QP	Pump volumetric flow rate	gpm	/CPUMP/, 18, 62
RESFG	Gas generator fuel-line control orifice resistance	sec^2/ft^5	/CRESGG/, 19, 63
RESOG	Gas generator oxidizer-line control orifice resistance	sec^2/ft^5	/CRESGG/, 19, 63
RGG	Gas generator resistance	sec^2/ft^5	/H1F1J2/, 42
RGT	Turbine gas constant	ft/°R	/CTURB/, 18, 62
RHPI	Pump inlet density	lb/ft^3	/CPUMP/, 18, 25, 52, 62
R	Resistance value	sec^2/ft^5	/CRES/, 37, 59
RX	Turbine exhaust system resistance	sec^2/ft^5	/CTURBX/, 42
SG	Liquid specific gravity at reference temperature	—	/CDEN/, 35, 55

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TN	Nominal gas generator combustion temperature	°F	/CMAIN/, 35
TOJE	Jacket exit total temperature	°R	/FLRL10/, 26
TOL1	Fuel pump discharge constant pressure specific heat tolerance	—	/PPRL10/, 52
TOL2	Oxidizer pump discharge constant pressure specific heat tolerance	—	/PPRL10/, 53
TOL3	Lox injector flow loss coefficient tolerance	—	/LXRL10/, 53
TOL4	Fuel pump inlet static pressure tolerance	—	/FLRL10/, 53
TOL5	Injector end chamber pressure tolerance	—	/LXRL10/, 53
TOL6	Fuel mass flow rate tolerance	—	/FLRL10/, 53
TOL7	Mixture ratio tolerance	—	/FLRL10/, 53
TOL8	Chamber flow rate tolerance	—	/CHRL10/, 53
TOL9	Thrust tolerance	—	/CHRL10/, 53
TOL10	Mass flow parameter tolerance	—	/TBRL10/, 53
TPIR	Pump inlet propellant temperature	°R	/CMAIN/, 35

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TPIS	Pump inlet propellant temperature	°F	/CPUMP/, 25, 51, 55
TRSG	Liquid reference temperature	°F	/CDEN/, 35, 55
TTI	Turbine inlet total temperature	°R	/CTURB/, 18, 55, 62
UVRAT	Turbine velocity ratio	—	/CTURB/, 18, 25, 62
WBP	Oxidizer turbine by-pass mass flow rate	lb/sec	/CJ2OUT/, 63
WCF	Thrust chamber fuel flow rate	lb/sec	/CCHAM/, 17, 26, 61
WCO	Thrust chamber oxidizer flow rate	lb/sec	/CCHAM/, 17, 26, 61
WC	Thrust chamber total flow rate	lb/sec	/CCHAM/, 17, 26, 61
WEF	Engine fuel flow rate	lb/sec	/CMAIN/, 17, 61
WEO	Engine oxidizer flow rate	lb/sec	/CMAIN/, 17, 61
WGF	Gas generator fuel flow rate	lb/sec	/CMAIN/, 19, 63
WGO	Gas generator oxidizer flow rate	lb/sec	/CMAIN/, 19, 63
WG	Gas generator total flow rate	lb/sec	/CMAIN/, 19, 63
WMIS	Miscellaneous flow rate	lb/sec	/CMAIN/, 36
WP	Pump mass flow rate	lb/sec	/CPUMP/, 18, 62

SUBROUTINE EPRINT NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WPRESF	Pressurization fuel mass flow rate	lb/sec	/CJ2IN/, 59
WPRESO	Pressurization oxidizer mass flow rate	lb/sec	/CJ2IN/, 59
WPUO	Propellant utilization oxidizer mass flow rate	lb/sec	/CJ2OUT/, 63
WT	Turbine gas flow rate	lb/sec	/CTURB/, 18, 26, 62
XISPE	Engine specific impulse.	sec	/CMAIN/, 17, 26, 61
XMREN	Nominal engine mixture ratio	—	/H1F1J2/, 35, 55
XMRE	Engine mixture ratio	—	/CMAIN/, 17, 61
XMRG	Gas generator mixture ratio	—	/CGROP/, 19, 63
XMRTC	Thrust chamber mixture ratio	—	/CCHAM/, 17, 25, 46, 48, 51, 59, 61
XNP	Pump speed	rpm	/CPUMP/, 18, 25, 52, 62
XNPN	Nominal pump speed	rpm	/PPRL10/, 52
XNPSH	Pump net positive suction head	ft	/CPUMP/, 18, 62
XNT	Turbine speed	rpm	/CTURB/, 18, 62
XNTST	Number of turbine pressure stages	—	/CTURB/, 43

```
$IBFTC EPRNTS  FULIST,REF
      SUBRØUTINE EPRINT (J,JA)
      CØMMØN /CJ2ØUT/ RPU,PV,WPUØ,PMAN,PMAN1,TMAN,TAVE,WBP,
     1PBNI,FPL,FPR,XMBPNI,ABPN,DBPN,PTIS(2),TTIS(2)
      CØMMØN /CTURMP/ PP(2), C8(4,2)
      CØMMØN /CJ2IN/ EL20,DEL21,DEL22,DEL23, EL24,DEL25,DEL26,CNST1,
     1CNST2,CNST3,CNST4,CNST5,CNST6,PUCØF(6,10),PUSEC(10),PUVA
                 3, CNST8,DEL35
      CØMMØN /CRES/ PRI(20),TRIS(20),PVRI(20),RHRI(20),VHRI(20),WRD(20),
     1DRI(20),DEL4(20),TRDS(20),DLTR(20),RHRD(20),RHRA(20),PRD(20),
     2R(20),VHRD(20),DRD(20),PRDS(20),PVRD(20),DEL5(20),WRI(20),PRIS(20)
      CØMMØN/CDEN/TRSG(2),SG(2)
      CØMMØN /CTURBX/ TXD,TXI,DELTX,PXDS,PXIS,RX,RGX,TXIS,WX,CSXD,XØMX,
     1GAMX,XMXD, PRXD,PXD,AXD,FX,TXDS,CPCX
      CØMMØN /CBAL/KBAL /CPA/PA /CRESGG/RESØG,RESFG
      CØMMØN /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
      CØMMØN /CGPRØP/      C3(6),C4(6),C5(6),C6(6),CSG,GAMG,CPG,XMWG,
     1XMRG,RG,XØMG
      CØMMØN /LXRL10/ TØL3,TØL5,CD2,CØN1,CØN2,CØN3,CØN4,CØN5
      CØMMØN /FLRL10/ CØN8,CØN6,TØL4,FLØPT,CØN7,TØJE,WCF1,TØL6,TØL7,
     1CØN10,AVT
      CØMMØN /CHRL10/  TØL8,TØL9,FCL
      CØMMØN /PPRL10/        XRATIØ(2),TØL1,TØL2,XNPN(2)
      CØMMØN /TBRL10/ CØN9,ETGER,CØN11,CØN12,CØN13,CØN14,CØN15,TØL10
     1,HP1,HP2
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),     PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
      CØMMØN /CCHAM/  INDPR ,CF ,CFTHE,ETCF,EPS,PNS,FC,ATC,FAUX,FEN,
     1DEL7,CS,CSTHE,ETCS,WC,XMRTC,WCF,WCØ
      CØMMØN/H1F1J2/RGG,DEL8,DEL9,XMREN
      CØMMØN /CMAIN/WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2), TN,
     1ØRES(4),WG,WGØ,WGF,PCIE,XMRE,PGG,XISPC,XISPE,FE,WEØ,WEF,TPIR(2)
      CØMMØN /MNRL10/ PCIEL, Q(2),XMRTCL,PII(2)
      CØMMØN /INCØN/ CØN
      CØMMØN /PNSPRI/ PNSPR
      IF (JA .EQ. 0) GØ TØ 100
      GØ TØ (9,1) ,KBAL
    1 GØ TØ ( 2,4,6,70) ,J
    2 WRITE (6,3)
    3 FØRMAT ( 1H1///56X, 20HH1 ENGINE REBALANCE ///)
      GØ TØ 40
    4 WRITE (6,5)
    5 FØRMAT ( 1H1/// 56X, 20HF1 ENGINE REBALANCE   ///)
      GØ TØ 40
```

FIGURE 106.   SYMBOLIC LISTING OF SUBROUTINE EPRINT

```
    6 WRITE (6,7 )
    7 FØRMAT ( 1H1 /// 56X, 20HJ2 ENGINE REBALANCE   ///)
      GØ TØ 250
    9 GØ TØ (10,20,30, 70),J
   10 WRITE (6,15)
   15 FØRMAT(1H1///52X,27HH1 ENGINE NØMINAL BALANCE  ///)
      GØ TØ 40
   20 WRITE (6,25)
   25 FØRMAT(1H1///52X,27HF1 ENGINE NØMINAL BALANCE  ///)
      GØ TØ 40
   30 WRITE (6,35)
   35 FØRMAT(1H1///52X,27HJ2 ENGINE NØMINAL BALANCE  ///)
      GØ TØ 250
   40 WRITE (6,45)FE,XISPE,WEØ,WEF,XMRE,FC,PNS,PCIE,XMRTC,WC,WCØ,WCF,CS,
     1CSTHE,CF,CFTHE
   45 FØRMAT( 9X,8HFE      = 1PE16.8,5X,8HXISPE  = 1PE16.8, 5X,8HWEØ    =
     4 1PE16.8, 5X,8HWEF    = 1PE16.8 / 9X,8HXMRE   = 1PE16.8 ,5X, 8HFC
     B    = 1PE16.8 ,5X, 8HPNS    = 1PE16.8 ,5X, 8HPCIE   = 1PE16.8 / 9
     CX, 8HXMRTC = 1PE16.8, 5X, 8HWC     = 1PE16.8 ,5X, 8HWCØ    = 1PE1
     D6.8, 5X, 8HWCF    = 1PE16.8/  9X, 8HCS     = 1PE16.8 ,5X, 8HCSTHE
     E = 1PE16.8, 5X, 8HCF     = 1PE16.8,5X, 8HCFTHE  = 1PE16.8 )
      WRITE (6,55) WP,QP,XNP,H,PPD,ETAP,BHPP,XNPSH,PPIS,RHPI,WT,PTI,
     1 PTES,TTI,CST,CPT,GAMT,RGT,XNT,UVRAT,ETAT,HPT
   55 FØRMAT( 9X,8HWP     = 1PE16.8,12X, 1H= 1PE16.8 , 5X, 8HQP     = 1P
     2E16.8, 12X, 1H= 1PE16.8 / 9X,8HXNP    = 1PE16.8 ,12X , 1H= 1PE16.8
     3, 5X, 8HH      = 1PE16.8 ,12X, 1H= 1PE16.8  / 9X, 8HPPD    = 1PE16
     4.8 ,12X, 1H= 1PE16.8, 5X,8HETAP   = 1PE16.8, 12X, 1H= 1PE16.8 / 9X
     5, 8HBHPP   = 1PE16.8 ,12X, 1H= 1PE16.8, 5X, 8HXNPSH  = 1PE16.8 ,
     612X, 1H= 1PE16.8 / 9X, 8HPPIS   = 1PE16.8,12X, 1H= 1PE16.8,  5X,8H
     7RHPI   = 1PE16.8 , 12X, 1H= 1PE16.8 / 9X, 8HWT     = 1PE16.8 ,12X,
     8 1H= 1PE16.8, 5X, 8HPTI    = 1PE16.8 , 12X, 1H= 1PE16.8 / 9 X,
     98HPTES   = 1PE16.8, 12X, 1H= 1PE16.8, 5X, 8HTTI    = 1PE16.8, 12X,
     11H= 1PE16.8 /  9X, 8HCST    = 1PE16.8, 12X, 1H= 1PE16.8,  5X,8HCPT
     2    = 1PE16.8 ,12X, 1H= 1PE16.8/  9X,8HGAMT   = 1PE16.8, 12X, 1H=
     31PE16.8 ,5X, 8HRGT    = 1PE16.8, 12X, 1H= 1PE16.8/  9X, 8HXNT    =
     4 1PE16.8, 12X, 1H= 1PE16.8,5X, 8HUVRAT  = 1PE16.8, 12X, 1H= 1PE16.
     58 / 9X, 8HETAT   = 1PE16.8 ,12X, 1H= 1PE16.8 ,5X, 8HHPT    = 1PE16
     6.8 ,12X, 1H= 1PE16.8 )
      WRITE (6,65) PGG,WG,WGØ,WGF,ØRES,XMRG,RESØG ,RESFG
   65 FØRMAT ( 9X, 8HPGG    = 1PE16.8 ,5X, 8HWG     = 1PE16.8 ,5X, 8HWGØ
     1    = 1PE16.8 ,5X, 8HWGF    = 1PE16.8 / 9X, 8HØRES   = 1PE16.8,12
     2X, 1H= 1PE16.8, 12X, 1H= 1PE16.8,12X, 1H= 1PE16.8 / 9X, 8HXMRG   =
     3 1PE16.8,5X, 8HRESØG  = 1PE16.8, 5X, 8HRESFG  = 1PE16.8)
      RETURN
   70 GØ TØ (73, 71) ,KBAL
   71 WRITE (6,72)
   72 FØRMAT ( 1H1 ///54X, 22HRL10 ENGINE REBALANCE   ///)
      GØ TØ 75
   73  WRITE (6,74)
   74 FØRMAT ( 1H1 ///51X, 29HRL10 ENGINE NØMINAL BALANCE    ///)
```

FIGURE 106. SYMBOLIC LISTING OF SUBROUTINE EPRINT (cont.)

```
  75 WRITE (6,76) KBAL,FC,PPI,XMRTC,Q,PTI(1),XNP,RHPI,TPIS, UVRAT(1),
    1 ETAT(1)
  76 FØRMAT ( 9X,8HKBAL    =  I3 , 18X,8HFC      = 1PE16.8, 5X,
    A 8HPPI     = 1PE16.8, 12X, 1H= 1PE16.8 /  9X, 8HXMRTC  = 1PE16.8,5X
    B ,8HQ       = 1PE16.8 ,12X, 1H= 1PE16.8, 5X, 8HPTI     = 1PE16.8 /
    C9X,8HXNP     = 1PE16.8,12X,1H= 1PE16.8,5X,8HRHPI    = 1PE16.8,12X,1H
    D= 1PE16.8/9X,8HTPIS    = 1PE16.8,12X,1H= 1PE16.8,5X,8HUVRAT   = 1PE1
    E6.8,5X, 8HETAT    = 1PE16.8    )
     WRITE (6, 78)  XISPE,HP2, HP1,PCIE,CS,WC,WCØ,WCF,CSTHE,CF,ETCF,
    1 CFTHE,TØJE,ETCS,H,PPD,ETAP,PPIS,CØN3,PNS,WT,HPT(1),PNSPR,FC
  78 FØRMAT  (  9X,8HXISPE   = 1PE16.8,5X, 8HHP2     = 1PE16.8, 5X, 8HHP1
    A     = 1PE16.8, 5X, 8HPCIE    = 1PE16.8/
    B                           9X, 8HCS      = 1PE16.8 ,5X, 8HWC     =  1PE1
    C6.8, 5X, 8HWCØ     = 1PE16.8 , 5X,8HWCF     = 1PE16.8 / 9X, 8HCSTHE
    D = 1PE16.8 ,5X , 8HCF      = 1PE16.8, 5X, 8HETCF    = 1PE16.8 ,5X,
    E8HCFTHE  = 1PE16.8 / 9X, 8HTØJE    = 1PE16.8,5X,8HETCS    = 1PE16.8,
    F5X,8HH       =1PE16.8,12X,1H= 1PE16.8 / 9X, 8HPPD     = 1PE16.8 ,12X
    G, 1H= 1PE16.8 ,5X, 8HETAP    = 1PE16.8 ,12X, 1H= 1PE16.8 /  9X, 8HP
    HPIS    = 1PE16.8 ,12X,1H= 1PE16.8,5X,8HCØN3    = 1PE16.8,5X,8HPNS
    I = 1PE16.8 / 9X,8HWT      = 1PE16.8,12X,1H= 1PE16.8,5X, 8HHPT    =1
    JPE16.8 ,5X, 8HPNSPR  = 1PE16.8 / 9X, 8HFC      = 1PE16.8 )
     RETURN
 100 GØ TØ (104,110,120,210) ,J
 104 WRITE (6,105)
 105 FØRMAT(1H1///55X,21HH1 ENGINE INPUT DATA  ///)
     GØ TØ 130
 110 WRITE (6,115)
 115 FØRMAT(1H1///55X, 21HF1 ENGINE INPUT DATA  ///)
     GØ TØ 130
 120 WRITE (6,125)
 125 FØRMAT(1H1///55X,21HJ2 ENGINE INPUT DATA  ///)
 130 GØ TØ (140,140,135,210) ,J
 140 WRITE (6,142 ) KBAL, INDPR,FEN,DEL7,XMREN,DEL9,PA,TN ,PPI,TPIR,
    2 TRSG, SG
 142 FØRMAT (  9X, 8HKBAL    = I3,18X, 8HINDPR  = I3,18X, 8HFEN     =
    2 1PE16.8, 5X, 8HDEL7    = 1PE16.8  /  9X, 8HXMREN  = 1PE16.8,5X, 8H
    3DEL9    = 1PE16.8,5X, 8HPA      = 1PE16.8,5X,8HTN      = 1PE16.8/
    4  9X, 8HPPI     = 1PE16.8,12X, 1H= 1PE16.8,5X, 8HTPIR    = 1PE16.8,
    4 12X, 1H= 1PE16.8 /  9X, 8HTRSG    = 1PE16.8, 12X, 1H= 1PE16.8,5X,
    5 8HSG      = 1PE16.8, 12X, 1H= 1PE16.8 )
     WRITE  (6,145) C7, DPI ,WMIS, C3,C4,C5 ,C6
 145 FØRMAT ( 9X,7HC7      4(1H= 1PE16.8,12X) /16X, 4(1H= 1PE16.8,12X)/
    A 16X, 4(1H= 1PE16.8,12X)/ 16X,4(1H= 1PE16.8,12X) /16X,1H= 1PE16.8,
    B12X, 1H= 1PE16.8,5X, 7HDPI      2(1H= 1PE16.8,12X) / 9X,7HWMIS    4
    C(1H= 1PE16.8,12X) / 9X,7HC3      4(1H= 1PE16.8,12X )/16X,1H= 1PE16
    D.8,12X,1H= 1PE16.8,5X, 7HC4      2(1H= 1PE16.8,12X)/16X, 4(1H= 1PE1
    E6.8,12X)/ 9X,7HC5      4(1H= 1PE16.8,12X)/16X,1H= 1PE16.8,12X,1H=
    F1PE16.8,5X,7HC6      2(1H= 1PE16.8,12X) /16X,4(1H= 1PE16.8,12X))
     WRITE (6,155) DRI,DRD,DEL4,DEL5, DLTR, R
 155 FØRMAT ( 9X,7HDRI      4(1H= 1PE16.8,12X)/16X,4(1H= 1PE16.8,12X)/16
```

FIGURE 106. SYMBOLIC LISTING OF SUBROUTINE EPRINT (cont.)

```
     AX, 4(1H= 1PE16.8,12X) /16X,4(1H= 1PE16.8,12X)/16X,4(1H= 1PE16.8,12
     BX)/ 9X,7HDRD     4(1H= 1PE16.8,12X)/16X, 4(1H= 1PE16.8,12X)/16X,4(
     C1H=1PE16.8,12X)/16X,4(1H=1PE16.8,12X)/16X,4(1H= 1PE16.8,12X)/ 9X,7
     DHDEL4    4(1H=1PE16.8,12X)/16X,4(1H=1PE16.8,12X)/16X,4(1H= 1PE16.8
     E,12X) /16X,4(1H= 1PE16.8,12X)/16X,4(1H= 1PE16.8,12X)/ 9X,7HDEL5
     F 4(1H= 1PE16.8,12X)/16X,4(1H= 1PE16.8,12X)/ 16X,4(1H= 1PE16.8,12X)
     G/16X,4(1H= 1PE16.8,12X)/16X,4(1H= 1PE16.8,12X)/ 9X,7HDLTR    4(1H=
     H 1PE16.8,12X)/16X, 4(1H= 1PE16.8,12X)/16X, 4(1H= 1PE16.8,12X) /16X
     I,4(1H= 1PE16.8,12X )/16X,4(1H= 1PE16.8,12X) / 9X,7HR       4(1H= 1
     JPE16.8,12X)/16X, 4(1H= 1PE16.8,12X)/16X, 4(1H= 1PE16.8,12X) /16X,
     K 4(1H= 1PE16.8,12X) /16X, 4(1H= 1PE16.8,12X))
      GØ TØ (180,182,245,245) ,J
  180 WRITE (6,1182)
 1182 FØRMAT(1H1///55X,21HH1 ENGINE INPUT DATA    ,20X,9HPAGE  2   ///)
      GØ TØ 181
  182 WRITE (6,1184)
 1184 FØRMAT(1H1///55X,21HF1 ENGINE INPUT DATA    ,20X,9HPAGE  2   ///)
  181 WRITE  (6,165)                         DPD,DEL1,DEL2,DEL3,DLTP ,RGG,RX,C
     A1
  165 FØRMAT  (                               9X,8HDPD    = 1PE16.8,12X, 1H= 1
     CPE16.8,5X, 8HDEL1   = 1PE16.8,12X,1H= 1PE16.8/ 9X,8HDEL2   = 1PE16
     D.8,12X,1H= 1PE16.8 ,5X,8HDEL3   = 1PE16.8,12X,1H= 1PE16.8/ 9X,8HDL
     ETP   = 1PE16.8,12X,1H= 1PE16.8,5X, 8HRGG    = 1PE16.8,5X, 8HRX
     F = 1PE16.8 / 9X, 7HC1          4(1H= 1PE16.8,12X)/16X,4(1H= 1PE16.8,
     G12X)/16X,4(1H= 1PE16.8,12X))
      WRITE (6,175) PTI,PTES,ATI,DW,IPB,GR,XNTST,ETAGR,DEL6,ATE
  175 FØRMAT ( 9X,8HPTI    =1PE16.8,12X,1H=1PE16.8,5X,8HPTES   =1PE16.8
     A ,12X, 1H= 1PE16.8/  9X,8HATI    = 1PE16.8,12X,1H= 1PE16.8,5X, 8HD
     BW     = 1PE16.8,12X, 1H= 1PE16.8/ 9X,8HIPB    =       I3,25X,1H=
     C  I3,18X,8HGR     = 1PE16.8,12X, 1H= 1PE16.8/ 9X,8HXNTST  = 1PE16.
     D8,12X,1H= 1PE16.8,5X,8HETAGR  = 1PE16.8,12X,1H= 1PE16.8/  9X,8HDEL
     E6   = 1PE16.8,12X,1H= 1PE16.8,5X,8HATE    = 1PE16.8,12X,1H= 1PE16.
     F8     )
      GØ TØ (184,186,245,245) ,J
  184 WRITE (6,185) DEL8,ETCF,EPS,PNS, CØN ,ETCS ,AXD,DELTX
  185 FØRMAT ( 9X,8HDEL8   = 1PE16.8,5X, 8HETCF   = 1PE16.8,5X,8HEPS
     A=  1PE16.8,5X,8HPNS    = 1PE16.8/ 9X,8HCØN    =1PE16.8, 5X,8HETCS
     B =  1PE16.8 ,5X, 8HAXD    = 1PE16.8,5X, 8HDELTX  = 1PE16.8 )
  132 WRITE (6,134) ATC,CPCX,  FX,XMRTC,HPAUX
  134 FØRMAT( 9X,8HATC    = 1PE16.8,5X,8HCPCX   = 1PE16.8,5X, 8HFX     =
     A 1PE16.8, 5X, 8HXMRTC  = 1PE16.8 /  9X, 8HHPAUX  = 1PE16.8 , 12X,
     B 1H= 1PE16.8 )
      RETURN
  186 WRITE (6,190) DEL8,ETCF,EPS,PNS,CØN,ETCS ,CPCX,XMRTC  ,ATC
  190 FØRMAT ( 9X,8HDEL8   = 1PE16.8,5X, 8HETCF   = 1PE16.8 ,5X,8HEPS
     A = 1PE16.8,5X,8HPNS    = 1PE16.8/  9X, 8HCØN    = 1PE16.8, 5X,
     B 8HETCS   = 1PE16.8, 5X, 8HCPCX   = 1PE16.8, 5X, 8HXMRTC  = 1PE16
     C .8 / 9X,8HATC    = 1PE16.8)
      RETURN
  210 WRITE (6,212)
```

FIGURE 106. SYMBOLIC LISTING OF SUBROUTINE EPRINT (cont.)

```
212 FØRMAT(1H1///51X,29HRL10 ENGINE INPUT DATA                    ///)
     WRITE ( 6,214) KBAL,FCL,XMRTC,ATC,PPI,TPIS
214 FØRMAT ( 9X,8HKBAL    = I3,18X, 8HFCL     = 1PE16.8,5X, 8HXMRTC  = 1
    APE16.8, 5X,8HATC     = 1PE16.8/  9X, 8HPPI     = 1PE16.8,12X,
    B1H= 1PE16.8,5X, 8HTPIS    = 1PE16.8, 12X, 1H= 1PE16.8 )
     WRITE (6,215)      XNPN,    XNP,RHPI,Q,HPAUX,PCIEL, TØL1
215 FØRMAT (  9X, 8HXNPN    = 1PE16.8 ,  12X, 1H=  1PE16.8
    A                                         ,5X,8HXNP     = 1PE16.8,12
    BX,1H= 1PE16.8/ 9X,8HRHPI    = 1PE16.8,12X,1H= 1PE16.8,5X,8HQ       =
    C1PE16.8,12X,1H= 1PE16.8/ 9X,8HHPAUX   = 1PE16.8,12X,1H= 1PE16.8,5X,
    D8HPCIEL   = 1PE16.8,5X,8HTØL1    = 1PE16.8)
     WRITE (6,225) TØL2,TØL3,TØL4,TØL5,TØL6,TØL7,TØL8,TØL9,TØL10,CØN1,
    1CØN2,CØN3,CØN4,CØN5,CØN6,CØN7,CØN8,CØN9,CØN10,CØN11,CØN12,CØN13,CØ
    2N14,CØN15,CD2,CØN,AVT,ETGER,EPS
225 FØRMAT ( 9X,8HTØL2    = 1PE16.8,5X,8HTØL3    = 1PE16.8,5X,8HTØL4    =
    A   1PE16.8,5X,8HTØL5    = 1PE16.8/  9X, 8HTØL6    = 1PE16.8,5X,   8H
    BTØL7    = 1PE16.8,5X,8HTØL8    = 1PE16.8,5X,8HTØL9    = 1PE16.8/  9X
    C, 8HTØL10   = 1PE16.8,5X,8HCØN1    = 1PE16.8,5X,8HCØN2    = 1PE16.8,
    D5X,8HCØN3    = 1PE16.8/  9X,  8HCØN4    = 1PE16.8, 5X,  8HCØN5    = 1
    EPE16.8,5X,8HCØN6    = 1PE16.8,5X,8HCØN7    = 1PE16.8/  9X,8HCØN8    =
    F 1PE16.8,5X,8HCØN9    = 1PE16.8,5X,8HCØN10   = 1PE16.8,5X,8HCØN11   =
    G 1PE16.8/  9X,8HCØN12   = 1PE16.8,5X,8HCØN13   = 1PE16.8,5X,8HCØN14
    H = 1PE16.8,5X,8HCØN15   = 1PE16.8/  9X,8HCD2     = 1PE16.8,5X,
    I 8HCØN     = 1PE16.8,5X, 8HAVT     = 1PE16.8,5X, 8HETGER   = 1PE16.8/
    J 9X,8HEPS     = 1PE16.8 )
     RETURN
135 WRITE (6,137) KBAL,INDPR,FEN, DEL7, XMREN,PA,TTI(1),PUVA, PPI,TPIS
    1, TRSG, SG
137 FØRMAT (9X,8HKBAL    = I3,18X,8HINDPR   = I3,18X,8HFEN     = 1PE16.8,
    A  5X,8HDEL7    = 1PE16.8/ 9X, 8HXMREN   = 1PE16.8, 5X, 8HPA       =
    B1PE16.8, 5X, 8HTTI     = 1PE16.8, 5X, 8HPUVA    = 1PE16.8 / 9X,8HPPI
    C     = 1PE16.8,12X,1H= 1PE16.8, 5X, 8HTPIS    = 1PE16.8,12X, 1H=
    D1PE16.8 /9X,  8HTRSG    = 1PE16.8, 12X,1H= 1PE16.8 ,5X,7HSG
    E 2(1H= 1PE16.8,12X))
     WRITE (6,138) C1,C3,C4, C5, C6,C8,PUCØF,PUSEC,DEL1,DEL2,FAUX,ETCF
138 FØRMAT (9X, 7HC1       4(1H= 1PE16.8,12X) /16X,4(1H=1PE16.8,12X)/16
    AX, 4(1H= 1PE16.8,12X)/9X,7HC3        4(1H= 1PE16.8,12X) /16X,1H=1PE
    B16.8,12X,1H= 1PE16.8,5X, 7HC4        2(1H= 1PE16.8,12X) /16X,4 (1H=
    C 1PE16.8,12X) /9X, 7HC5        4(1H= 1PE16.8,12X)    /16X,1H=1PE16.8,
    D12X,1H= 1PE16.8, 5X, 7HC6       2(1H= 1PE16.8,12X)/ 16X, 4(1H= 1PE1
    E6.8,12X)/9X,7HC8        4(1H= 1PE16.8,12X)/16X,4(1H= 1PE16.8,12X)/9X
    F,7HPUCØF    4(1H= 1PE16.8,12X) ,14(/ 16X, 4(1H= 1PE16.8, 12X) ) /
    G          9X,     7HPUSEC
    J  4(1H= 1PE16.8,12X)/ 16X, 4(1H= 1PE16.8,12X)/ 16X, 1H= 1PE16.8,12
    KX, 1H= 1PE16.8,5X, 7HDEL1     2(1H= 1PE16.8,12X)/ 9X, 7HDEL2      1H
    L= 1PE16.8,12X, 1H= 1PE16.8,5X, 8HFAUX    = 1PE16.8,5X, 8HETCF    =
    M1PE16.8)
     WRITE (6,242) DEL4, DEL5,DRI
242 FØRMAT (9X, 7HDEL4     4(4(1H= 1PE16.8,12X) /16X) , 4(1H= 1PE16.8,
    A 12X) /9X, 7HDEL5     4(4(1H= 1PE16.8,12X) /16X),4(1H= 1PE16.8,
```

FIGURE 106. SYMBOLIC LISTING OF SUBROUTINE EPRINT (cont.)

```
    B 12X) /9X, 7HDRI      4(4(1H= 1PE16.8,12X) /16X),4(1H= 1PE16.8,
    C 12X) )
      WRITE (6,290)
290 FØRMAT(1H1///55X,21HJ2 ENGINE INPUT DATA    ,20X,9HPAGE  2   ///)
      WRITE (6,139) DRD,DLTR,R,          DW,        ATE,  DPI
    1    ,     ATI,     DLTP,  DPD,    IPB,DEL3,    PTI,  HPAUX,      CNST1,
    ACNST2,    CNST3,    CNST4, CNST5,    CNST6,CNST7,CNST8, DEL20,DEL21,
    BDEL22, DEL23   ,    DEL24, DEL25,    DEL26,    DEL27,    DEL28, DEL29,
    1DEL35,XMRTC,     PNS,   EPS,
    CETCS,  ATC,          DMØØ,  DMFØ,                  ABPNI,    EPSM,
    DWPRESF,   WPRESØ ,   CØN
139 FØRMAT(9X, 7HDRD       4(4(1H= 1PE16.8,12X) /16X),4(1H= 1PE16.8,
    D 12X) /9X, 7HDLTR     4(4(1H= 1PE16.8,12X) /16X),4(1H= 1PE16.8,
    E 12X) /9X, 7HR        4(4(1H= 1PE16.8,12X) /16X),4(1H= 1PE16.8,
    1 12X)  /9X,8HDW      = 1PE16.8,12X,1H= 1PE16.8 , 5X, 7HATE      2(1H
    A= 1PE16.8,12X) /9X,8HDPI     = 1PE16.8,12X,1H=1PE16.8,5X,7HATI
    B2(1H= 1PE16.8,12X) /9X,8HDLTP    = 1PE16.8,12X, 1H= 1PE16.8,5X,
    C7HDPD    2(1H= 1PE16.8,12X)/9X,8HIPB     = I3,25X,1H= I3,18X,8HDEL3
    D   =1PE16.8,5X,8H          =1PE16.8/9X,8HPTI      =1PE16.8,5X,8H
    Z=1PE16.8,5X,8HHPAUX  =1PE16.8,5X,8H          =1PE16.8/9X,8HCNST1   =
    X1PE16.8,5X,                                               8HCNST2   =
    E 1PE16.8,5X,8HCNST3  =1PE16.8,5X,8HCNST4   =1PE16.8/9X,8HCNST5   =
    F 1PE16.8,5X,8HCNST6  =1PE16.8,5X,8HCNST7   =1PE16.8,5X,8HCNST8   =
    G 1PE16.8/9X,8HDEL20  =1PE16.8,5X,8HDEL21   =1PE16.8,5X,8HDEL22   =
    H 1PE16.8,5X,8HDEL23  =1PE16.8/9X,8HDEL24   =1PE16.8,5X,8HDEL25   =
    2 1PE16.8,5X,8HDEL26  =1PE16.8,5X,8HDEL27   =1PE16.8/9X,8HDEL28   =
    J 1PE16.8,5X,8HDEL29  =1PE16.8,5X,8HDEL35   =1PE16.8,5X,8HXMRTC   =
    K 1PE16.8/9X,8HPNS    =1PE16.8,5X,8HEPS     =1PE16.8,5X,8HETCS    =
    L 1PE16.8,5X,8HATC    =1PE16.8/9X,8HDMØØ    =1PE16.8,5X,8HDMFØ    =
    M 1PE16.8,5X,8HABPNI  =1PE16.8,5X,8HEPSM    =1PE16.8/9X,8HWPRESF =
    N 1PE16.8,5X,8HWPRESØ = 1PE16.8 ,  8HCØN     =1PE16.8)
      RETURN
250 WRITE(6,345)FE,XISPE,WEØ,WEF,XMRE,FC,PNS,PCIE,XMRTC,WC,WCØ,WCF,CS,
    1CSTHE,CF,CFTHE
345 FØRMAT( 9X,8HFE       = 1PE16.8,5X,8HXISPE  = 1PE16.8, 5X,8HWEØ     =
    4 1PE16.8, 5X,8HWEF     = 1PE16.8 / 9X,8HXMRE    = 1PE16.8 ,5X, 8HFC
    B     = 1PE16.8 ,5X, 8HPNS      = 1PE16.8 ,5X, 8HPCIE     = 1PE16.8  / 9
    CX, 8HXMRTC  = 1PE16.8, 5X, 8HWC       = 1PE16.8 ,5X, 8HWCØ      = 1PE1
    D6.8, 5X, 8HWCF     = 1PE16.8/  9X, 8HCS       = 1PE16.8 ,5X, 8HCSTHE
    E = 1PE16.8, 5X, 8HCF      = 1PE16.8,5X, 8HCFTHE  = 1PE16.8 )
      WRITE(6,355) WP,QP,XNP,H,PPD,ETAP,BHPP,XNPSH,PPIS,RHPI,WT,PTI,
    1PTES,TTI,CPT,GAMT,RGT,XNT,UVRAT,ETAT,HPT,PRT
355 FØRMAT( 9X,8HWP      = 1PE16.8,12X, 1H= 1PE16.8 , 5X, 8HQP       = 1P
    2E16.8, 12X, 1H= 1PE16.8 / 9X,8HXNP      = 1PE16.8 ,12X , 1H= 1PE16.8
    3, 5X, 8HH       = 1PE16.8 ,12X, 1H= 1PE16.8  / 9X, 8HPPD      = 1PE16
    4.8 ,12X, 1H= 1PE16.8, 5X,8HETAP     = 1PE16.8, 12X, 1H= 1PE16.8 / 9X
    5, 8HBHPP    = 1PE16.8 ,12X, 1H= 1PE16.8, 5X, 8HXNPSH  = 1PE16.8 ,
    612X, 1H= 1PE16.8 / 9X, 8HPPIS     = 1PE16.8,12X, 1H= 1PE16.8,  5X,8H
    7RHPI    = 1PE16.8 , 12X, 1H= 1PE16.8 / 9X, 8HWT       = 1PE16.8 ,12X,
    8 1H= 1PE16.8, 5X, 8HPTI      = 1PE16.8 , 12X, 1H= 1PE16.8 / 9 X,
```

FIGURE 106. SYMBOLIC LISTING OF SUBROUTINE EPRINT (cont.)

```
    98HPTES   = 1PE16.8, 12X, 1H= 1PE16.8, 5X, 8HTTI    = 1PE16.8, 12X,
    A1H= 1PE16.8/9X,8HCPT    =1PE16.8,5X,8H        =1PE16.8,5X,8HGAMT
    B=   1PE16.8,5X,8H       =1PE16.8/9X,8HRGT     =1PE16.8,5X,8H
    C=   1PE16.8,5X,8HXNT    =1PE16.8,5X,8H        =1PE16.8/9X,8HUVRAT
    D=   1PE16.8,5X,8H       =1PE16.8,5X,8HETAT    =1PE16.8,5X,8H
    E=   1PE16.8/9X,8HHPT    =1PE16.8,5X,8H        =1PE16.8,5X,8HPRT
    F=   1PE16.8,5X,8H       =1PE16.8    )
     WRITE(6,365) WG,WGØ,WGF,DBPN,XMRG,CSG,PGG,WPUØ,WBP,RESØG,RESFG
365  FØRMAT(      9X,8HWG    =1PE16.8,5X,8HWGØ     =1PE16.8,5X,8HWGF
    A=   1PE16.8,5X,8HDBPN   =1PE16.8/9X,8HXMRG    =1PE16.8,5X,8HCSG
    B=   1PE16.8,5X,8HPGG    =1PE16.8,5X,8HWPUØ    =1PE16.8/9X,8HWBP
    C=   1PE16.8,5X,8HRESØG  =1PE16.8,5X,8HRESFG   =1PE16.8   )
245  RETURN
     END
```

FIGURE 106. SYMBOLIC LISTING OF SUBROUTINE EPRINT (cont.)

LXXXVI. SUBROUTINE INFL

The purpose of this subroutine is to generate a set of influence coefficients. This subroutine must be used in conjunction with an engine balance/rebalance program. Influence coefficients are generated by perturbing an independent variable from its nominal value and rebalancing the engine system to determine the corresponding changes in the dependent variables of interest.

Each independent variable is perturbed in both directions from its nominal value. This results in a set of influence coefficients which more nearly depict the tangents, at the nominal operating points, to the performance curves which relate the variations in each of the dependent variables due to a variation in an independent variable.

The last independent variable perturbed is nominal engine thrust. The influence coefficients generated by this perturbation may be used to approximate the effect of changing the nominal engine thrust level.

CALLING SEQUENCE

The calling sequence is:

```
CALL INFL (CON, KEND, TABI, TABD, INDV, DEPV,
          N1, N2, Y, VIND, VDEP, NLIST)
```

SOLUTION METHOD

Set counters for the number of independent and dependent variables:

```
1.  NIND = N1
2.  ND = N2
```

Test for first time in subroutine for this set of parameters:

```
3.  If NCON = 0, go to 4
    If NCON ≠ 0, go to 20
```

Set nominal independent and dependent variables and the balance/rebalance indicator:

```
4.  Do 5, I = 1, ND
5.  VDEP(I) = TABD(I)
6.  KBAL = 2
7.  Do 8, I = 1, NIND
8.  VIND(I) = TABI(I)
```

Zero out Y array:

```
 9.  Do 11, I = 1, NIND
10.  Do 11, J = 1, ND
11.  Y(I,J) = 0
```

Set array DD equal to array VIND:

```
12.  Do 13, I = 1, NIND
13.  DD(I) = VIND(I)
```

Increment independent variable control counter:

```
14.  NCON = NCON + 1
```

Perturb independent variable:

```
15.  DD(NCON) = (DD(NCON))(CON)
16.  L = 1
```

Reset independent variables with perturbed value:

```
17.  Do 18, I = 1, NIND
18.  TABI(I) = DD(I)
19.  Return
```

Put current values of the dependent variables into table CURDP:

```
20.  Do 21, I = 1, ND
21.  CURDP(I) = TABD(I)
```

Compute dimensional influence coefficient:

```
22.  Do 23, N = 1, ND
23.  Y(NCON, N) = -(CURDP(N) - VDEP(N)) + Y(NCON, N)
```

Test if independent variable has been perturbed in both directions:

24. If $L \leq 1$, go to 25

If $L > 1$, go to 30

Perturb independent variable in other direction:

25. $DD(NCON) = (DD(NCON))/(CON^2)$

Change sign of Y array:

26. Do 27, N = 1, ND

27. $Y(NCON, N) = -Y(NCON, N)$

28. $L = 2$

29. Go to 17

Reset perturbed independent variable to nominal value:

30. $DD(NCON) = (DD(NCON))(CON)$

Compute non-dimensional influence coefficient:

31. $DEL = (VIND(NCON))/((VIND(NCON))(CON)$

$- (VIND(NCON))/(CON))$

32. Do 33, N = 1, ND

33. $Y(NCON, N) = (Y(NCON, N))(DEL)/(VDEP(N))$

Test if last independent variable should be perturbed:

```
34.  If $NCON + 1 < NIND$, go to 14
     If $NCON + 1 = NIND$, go to 35
     If $NCON + 1 > NIND$, go to 37
```

Reset balance/rebalance indicator for a balance:

```
35.  $KBAL = 1$
36.  Go to 14
```

Reset indicators and print out the influence coefficient:

```
37.  $NCON = 0$
38.  $KEND = 1$
39.  $NNC = 1$
40.  If $NIND \leq 8$, go to 43
     If $NIND > 8$, go to 41
41.  $NND = 8$
42.  Go to 44
43.  $NND = NIND$
44.  Write output heading
45.  Write output
46.  Do 47, $K = 1$, ND
```

```
47.  Write output
48.  If $NND < NIND$, go to 49
     If $NND \geq NIND$, go to 19
49.  $NNC = NNC + NND$
50.  $NNDA = NND + 8$
51.  If $NNDA < NIND$, go to 54
     If $NNDA \geq NIND$, go to 52
52.  $NND = NIND$
53.  Go to 44
54.  $NND = NNDA$
55.  Go to 44
```

SUBROUTINE INFL NOMENCLATURE

Symbol	Description	Units	Reference
CON	Amount to perturb independent variables	—	CALL, 15, 25, 30, 31
CURDP	Array containing the current values of the dependent variables	—	DIM, 21, 23
DD	Array containing the independent variables	—	DIM, 13, 15, 18, 25, 30
DEL	—	—	31, 33
DEPV	Array containing alpha names of dependent variables	—	CALL, DIM
I	Loop control counter	—	4, 5, 7, 8, 9, 11, 12, 13, 17, 18, 20, 21
INDV	Array containing alpha names of independent variables	—	CALL, DIM
J	Loop control counter	—	10, 11
K	Loop control counter	—	46
KBAL	Balance/rebalance indicator	—	/CBAL/, 6, 35
KEND	Flag to terminate re-entry to subroutine	—	CALL, 38
L	Indicator to determine which direction to perturb the independent variable	—	16, 24, 28
N1	Number of independent variables	—	CALL, 1

SUBROUTINE INFL NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
N2	Number of dependent variables	—	CALL, 2
N	Loop control counter	—	22, 23, 26, 27, 32, 33
NCON	Counter to determine which independent variable is to be perturbed	—	DATA, 3, 14, 15, 23, 25, 27, 30, 31, 33, 34, 37
ND	Number of dependent variables	—	2, 4, 10, 20, 22, 26, 32, 46
NIND	Number of independent variables	—	1, 7, 9, 12, 17, 34, 40, 43, 48, 51, 52
NLIST	Engine name for output heading	—	CALL
NNC	Counter for output control	—	39, 49
NND	Counter for output control	—	41, 43, 48, 49, 50, 52, 54
NNDA	Counter for output control	—	50, 51, 54
TABD	Array containing the dependent variables	—	CALL, DIM, 5, 21
TABI	Array containing the independent variables	—	CALL, DIM, 8, 18
VDEP	Array containing nominal dependent variables	—	CALL, DIM, 5, 23, 33
VIND	Array containing nominal independent variables	—	CALL, DIM, 8, 13, 31
Y	Array containing influence coefficients	—	CALL, DIM, 11, 23, 27, 33

```
$IBFTC INFL    REF,FULIST
      SUBROUTINE INFL(CON,KEND,TABI,TABD,INDV,DEPV,N1,N2,Y,VIND,VDEP,NLIST)
      COMMON/CBAL/KBAL
      DIMENSION TABI(10),TABD(10),INDV(10),DEPV(10),Y(10,10),VIND(10),
     1VDEP(10),DD(10),CURDP(10)
      DATA NCON /0/
      NIND = N1
      ND = N2
      IF(NCON)9996,9995,9996
C SET UP NOMINAL DEPENDENT VARIABLES
 9995 DO 95 I =1,ND
   95 VDEP(I)=TABD(I)
      KBAL = 2
C SET UP NOMINAL INDEPENDENT VARIABLES
      DO 94 I =1,NIND
   94 VIND(I)=TABI(I)
      DO 33 I =1,NIND
      DO 31 J =1,ND
   31 Y(I,J)=0.
   33 CONTINUE
   32 DO 10 I =1,NIND
   10 DD(I) = VIND(I)
   11 NCON = NCON+1
      DD(NCON) = DD(NCON)*CON
      L = 1
 9997 DO 97 I = 1,NIND
   97 TABI(I) = DD(I)
   16 RETURN
 9996 DO 96 I =1,ND
   96 CURDP(I) = TABD(I)
  60  DO 61 N=1,ND
      Y(NCON,N) = -(CURDP(N) - VDEP(N)) + Y(NCON,N)
  61  CONTINUE
      IF(L-1)62,62,65
  62  DD(NCON) = DD(NCON)/CON**2
      DO 66 N=1,ND
      Y(NCON,N)=-Y(NCON,N)
  66  CONTINUE
      L =2
      GO TO 9997
   65 DD(NCON) = DD(NCON) * CON
      DEL=VIND(NCON)/(VIND(NCON)*CON-VIND(NCON)/CON)
      DO 67 N=1,ND
      Y(NCON,N)=Y(NCON,N)*DEL/VDEP(N)
  67  CONTINUE
      IF((NCON-NIND)+1)11,18,17
   18 KBAL = 1
      GO TO 11
   17 NCON = 0
      KEND = 1
```

FIGURE 107. SYMBOLIC LISTING OF SUBROUTINE INFL

```
      NNC = 1
      IF(NIND-8)43,43,41
   41 NND = 8
      GØ TØ 42
   43 NND = NIND
   42 WRITE(6,40) NLIST
   40 FØRMAT(1H1 45X,39HNØN-DIMENSIØNAL INFLUENCE CØEFFICIENTS    ///)
      WRITE(6,44) (INDV(J),J=NNC,NND        )
44    FØRMAT(1H0 13X,7(A6, 9X), A6  //)
      DØ 46 K= 1,ND
      WRITE(6,47) DEPV(K), (Y(J,K), J = NNC,NND       )
   46 CØNTINUE
   47 FØRMAT(3H0  A6,8(3X,E12.5)  )
      IF(NND-NIND)50,16,16
   50 NNC = NNC + NND
      NNDA = NND + 8
      IF(NNDA-NIND)52,51,51
   51 NND = NIND
      GØ TØ 42
   52 NND = NNDA
      GØ TØ 42
      END
```

FIGURE 107. SYMBOLIC LISTING OF SUBROUTINE INFL (cont.)

LXXXVII. SUBROUTINE STINFL

The purpose of this subroutine is to make the generated influence coefficients of the engine balance/rebalance programs available to the powered flight simulation program. This is accomplished by setting the influence coefficients and the nominal dependent and independent variables used to generate them, into COMMON arrays addressable by these two sections of the TENSOR program.

CALLING SEQUENCE

```
CALL STINFL(N)
```

SOLUTION METHOD

Set vehicle stage number:

```
1.  K = N
```

Set the independent variables and influence coefficients into COMMON arrays:

```
2.  Do 5 I = 1, NIND

3.  VINDA(I, K) = VIND(I)

4.  Do 5 II = 1, ND
```

```
 5.  YY(I, II, K) = Y(I, II)
```

Set dependent variables and count of dependent and independent variables into COMMON arrays:

```
 6.  Do 7I = 1, ND
 7.  VDEPA(I, K) = VDEP(I)
 8.  NINDA(K) = NIND
 9.  NDA(K) = ND
10.  Return
11.  End
```

SUBROUTINE STINFL NOMENCLATURE

Symbol	Description	Units	Reference
K	Vehicle stage number	—	1
N	Vehicle stage number	—	CALL, 1
NDA	Array of the number of dependent variables, by stages	—	/INFTLB/, 9
ND	Number of dependent variables	—	/INFTLB/, 9
NIND	Number of independent variables	—	/INFTLB/, 8
NINDA	Array of the number of independent variables, by stages	—	/INFTLB/, 8
VDEP	Array of dependent variables	—	/INFTLB/, 7
VDEPA	Array of dependent variables, by stages	—	/INFTLB/, 7
VIND	Array of independent variables	—	/INFTLB/, 2
VINDA	Array of independent variables, by stages	—	/INFTLB/, 2
Y	Array of influence coefficients generated by Subroutine INFL	—	/INFTLB/, 5
YY	Array of influence coefficients generated by Subroutine INFL, by stages	—	/INFTLB/, 5

```
$IBFTC STINFL  FULIST,REF
      SUBRØUTINE STINFL(N)
      CØMMØN /INFLTB/ YY(10,10,6),VINDA(10,6),VDEPA(10,6),NINDA(6),NIND,
     1NDA(6),Y(10,10),VDEP(10),VIND(10),ND
      K = N
      DØ 75 I=1,NIND
      VINDA(I,K) = VIND(I)
      DØ 75 II = 1,ND
   75 YY(I,II,K) =  Y(I,II)
      DØ 80 I=1,ND
   80 VDEPA(I,K) = VDEP(I)
      NINDA(K) = NIND
      NDA(K) = ND
      RETURN
      END
```

FIGURE 108. SYMBOLIC LISTING OF SUBROUTINE STINFL

LXXXVIII. SUBROUTINE SETNOM

This subroutine is used to save and reset the nominal Block I input data, between cases.

CALLING SEQUENCE

The calling sequence is:

CALL SETNOM (J, JJ)

SOLUTION METHOD

Determine whether to set or reset nominals.

1. If JJ $>$ 0, go to 172

   If JJ $\leq$ 0, go to 2

Determine engine nominals to be set.

2. If J = 1, go to 3

   If J = 2, go to 68

   If J = 3, go to 342

   If J = 4, go to 129

Set H-1 engine nominals.

```
3.   Do 24   ITI = 1, 2
4.   XTPIR (ITI)    = TPIR (ITI)
5.   XPPI (ITI)     = PPI (ITI)
6.   XHPAUX (ITI)   = HPAUX (ITI)
7.   XTRSG (ITI)    = TRSG (ITI)
8.   XSG (ITI)      = SG (ITI)
9.   XDPI (ITI)     = DPI (ITI)
10.  XDPD (ITI)     = DPD (ITI)
11.  XDEL1 (ITI)    = DEL1 (ITI)
12.  XDEL2 (ITI)    = DEL2 (ITI)
13.  XDEL3 (ITI)    = DEL3 (ITI)
14.  XDLTP (ITI)    = DLTP (ITI)
15.  XPTI (ITI)     = PTI (ITI)
16.  XPTES (ITI)    = PTES (ITI)
17.  XATI (ITI)     = ATI (ITI)
18.  XDW (ITI)      = DW (ITI)
19.  IIPB (ITI)     = IPB (ITI)
20.  XGR (ITI)      = GR (ITI)
21.  XXNTST (ITI)   = XNTST (ITI)
22.  XETAGR (ITI)   = ETAGR (ITI)
23.  XDEL6 (ITI)    = DEL6 (ITI)
```

```
24.  XATE (ITI)        =  ATE (ITI)
25.  Do  29     II   =  1, 6
26.  XC3 (II)          =  C3 (II)
27.  XC4 (II)          =  C4 (II)
28.  XC5 (II)          =  C5 (II)
29.  XC6 (II)          =  C6 (II)
30.  Do  36    III   =  1, 20
31.  XDRI (III)        =  DRI (III)
32.  XDRD (III)        =  DRD (III)
33.  XDEL4 (III)       =  DEL4 (III)
34.  XDEL5 (III)       =  DEL5 (III)
35.  XDLTR (III)       =  DLTR (III)
36.  XR (III)          =  R (III)
37.  Do  38     NI   =  1, 4
38.  XWMIS (NI)        =  WMIS (NI)
39.  Do  41     IN   =  1, 6
40.  Do  41    INT   =  1, 2
41.  XC1 (IN, INT)  =  C1 (IN, INT)
42.  Do  44    NIT   =  1, 9
43.  Do  44    NTI   =  1, 2
44.  XC7 (NIT, NTI)  =  C7 (NIT, NTI)
```

45. XRGG = RGG

46. XTN = TN

47. XDEL9 = DEL9

48. XXMREN = XMREN

49. XRX = RX

50. XFEN = FEN

51. XDEL7 = DEL7

52. XPA = PA

53. XDEL8 = DEL8

54. XETCF = ETCF

55. KKBAL = KBAL

56. XATC = ATC

57. XXMRTC = XMRTC

58. XCPCX = CPCX

59. XEPS = EPS

60. XPNS = PNS

61. IINDPR = INDPR

62. XETCS = ETCS

63. XCON = CON

64. XDELTX = DELTX

65. XAXD = AXD

66. XFX = FX

67. Go to 171

Set F-1 engine nominals.

68. Do 88 ITI = 1, 2

69. FPPI (ITI) = PPI (ITI)

70. FTPIR (ITI) = TPIR (ITI)

71. FTRSG (ITI) = TRSG (ITI)

72. FSG (ITI) = SG (ITI)

73. FDPI (ITI) = DPI (ITI)

74. FDPD (ITI) = DPD (ITI)

75. FDEL1 (ITI) = DEL1 (ITI)

76. FDEL2 (ITI) = DEL2 (ITI)

77. FDEL3 (ITI) = DEL3 (ITI)

78. FDLTP (ITI) = DLTP (ITI)

79. FPTI (ITI) = PTI (ITI)

80. FPTES (ITI) = PTES (ITI)

81. FATI (ITI) = ATI (ITI)

82. FDW (ITI) = DW (ITI)

83. KIPB (ITI) = IPB (ITI)

84. FGR (ITI) = GR (ITI)

85. FXNTST (ITI) = XNTST (ITI)

```
 86.   FETAGR (ITI)   =  ETAGR (ITI)

 87.   FDEL6 (ITI)    =  DEL6 (ITI)

 88.   FATE (ITI)     =  ATE (ITI)

 89.   Do  93      II =  1, 6

 90.   FC3 (II)       =  C3 (II)

 91.   FC4 (II)       =  C4 (II)

 92.   FC5 (II)       =  C5 (II)

 93.   FC6 (II)       =  C6 (II)

 94.   Do  100    III =  1, 20

 95.   FDRI (III)     =  DRI (III)

 96.   FDRD (III)     =  DRD (III)

 97.   FDEL4 (III)    =  DEL4 (III)

 98.   FDEL5 (III)    =  DEL5 (III)

 99.   FDLTR (III)    =  DLTR (III)

100.   FR (III)       =  R (III)

101.   Do  102     NI =  1, 4

102.   FWMIS (NI)     =  WMIS (NI)

103.   Do  105     IN =  1, 6

104.   Do  105  INT   =  1, 2

105.   FC1 (IN, INT)  =  C1 (IN, INT)

106.   Do  108  NIT   =  1, 9
```

```
107.   Do 108  NTI  =  1, 2
108.   FC7(NIT, NTI)  =  C7(NIT, NTI)
109.   FRGG          =  RGG
110.   FTN           =  TN
111.   FDEL9         =  DEL9
112.   FXMREN        =  XMREN
113.   FRX           =  RX
114.   FFEN          =  FEN
115.   FDEL7         =  DEL7
116.   FPA           =  PA
117.   FDEL8         =  DEL8
118.   FETCF         =  ETCF
119.   FEPS          =  EPS
120.   FPNS          =  PNS
121.   KINDPR        =  INDPR
122.   FCPCX         =  CPCX
123.   FETCS         =  ETCS
124.   FCON          =  CON
125.   IKBAL         =  KBAL
126.   FATC          =  ATC
127.   FXMRTC        =  XMRTC
```

```
128.   Go to 171

Set RL-10 engine nominals.

129.   Do 136 NTT = 1, 2

130.   XQ (NTT)        = Q (NTT)

131.   XTPIS (NTT)     = TPIS (NTT)

132.   XXNPN (NTT)     = XNPN (NTT)

133.   XXNP (NTT)      = XNP (NTT)

134.   RHPAUX (NTT)    = HPAUX (NTT)

135.   RPPI (NTT)      = PPI (NTT)

136.   XRHPI (NTT)     = RHPI (NTT)

137.   XPCIE           = PCIEL

138.   XTOL1           = TOL1

139.   XTOL2           = TOL2

140.   XCON1           = CON1

141.   XCON2           = CON2

142.   XCON3           = CON3

143.   XCON4           = CON4

144.   XCON5           = CON5

145.   XCD2            = CD2

146.   XTOL3           = TOL3

147.   XTOL4           = TOL4
```

148. XTOL5 = TOL5

149. XCON6 = CON6

150. XCON7 = CON7

151. XCON8 = CON8

152. XTOL6 = TOL6

153. XTOL7 = TOL7

154. XTOL8 = TOL8

155. XTOL9 = TOL9

156. XFCL = FCL

157. XAVT = AVT

158. XETGER = ETGER

159. XCON9 = CON9

160. XCON10 = CON10

161. XCON11 = CON11

162. XCON12 = CON12

163. XCON13 = CON13

164. XCON14 = CON14

165. XCON15 = CON15

166. XTOL10 = TOL10

167. REPS = EPS

168. JKBAL = KBAL

```
169.   RATC          = ATC

170.   RXMRTC        = XMRTC

171.   Return

Determine engine nominals to be reset.

172.   If J = 1, go to 173

       If J = 2, go to 238

       If J = 3, go to 423

       If J = 4, go to 299

Reset H-1 engine nominals.

173.   Do 194 INI = 1, 2

174.   HPAUX(INI)    = XHPAUX(INI)

175.   PPI(INI)      = XPPI(INI)

176.   TPIR(INI)     = XTPIR(INI)

177.   TRSG(INI)     = XTRSG(INI)

178.   SG(INI)       = XSG(INI)

179.   DPI(INI)      = XDPI(INI)

180.   DPD(INI)      = XDPD(INI)

181.   DEL1(INI)     = XDEL1(INI)

182.   DEL2(INI)     = XDEL2(INI)

183.   DEL3(INI)     = XDEL3(INI)

184.   DLTP(INI)     = XDLTP(INI)
```

```
185.   PTI (INI)        =  XPTI (INI)
186.   PTES (INI)       =  XPTES (INI)
187.   ATI (INI)        =  XATI (INI)
188.   DW (INI)         =  XDW (INI)
189.   GR (INI)         =  XGR (INI)
190.   IPB (INI)        =  IIPB (INI)
191.   XNTST (INI)      =  XXNTST (INI)
192.   ETAGR (INI)      =  XETAGR (INI)
193.   DEL6 (INI)       =  XDEL6 (INI)
194.   ATE (INI)        =  XATE (INI)
195.   Do  199   NNI    =  1, 6
196.   C3 (NNI)         =  XC3 (NNI)
197.   C4 (NNI)         =  XC4 (NNI)
198.   C5 (NNI)         =  XC5 (NNI)
199.   C6 (NNI)         =  XC6 (NNI)
200.   Do  206    IIK   =  1, 20
201.   DRI (IIK)        =  XDRI (IIK)
202.   DRD (IIK)        =  XDRD (IIK)
203.   DEL4 (IIK)       =  XDEL4 (IIK)
204.   DEL5 (IIK)       =  XDEL5 (IIK)
205.   DLTR (IIK)       =  XDLTR (IIK)
```

```
206.  R(IIK)            =  XR(IIK)
207.  Do 208  NNN =  1, 4
208.  WMIS(NNN)  =  XWMIS(NNN)
209.  Do 211   NXT  =  1, 6
210.  Do 211  NWT =  1, 2
211.  C1(NXT, NWT)  =  XC1(NXT, NWT)
212.  Do 214  NAA  =  1, 9
213.  Do 214  NBB  =  1, 2
214.  C7(NAA, NBB)  =  XC7(NAA, NBB)
215.  RGG               =  XRGG
216.  TN                =  XTN
217.  DEL9              =  XDEL9
218.  XMREN             =  XXMREN
219.  RX                =  XRX
220.  FEN               =  XFEN
221.  DEL7              =  XDEL7
222.  PA                =  XPA
223.  DEL8              =  XDEL8
224.  ETCF              =  XETCF
225.  EPS               =  XEPS
226.  PNS               =  XPNS
```

```
227.   INDPR        = IINDPR
228.   ETCS         = XETCS
229.   CON          = XCON
230.   DELTX        = XDELTX
231.   AXD          = XAXD
232.   FX           = XFX
233.   KBAL         = KKBAL
234.   ATC          = XATC
235.   XMRTC        = XXMRTC
236.   CPCX         = XCPCX
237.   Go to 171
```

Reset F-1 engine nominals.

```
238.   Do 258  INI  = 1, 2
239.   PPI(INI)     = FPPI(INI)
240.   TPIR(INI)    = FTPIR(INI)
241.   TRSG(INI)    = FTRSG(INI)
242.   SG(INI)      = FSG(INI)
243.   DPI(INI)     = FDPI(INI)
244.   DPD(INI)     = FDPD(INI)
245.   DEL1 (INI)   = FDEL1 (INI)
246.   DEL2 (INI)   = FDEL2 (INI)
```

```
247.   DEL3 (INI)      =  FDEL3 (INI)
248.   DLTP (INI)      =  FDLTP (INI)
249.   PTI (INI)       =  FPTI (INI)
250.   PTES (INI)      =  FPTES (INI)
251.   ATI (INI)       =  FATI (INI)
252.   DW (INI)        =  FDW (INI)
253.   GR (INI)        =  FGR (INI)
254.   IPB (INI)       =  KIPB (INI)
255.   XNTST (INI)     =  FXNTST (INI)
256.   ETAGR (INI)     =  FETAGR (INI)
257.   DEL6 (INI)      =  FDEL6 (INI)
258.   ATE (INI)       =  FATE (INI)
259.   Do 263   NNI    =  1, 6
260.   C4 (NNI)        =  FC4 (NNI)
261.   C3 (NNI)        =  FC3 (NNI)
262.   C5 (NNI)        =  FC5 (NNI)
263.   C6 (NNI)        =  FC6 (NNI)
264.   Do 270    IIK   =  1, 20
265.   DRI (IIK)       =  FDRI (IIK)
266.   DRD (IIK)       =  FDRD (IIK)
267.   DEL4 (IIK)      =  FDEL4 (IIK)
```

```
268.   DEL5(IIK)      =  FDEL5(IIK)

269.   DLTR(IIK)      =  FDLTR(IIK)

270.   R(IIK)         =  FR(IIK)

271.   Do 272  NNN =  1, 4

272.   WMIS(NNN)   =  FWMIS(NNN)

273.   Do 275   NXT  =  1, 6

274.   Do 275  NWT =  1, 2

275.   C1(NXT, NWT)  =  FC1(NXT, NWT)

276.   Do 278   NAA  =  1, 9

277.   Do 278   NBB  =  1, 2

278.   C7(NAA, NBB)  =  FC7(NAA, NBB)

279.   RGG            =  FRGG

280.   TN             =  FTN

281.   DEL9           =  FDEL9

282.   XMREN          =  FXMREN

283.   RX             =  FRX

284.   FEN            =  FFEN

285.   DEL7           =  FDEL7

286.   PA             =  FPA

287.   DEL8           =  FDEL8

288.   ETCF           =  FETCF
```

289. EPS = FEPS

290. PNS = FPNS

291. INDPR = KINDPR

292. ETCS = FETCS

293. KBAL = IKBAL

294. ATC = FATC

295. XMRTC = FXMRTC

296. CPCX = FCPCX

297. CON = FCON

298. Go to 171

Reset RL-10 engine nominals.

299. Do 306 NDE = 1, 2

300. Q (NDE) = XQ (NDE)

301. TPIS (NDE) = XTPIS (NDE)

302. XNPN (NDE) = XXNPN (NDE)

303. XNP (NDE) = XXNP (NDE)

304. HPAUX (NDE) = RHPAUX (NDE)

305. PPI (NDE) = RPPI (NDE)

306. RHPI (NDE) = XRHPI (NDE)

307. PCIEL = XPCIE

308. TOL1 = XTOL1

309. TOL2 = XTOL2

310. CON1 = XCON1

311. CON2 = XCON2

312. CON3 = XCON3

313. CON4 = XCON4

314. CON5 = XCON5

315. CD2 = XCD2

316. TOL3 = XTOL3

317. TOL4 = XTOL4

318. TOL5 = XTOL5

319. CON6 = XCON6

320. CON7 = XCON7

321. CON8 = XCON8

322. TOL6 = XTOL6

323. TOL7 = XTOL7

324. TOL8 = XTOL8

325. TOL9 = XTOL9

326. FCL = XFCL

327. AVT = XAVT

328. ETGER = XETGER

329. CON9 = XCON9

```
330.  CON10       = XCON10

331.  CON11       = XCON11

332.  CON12       = XCON12

333.  CON13       = XCON13

334.  CON14       = XCON14

335.  CON15       = XCON15

336.  TOL10       = XTOL10

337.  EPS         = REPS

338.  KBAL        = JKBAL

339.  ATC         = RATC

340.  XMRTC       = RXMRTC

341.  Go to  171

Set J-2 engine nominals.

342.  Do 359   II = 1, 2

343.  ZATI (II)   = ATI (II)

344.  ZPTI (II)   = PTI (II)

345.  ZDLTP (II)  = DLTP (II)

346.  ZDPD (II)   = DPD (II)

347.  ZPPI (II)   = PPI (II)

348.  ZTPIS (II)  = TPIS (II)

349.  ZDPI (II)   = DPI (II)
```

```
350.  ZIPB (II)       = IPB (II)
351.  ZTRSG (II)      = TRSG (II)
352.  ZSG (II)        = SG (II)
353.  ZDEL1 (II)      = DEL1 (II)
354.  ZDEL2 (II)      = DEL2 (II)
355.  ZDW (II)        = DW (II)
356.  ZTTI (II)       = TTI (II)
357.  ZDEL3 (II)      = DEL3 (II)
358.  ZHPAUX (II)     = HPAUX (II)
359.  ZATE (II)       = ATE (II)
360.  Do 364      I   = 1, 6
361.  ZC3 (I)         = C3 (I)
362.  ZC4 (I)         = C4 (I)
363.  ZC5 (I)         = C5 (I)
364.  ZC6 (I)         = C6 (I)
365.  Do 367   NN     = 1, 4
366.  Do 367  NNN     = 1, 2
367.  ZC8(NN, NNN) =  C8(NN, NNN)
368.  Do 370     KK = 1, 6
369.  Do 370    KKK = 1, 2
370.  ZC1(KK, KKK) = C1(KK, KKK)
```

371. ZINDPR = INDPR

372. ZXMRTC = XMRTC

373. ZPNS = PNS

374. ZEPS = EPS

375. ZETCS = ETCS

376. ZATC = ATC

377. ZETCF = ETCF

378. ZPA = PA

379. ZKBAL = KBAL

380. ZFAUX = FAUX

381. ZFEN = FEN

382. ZCNST1 = CNST1

383. ZXMREN = XMREN

384. ZDEL20 = DEL20

385. ZPUVA = PUVA

386. ZDEL7 = DEL7

387. ZDEL21 = DEL21

388. ZDEL22 = DEL22

389. ZDEL23 = DEL23

390. ZDEL24 = DEL24

391. ZDEL25 = DEL25

```
392.  ZDEL26        = DEL26
393.  ZDEL27        = DEL27
394.  ZDEL28        = DEL28
395.  ZDEL29        = DEL29
396.  ZDEL35        = DEL35
397.  ZCNST2        = CNST2
398.  ZCNST3        = CNST3
399.  ZCNST4        = CNST4
400.  ZCNST5        = CNST5
401.  ZCNST6        = CNST6
402.  ZCNST7        = CNST7
403.  ZCNST8        = CNST8
404.  ZEPSM         = EPSM
405.  ZABPNI        = ABPNI
406.  Do 408    KY  = 1, 6
407.  Do 408   KYY  = 1, 10
408.  ZPUCOF(KY, KYY) = PUCOF(KY, KYY)
409.  Do 410    KO  = 1, 10
410.  ZPUSEC(KO)    = PUSEC(KO)
411.  Do 417    IG  = 1, 20
412.  ZDEL4 (IG)    = DEL4 (IG)
```

```
413.  ZDEL5 (IG)   = DEL5 (IG)

414.  ZDRI (IG)    = DRI (IG)

415.  ZDRD (IG)    = DRD (IG)

416.  ZDLTR (IG)   = DLTR (IG)

417.  ZR (IG)      = R (IG)

418.  ZDMOO        = DMOO

419.  ZDMFO        = DMFO

420.  ZWPESF       = WPRESF

421.  ZWPESO       = WPRESO

422.  ZCON         = CON

423.  Go to 171
```

Reset J-2 engine nominals.

```
424.  Do 442    II = 1, 2

425.  ATI (II)     = ZATI (II)

426.  DLTP (II)    = ZDLTP (II)

427.  DPD (II)     = ZDPD (II)

428.  PPI (II)     = ZPPI (II)

429.  TPIS (II)    = ZTPIS (II)

430.  DPI (II)     = ZDPI (II)

431.  IPB (II)     = ZIPB (II)

432.  TRSG (II)    = ZTRSG (II)
```

```
433.   SG (II)        =  ZSG (II)

434.   DEL1 (II)      =  ZDEL1 (II)

435.   DEL2 (II)      =  ZDEL2 (II)

436.   DW (II)        =  ZDW (II)

437.   Continue

438.   TTI (II)       =  ZTTI (II)

439.   DEL3 (II)      =  ZDEL3 (II)

440.   HPAUX (II)     =  ZHPAUX (II)

441.   PTI (II)       =  ZPTI (II)

442.   ATE (II)       =  ZATE (II)

443.   Do 446      I  =  1, 6

444.   C3 (I)         =  ZC3 (I)

445.   C4 (I)         =  ZC4 (I)

446.   C5 (I)         =  ZC5 (I)

447.   C6 (I)         =  ZC6 (I)

448.   Do 450     NN  =  1, 4

449.   Do 450   NNN   =  1, 2

450.   C8(NN, NNN)  =  ZC8(NN, NNN)

451.   Do 453     KK  =  1, 6

452.   Do 453    KKK  =  1, 2

453.   C1(KK, KKK)    =  ZC1(KK, KKK)
```

454. INDPR = ZINDPR
455. XMRTC = ZXMRTC
456. PNS = ZPNS
457. EPS = ZEPS
458. ETCS = ZETCS
459. ATC = ZATC
460. ETCF = ZETCF
461. PA = ZPA
462. KBAL = ZKBAL
463. FAUX = ZFAUX
464. FEN = ZFEN
465. CNST1 = ZCNST1
466. XMREN = ZXMREN
467. DEL20 = ZDEL20
468. PUVA = ZPUVA
469. DEL7 = ZDEL7
470. DEL21 = ZDEL21
471. DEL22 = ZDEL22
472. DEL23 = ZDEL23
473. DEL24 = ZDEL24
474. DEL25 = ZDEL25

```
475.   DEL26          =  ZDEL26

476.   DEL27          =  ZDEL27

477.   DEL28          =  ZDEL28

478.   DEL29          =  ZDEL29

479.   DEL35          =  ZDEL35

480.   CNST2          =  ZCNST2

481.   CNST3          =  ZCNST3

482.   CNST4          =  ZCNST4

483.   CNST5          =  ZCNST5

484.   CNST6          =  ZCNST6

485.   CNST7          =  ZCNST7

486.   CNST8          =  ZCNST8

487.   EPSM           =  ZEPSM

488.   ABPNI          =  ZABPNI

489.   Do 490    KO   =  1, 10

490.   PUSEC (KO)     =  ZPUSEC (KO)

491.   Do 493    KY   =  1, 6

492.   Do 493   KYY   =  1, 10

493.   PUCOF(KY, KYY)  =  ZPUCOF(KY, KYY)

494.   Do 500    IG   =  1, 20

495.   DEL4 (IG)      =  ZDEL4 (IG)
```

496. DEL5 (IG) = ZDEL5 (IG)

497. DRI (IG) = ZDRI (IG)

498. DRD (IG) = ZDRD (IG)

499. DLTR (IG) = ZDLTR (IG)

500. R (IG) = ZR (IG)

501. DMOO = ZDMOO

502. DMFO = ZDMFO

503. WPRESF = ZWPESF

504. WPRESO = ZWPESO

505. CON = ZCON

506. Go to 171

SUBROUTINE SETNOM NOMENCLATURE

Symbol	Description	Units	Reference
ABPNI	By-pass nozzle inlet duct area	in^2	/CJ2IN/, 405, 488
ATC	Thrust chamber throat area	in^2	/CCHAM/, 56, 126, 169, 234, 294, 339, 376, 459
ATE	Turbine exit flange area	in^2	/CTURB/, 24, 88, 194, 258, 359, 442
ATI	Turbine inlet nozzle area	in^2	/CTURB/, 17, 81, 187, 251, 343, 425
AVT	Venturi throat area	in^2	/FLRL10/, 157, 327
AXD	Turbine exhaust discharge area	in^2	/CTURBX/, 65, 231
C1	Array of coefficients for pump performance	—	/CPUMP/, 41, 105, 211, 275, 370, 453
C3	Array of coefficients for specific heat ratio curve	—	/CPUMP/, 26, 90, 196, 261, 361, 444
C4	Array of coefficients for specific heat curve	—	/CGPROP/, 27, 91, 197, 260, 362, 445
C5	Array of coefficients for molecular weight curve	—	/CGPROP/, 28, 92, 198, 262, 363, 446
C6	Array of coefficients for gas generator mixture ratio curve	—	/CGPROP/, 29, 93, 199, 263, 364, 447
C7	Array of coefficients for turbine efficiency curve	—	/CTURB/, 44, 108, 214, 278
C8	Array of coefficients for turbine performance map	—	/CJ2IN/, 367, 450

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
CD2	Lox injector flow loss coefficient	—	/LXRL10/, 145, 315
CNST1	Ratio of thrust chamber injector end pressure to thrust chamber nozzle stagnation pressure	lb/in^2	/CJ2IN/, 382, 465
CNST2	Constant used in calculating pressure drop from fuel injector face to fuel pressurization bleed line	$\frac{lb-sec}{lb-in^2}$	/CJ2IN/, 397, 480
CNST3	Constant used in calculating pressure drop from fuel pressurization bleed line to thrust chamber manifold inlet	$\frac{lb-sec}{lb-in^2}$	/CJ2IN/, 398, 481
CNST4	First guess of fuel turbine exit static pressure	lb/in^2	/CJ2IN/, 399, 482
CNST5	First guess of oxidizer turbine exit static pressure	lb/in^2	/CJ2IN/, 400, 483
CNST6	First guess of by-pass nozzle inlet Mach number	—	/CJ2IN/, 401, 484
CNST7	First guess of by-pass nozzle mass flow rate	lb/sec	/CJ2IN/, 402, 485
CNST8	First guess of oxidizer turbine inlet total pressure	lb/sec	/CJ2IN/, 403, 486
CON	First guess of ratio of turbine mass flow rate to total fuel mass flow rate	—	/INCON/, 63, 124, 229, 297, 422, 505

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
CON1	Constant used in calculating lox pump inlet static pressure	$\frac{sec^2}{ft^3 - in^2}$	/LXRL10/, 140, 310
CON2	Flow loss coefficient for line from lox pump discharge to injector inlet	$\frac{sec^2}{ft^3 - in^2}$	/LXRL10/, 141, 311
CON3	Ground mixture ratio control valve flow loss coefficient	$\frac{sec^2}{ft^3 - in^2}$	/LXRL10/, 142, 312
CON4	Actual lox injector area	in^2	/LXRL10/, 143, 313
CON5	Ratio of lox pump speed to fuel pump speed	—	/LXRL10/, 144, 314
CON6	Constant used in calculating fuel pump inlet static pressure	$\frac{sec^2}{ft^3 - in^2}$	/FLRL10/, 149, 319
CON7	Flow loss coefficient for line from fuel pump discharge to jacket inlet manifold	$\frac{sec^2}{ft^3 - in^2}$	/FLRL10/, 150, 320
CON8	Flow loss coefficient for fuel pump discharge orifice	$\frac{sec^2}{ft^3 - in^2}$	/FLRL10/, 151, 321
CON9	First guess of ratio of turbine mass flow rate to total fuel mass flow rate	—	/TBRL10/, 159, 329
CON10	Flow loss coefficient for line from jacket exit manifold to venturi inlet	$\frac{sec^2}{ft^3 - in^2}$	/FLRL10/, 160, 330

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
CON11	Constant used in computing mass flow rate	—	/TBRL10/, 161, 331
CON12	Ratio of turbine exit static pressure to turbine exit total pressure	—	/TBRL10/, 162, 332
CON13	Flow loss coefficient for line from turbine exit to injector inlet	$\frac{\sec^2}{in^4 - °R}$	/TBRL10/, 163, 333
CON14	Flow loss coefficient for main fuel shut-off valve	$\frac{\sec^2}{in^2 - °R}$	/TBRL10/, 164, 334
CON15	Fuel injector area	in^2	/TBRL10/, 165, 335
CPCX	Ratio of thrust chamber wall pressure to nozzle stagnation pressure	—	/CTURBX/, 58, 122, 236, 296
DEL1	Pump inlet static pressure tolerance	psi	/CPUMP/, 11, 75, 181, 245, 353, 434
DEL2	Pump discharge static pressure tolerance	psi	/CPUMP/, 12, 76, 182, 246, 354, 435
DEL3	Pump speed tolerance	rpm	/CPUMP/, 13, 77, 183, 247, 357, 439
DEL4	Resistance inlet static pressure tolerance	psi	/CRES/, 33, 97, 203, 267, 412, 495
DEL5	Resistance exit density tolerance	lb/ft^3	/CRES/, 34, 98, 204, 268, 413, 496
DEL6	Turbine horsepower tolerance	bhp	/CMAIN/, 23, 87, 193, 257

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DEL7	Engine thrust tolerance	lb	/CCHAM/, 51, 115, 221, 285, 386, 469
DEL8	Turbine exit static pressure tolerance	psi	/H1F1J2/, 53, 117, 223, 287
DEL9	Engine mixture ratio tolerance	—	/H1F1J2/, 47, 111, 217, 281
DEL20	Mixture ratio tolerance	—	/CJ2IN/, 384, 467
DEL21	Oxidizer pump discharge static pressure tolerance	—	/CJ2IN/, 387, 470
DEL22	Oxidizer pump inlet static pressure tolerance	—	/CJ2IN/, 388, 471
DEL23	Fuel turbine horsepower tolerance	—	/CJ2IN/, 389, 472
DEL24	Oxidizer turbine horsepower tolerance	—	/CJ2IN/, 390, 473
DEL25	Turbine exhaust manifold total pressure tolerance	—	/CJ2IN/, 391, 474
DEL26	By-pass nozzle inlet mass flowrate parameter tolerance	—	/CJ2IN/, 392, 475
DEL27	Fuel turbine inlet total pressure tolerance	—	/CJ2IN/, 393, 476
DEL28	A^* equation tolerance	—	/CJ2IN/, 394, 477
DEL29	Oxidizer turbine inlet total pressure tolerance	—	/CJ2IN/, 395, 478
DEL35	Oxidizer turbine mass flowrate tolerance	—	/CJ2IN/, 396, 479

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
DELTX	Turbine exhaust system temperature drop	°R	/CTURBX/, 64, 230
DLTP	Pump propellant temperature rise	°F	/CPUMP/, 14, 78, 184, 248, 345, 426
DLTR	Liquid temperature change across resistance	°F	/CRES/, 35, 99, 205, 269, 416, 499
DMFO	Main fuel orifice diameter	in	/CJ2IN/, 419, 502
DMOO	Main oxidizer orifice diameter	in	/CJ2IN/, 418, 501
DPD	Pump discharge flange diameter	in	/CPUMP/, 10, 74, 180, 244, 346, 427
DPI	Pump inlet flange diameter	in	/CPUMP/, 9, 73, 179, 243, 349, 430
DRD	Resistance exit diameter	in	/CRES/, 32, 96, 202, 266, 415, 498
DRI	Resistance inlet diameter	in	/CRES/, 31, 95, 201, 265, 414, 497
DW	Turbine wheel diameter	in	/CTURB/, 18, 82, 188, 252, 355, 436
EPS	Thrust chamber expansion area ratio	—	/CCHAM/, 59, 119, 167, 225, 289, 337, 374, 457
EPSM	Thrust chamber nozzle expansion ratio at turbine exhaust manifold	—	/CJ2IN/, 404, 487

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
ETAGR	Turbine to pump gearing efficiency	—	/CMAIN/, 22, 86, 192, 256
ETCF	Thrust chamber nozzle efficiency	—	/CCHAM/, 54, 118, 224, 288, 377, 460
ETCS	Thrust chamber combustion efficiency	—	/CCHAM/, 62, 123, 228, 292, 375, 458
ETGER	Gear box efficiency	—	/TBRL10/, 158, 328
FATC	Used for storing ATC	in^2	126, 294
FATE	Used for storing ATE	in^2	88, 258
FATI	Array for storing ATI	in^2	DIM, 81, 251
FAUX	Auxiliary engine thrust	lb	/CCHAM/, 380, 463
FC1	Array for storing C1	—	DIM, 105, 275
FC3	Array for storing C3	—	DIM, 90, 261
FC4	Array for storing C4	—	DIM, 91, 260
FC5	Array for storing C5	—	DIM, 92, 262
FC6	Array for storing C6	—	DIM, 93, 263
FC7	Array for storing C7	—	DIM, 108, 278
FCL	Loaded value of thrust	lb	/CHRL10/, 156, 326
FCON	Used for storing CON	—	124, 297
FCPCX	Used for storing CPCX	—	122, 296
FDEL1	Array for storing DEL1	psi	DIM, 75, 245

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
FDEL2	Array for storing DEL2	psi	DIM, 76, 246
FDEL3	Array for storing DEL3	rpm	DIM, 77, 247
FDEL4	Array for storing DEL4	psi	DIM, 97, 267
FDEL5	Array for storing DEL5	lb/ft^3	DIM, 98, 268
FDEL6	Array for storing DEL6	bhp	DIM, 87, 257
FDEL7	Used for storing DEL7	lb	115, 285
FDEL8	Used for storing DEL8	bhp	117, 287
FDEL9	Used for storing DEL9	—	111, 281
FDLTP	Array for storing DLTP	°F	DIM, 78, 248
FDLTR	Array for storing DLTR	°F	DIM, 99, 269
FDPD	Array for storing DPD	in	DIM, 74, 244
FDPI	Array for storing DPI	in	DIM, 73, 243
FDRD	Array for storing DRD	in	DIM, 96, 266
FDRI	Array for storing DRI	in	DIM, 95, 265
FDW	Used for storing DW	in	DIM, 82, 252
FEN	Nominal engine thrust	lb	/CCHAM/, 50, 114, 220, 284, 381, 464
FEPS	Used for storing EPS	—	119, 289
FETAGR	Array for storing ETAGR	—	DIM, 86, 256
FETCF	Used for storing ETCF	—	118, 288

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
FETCS	Used for storing ETCS	—	123, 292
FFEN	Used for storing FEN	lb	114, 284
FGR	Array for storing GR	—	DIM, 84, 253
FPA	Used for storing PA	psia	116, 286
FPNS	Used for storing PNS	psia	120, 290
FPPI	Array for storing PPI	psia	DIM, 69, 239
FPTES	Array for storing PTES	psia	DIM, 80, 250
FPTI	Array for storing PTI	psia	DIM, 79, 249
FRGG	Used for storing RGG	sec^2/ft^5	109, 279
FR	Array for storing R	sec^2/ft^5	DIM, 100, 270
FRX	Used for storing RX	sec^2/ft^5	113, 283
FSG	Array for storing SG	—	DIM, 72, 242
FTN	Used for storing TN	°F	110, 280
FTPIR	Array for storing TPIR	°R	DIM, 70, 240
FTRSG	Array for storing TRSG	°F	DIM, 71, 241
FWMIS	Array for storing WMIS	lb/sec	DIM, 102, 272
FX	Turbine exhaust thrust	lb	/CTURBX/, 66, 232
FXMREN	Used for storing XMREN	—	112, 282
FXMRTC	Used for storing XMRTC	—	127, 295

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
FXNTST	Array for storing XNTST	—	DIM, 85, 255
GR	Turbine to pump gear ratio	—	/CMAIN/, 20, 84, 189, 253
HPAUX	Auxiliary drive horsepower	bhp	/CMAIN/, 6, 134, 174, 304, 358, 439
IINDPR	Used for storing INDPR	—	61, 227
IIPB	Array for storing IPB	—	DIM, 19, 190
IKBAL	Used for storing KBAL	—	125, 293
INDPR	Propellant combination indicator	—	/CCHAM/, 61, 121, 227, 291, 371, 454
IPB	Pump speed balance indicator	—	/CPUMP/, 19, 83, 190, 254, 350, 411
I	Loop control counter	—	360, 361, 362, 363, 364, 443, 444, 445, 446, 447
II	Loop control indicator	—	25, 26, 27, 28, 29, 89, 90, 91, 92, 93, 342, 343, 344, 345, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 424, 425, 426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 438, 439, 440, 441, 442
III	Loop control indicator	—	30, 31, 32, 33, 34, 35, 36, 94, 95, 96, 97, 98, 99, 100

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
IG	Loop control indicator	—	411, 412, 413, 414, 415, 416, 417, 494, 495, 496, 497, 498, 499, 500
IIK	Loop control indicator	—	200, 201, 202, 203, 204, 205, 206, 264, 265, 266, 267, 268, 269, 270
IN	Loop control indicator	—	39, 41, 103, 105
INI	Loop control indicator	—	173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 238, 239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258
INT	Loop control indicator	—	104, 105
ITI	Loop control indicator	—	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88
J	Indicator to determine which nominals will be set J = 1; H-1 Engine J = 2; F-1 Engine J = 3; J-2 Engine J = 4; RL-10 Engine	—	CALL, 2, 172

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
JJ	Indicator to set or reset the nominals JJ = 0; set nominals JJ = 1; reset nominals	—	CALL, 1
JKBAL	Used for storing KBAL	—	168, 338
KBAL	Engine balance/rebalance indicator	—	/CBAL/, 55, 125, 168, 233, 293, 338, 379, 462
KINDPR	Used for storing INDPR	—	121, 291
KIPB	Array for storing IPB	—	DIM, 83, 254
KK	Loop control counter	—	368, 370, 451, 453
KKBAL	Used for storing KBAL	—	55, 233
KKK	Loop control counter	—	369, 370, 452, 453
KO	Loop control counter	—	409, 410, 489, 490
KY	Loop control counter	—	406, 408, 491, 493
KYY	Loop control counter	—	407, 408, 492, 493
NAA	Loop control indicator	—	212, 214, 276, 278
NBB	Loop control indicator	—	213, 214, 277, 278
NDE	Loop control indicator	—	299, 300, 301, 302, 303, 304, 305, 306
NI	Loop control indicator	—	37, 38, 101, 102
NIT	Loop control indicator	—	42, 44, 106, 108
NN	Loop control indicator	—	365, 367, 448, 450

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
NNI	Loop control indicator	—	195, 196, 197, 198, 199, 259, 260, 261, 262, 263
NNN	Loop control indicator	—	207, 208, 366, 367, 449, 450
NTI	Loop control indicator	—	43, 44, 107, 108
NTT	Loop control indicator	—	129, 130, 131, 132, 133, 134, 135, 136
NWT	Loop control indicator	—	210, 211, 274, 275
NXT	Loop control indicator	—	209, 211, 273, 275
PA	Ambient pressure	psia	/CPA/, 52, 116, 222, 286, 378, 461
PCIEL	Thrust chamber injector end pressure	psia	/MNRL10/, 137, 307
PNS	Thrust chamber nozzle stagnation pressure	psia	/CCHAM/, 60, 120, 226, 290, 373, 456
PPI	Pump inlet total pressure	psia	/CPUMP/, 5, 69, 135, 175, 239, 305, 347, 428
PTES	Turbine exit static pressure	psia	/CTURB/, 16, 80, 186, 250
PTI	Turbine inlet total pressure	psia	/CTURB/, 15, 79, 185, 249, 344, 440
PUCOF	Array of coefficients for PU valve flow loss coefficient	—	/CJ2IN/, 408, 493
PUSEC	Array of PU valve flow loss coefficient curve segment limits	—	/CJ2IN/, 410, 490

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
PUVA	Propellant utilization valve angle	deg	/CJ2IN/, 385, 468
Q	Pump inlet volume flow rate	ft^3/sec	/MNRL10/, 130, 300
R	Resistance value	sec^2/ft^5	/CRES/, 36, 100, 206, 270, 417, 500
RATC	Used for storing ATC	in^2	169, 339
REPS	Used for storing EPS	—	167, 337
RGG	Gas generator resistance	sec^2/ft^5	/H1F1J2/, 45, 109, 215, 279
RHPAUX	Array for storing HPAUX	—	DIM, 134, 304
RHPI	Pump inlet density	lb/ft^3	/CPUMP/, 136, 306
RPPI	Array for storing PPI	psia	DIM, 135, 305
RX	Turbine exhaust system resistance	sec^2/ft^5	/CTURBX/, 49, 113, 219, 283
RXMRTC	Used for storing XMRTC	—	170, 340
SG	Liquid specific gravity at reference temperature	—	/CDEN/, 8, 72, 178, 242, 352, 433
TN	Nominal gas generator combustion temperature	°F	/CMAIN/, 46, 110, 216, 280
TOL1	Fuel pump discharge constant pressure specific heat tolerance	—	/PPRL10/, 138, 308
TOL2	Oxidizer pump discharge constant pressure specific heat tolerance	—	/PPRL10/, 139, 309

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
TOL3	Lox injector flow loss coefficient tolerance	—	/LXRL10/, 146, 316
TOL4	Fuel pump inlet static pressure tolerance	—	/FLRL10/, 147, 317
TOL5	Injector end chamber pressure tolerance	—	/LXRL10/, 148, 318
TOL6	Fuel mass flow rate tolerance	—	/FLRL10/, 152, 322
TOL7	Mixture ratio tolerance	—	/FLRL10/, 153, 323
TOL8	Chamber flow rate tolerance	—	/CHRL10/, 154, 324
TOL9	Thrust tolerance	—	/CHRL10/, 155, 325
TOL10	Mass flow parameter tolerance	—	/TBRL10/, 166, 336
TPIR	Pump inlet propellant temperature	°R	/CMAIN/, 4, 70, 176, 240
TPIS	Pump inlet propellant temperature	°F	/CPUMP/, 131, 301, 348, 429
TRSG	Liquid reference temperature	°F	/CDEN/, 7, 71, 177, 241, 351, 432
TTI	Turbine inlet total temperature	°R	/CTURB/, 356, 438
WMIS	Miscellaneous flow rate	lb/sec	/CMAIN/, 38, 102, 208, 272
WPRESF	Pressurization fuel mass flow rate	lb/sec	/CJ2IN/, 420, 503

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
WPRESO	Pressurization oxidizer mass flow rate	lb/sec	/CJ2IN/, 421, 504
XATC	Used for storing ATC	in^2	56, 234
XATE	Used for storing ATE	in^2	DIM, 24, 194
XATI	Used for storing ATI	in^2	DIM, 17, 187
XAVT	Used for storing AVT	in^2	157, 327
XAXD	Used for storing AXD	in^2	65, 231
XC1	Array for storing C1	—	DIM, 41, 211
XC3	Array for storing C3	—	DIM, 26, 196
XC4	Array for storing C4	—	DIM, 27, 197
XC5	Array for storing C5	—	DIM, 28, 198
XC6	Array for storing C6	—	DIM, 29, 199
XC7	Array for storing C7	—	DIM, 44, 213
XCD2	Used for storing CD2	—	145, 315
XCON	Used for storing CON	—	63, 229
XCON1	Used for storing CON1	$\frac{sec^2}{ft^3 - in^2}$	140, 310
XCON2	Used for storing CON2	$\frac{sec^2}{ft^3 - in^2}$	141, 311
XCON3	Used for storing CON3	$\frac{sec^2}{ft^3 - in^2}$	142, 312

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
XCON4	Used for storing CON4	in^2	143, 313
XCON5	Used for storing CON5	—	144, 314
XCON6	Used for storing CON6	$\frac{sec^2}{ft^3 - in^2}$	149, 319
XCON7	Used for storing CON7	$\frac{sec^2}{ft^3 - in^2}$	150, 320
XCON8	Used for storing CON8	$\frac{sec^2}{ft^3 - in^2}$	151, 321
XCON9	Used for storing CON9	—	159, 329
XCON10	Used for storing CON10	$\frac{sec^2}{ft^3 - in^2}$	160, 330
XCON11	Used for storing CON11	—	161, 331
XCON12	Used for storing CON12	—	162, 332
XCON13	Used for storing CON13	$\frac{sec^2}{ft^3 - in^2}$	163, 333
XCON14	Used for storing CON14	$\frac{sec^2}{ft^3 - in^2}$	164, 334
XCON15	Used for storing CON15	in^2	165, 335
XCPCX	Used for storing CPCX	—	58, 236
XDEL1	Array for storing DEL1	psi	DIM, 11, 181
XDEL2	Array for storing DEL2	psi	DIM, 12, 182

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
XDEL3	Array for storing DEL3	rpm	DIM, 13, 183
XDEL4	Array for storing DEL4	psi	DIM, 33, 203
XDEL5	Array for storing DEL5	lb/ft^3	DIM, 34, 204
XDEL6	Array for storing DEL6	bhp	DIM, 23, 193
XDEL7	Used for storing DEL7	lb	51, 221
XDEL8	Used for storing DEL8	bhp	53, 223
XDEL9	Used for storing DEL9	—	47, 217
XDELTX	Used for storing DELTX	°R	64, 230
XDLTP	Array for storing DLTP	°F	DIM, 14, 184
XDLTR	Array for storing DLTR	°F	DIM, 35, 205
XDPD	Array for storing DPD	in	DIM, 10, 180
XDPI	Array for storing DPI	in	DIM, 9, 179
XDRD	Array for storing DRD	in	DIM, 32, 202
XDRI	Array for storing DRI	in	DIM, 31, 201
XDW	Array for storing DW	in	DIM, 18, 188
XEPS	Used for storing EPS	—	59, 225
XETAGR	Array for storing ETAGR	—	DIM, 22, 192
XETCF	Used for storing ETCF	—	54, 224
XETCS	Used for storing ETCS	—	62, 228

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
XETGER	Used for storing ETGER	—	158, 328
XFCL	Used for storing FCL	lb	156, 326
XFEN	Used for storing FEN	lb	50, 220
XFX	Used for storing FX	lb	66, 232
XGR	Array for storing GR	—	DIM, 20, 189
XHPAUX	Array for storing HPAUX	bhp	DIM, 6, 174
XMREN	Nominal engine mixture ratio	—	/H1F1J2/, 48, 112, 218, 282, 383, 466
XMRTC	Thrust chamber mixture ratio	—	/CCHAM/, 57, 127, 170, 235, 295, 340, 372, 465
XNPN	Nominal pump speed	rpm	/PPRL10/, 132, 302
XNP	Pump speed	rpm	/CPUMP/, 133, 303
XNTST	Number of turbine pressure stages	—	/CTURB/, 21, 85, 191, 255
XPA	Used for storing PA	psia	52, 222
XPCIE	Used for storing PCIE	psia	137, 307
XPNS	Used for storing PNS	psia	60, 226
XPPI	Array for storing PPI	psia	DIM, 5, 175
XPTES	Array for storing PTES	psia	DIM, 16, 186
XPTI	Array for storing PTI	psia	DIM, 15, 185
XQ	Array for storing Q	ft^3/sec	DIM, 130, 300

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
XR	Array for storing R	sec^2/ft^5	DIM, 36, 206
XRGG	Used for storing RGG	sec^2/ft^5	45, 215
XRHPI	Array for storing RHPI	lb/ft^3	DIM, 136, 306
XRX	Used for storing RX	sec^2/ft^5	49, 219
XSG	Array for storing SG	—	DIM, 8, 178
XTN	Used for storing TN	°F	46, 216
XTOL1	Used for storing TOL1	—	138, 308
XTOL2	Used for storing TOL2	—	139, 309
XTOL3	Used for storing TOL3	—	146, 316
XTOL4	Used for storing TOL4	—	147, 317
XTOL5	Used for storing TOL5	—	148, 318
XTOL6	Used for storing TOL6	—	152, 322
XTOL7	Used for storing TOL7	—	153, 323
XTOL8	Used for storing TOL8	—	154, 324
XTOL9	Used for storing TOL9	—	155, 325
XTOL10	Used for storing TOL10	—	166, 336
XTPIR	Array for storing TPIR	°R	DIM, 4, 176
XTPIS	Array for storing TPIS	°F	DIM, 131, 301
XTRSG	Array for storing TRSG	°R	DIM, 7, 177

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
XWMIS	Array for storing WMIS	lb/sec	DIM, 38, 208
XXMREN	Used for storing XMREN	—	48, 218
XXMRTC	Used for storing XMRTC	—	57, 235
XXNP	Array for storing XNP	rpm	DIM, 133, 303
XXNPN	Array for storing XNPN	rpm	DIM, 132, 302
XXNTST	Array for storing XNTST	—	DIM, 21, 192
ZATC	Used for storing ATC	in^2	374, 459
ZATE	Used for storing ATE	in^2	DIM, 359, 442
ZATI	Array for storing ATI	in^2	DIM, 343, 425
ZC1	Array for storing C1	—	DIM, 370, 453
ZC3	Array for storing C3	—	DIM, 361, 444
ZC4	Array for storing C4	—	DIM, 362, 445
ZC5	Array for storing C5	—	DIM, 363, 446
ZC6	Array for storing C6	—	DIM, 364, 447
ZC8	Array for storing C8	—	DIM, 367, 450
ZCNST1	Used for storing CNST1	lb/in^2	382, 465
ZDEL1	Array for storing DEL1	psi	DIM, 353, 434
ZDEL2	Array for storing DEL2	psi	DIM, 354, 435
ZDEL3	Array for storing DEL3	rpm	DIM, 357, 439

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
ZDEL4	Array for storing DEL4	psi	DIM, 412, 495
ZDEL5	Array for storing DEL5	lb/ft^3	DIM, 413, 496
ZDEL7	Used for storing DEL7	lb	386, 469
ZDEL20	Used for storing DEL20	—	384, 467
ZDEL21	Used for storing DEL21	—	387, 470
ZDEL22	Used for storing DEL22	—	388, 471
ZDEL23	Used for storing DEL23	—	389, 472
ZDEL24	Used for storing DEL24	—	390, 473
ZDEL25	Used for storing DEL25	—	391, 474
ZDEL26	Used for storing DEL26	—	392, 475
ZDEL27	Used for storing DEL27	—	393, 476
ZDEL28	Used for storing DEL28	—	394, 477
ZDEL29	Used for storing DEL29	—	395, 478
ZDEL35	Used for storing DEL35	—	396, 479
ZCNST2	Used for storing CNST2	—	397, 480
ZCNST3	Used for storing CNST3	—	398, 481
ZCNST4	Used for storing CNST4	—	399, 482
ZCNST5	Used for storing CNST5	—	400, 483
ZCNST6	Used for storing CNST6	—	401, 484

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
ZCNST7	Used for storing CNST7	—	402, 485
ZCNST8	Used for storing CNST8	—	403, 486
ZCON	Used for storing CON	—	422, 505
ZABPNI	Used for storing ABPNI	—	405, 488
ZDLTP	Array for storing DLTP	°F	DIM, 345, 426
ZDLTR	Array for storing DLTR	°F	DIM, 416, 499
ZDMFO	Used for storing DMFO	in	419, 502
ZDMOO	Used for storing DMOO	in	418, 501
ZDPD	Array for storing DPD	in	DIM, 346, 427
ZDPI	Array for storing DPI	in	DIM, 349, 430
ZDRD	Array for storing DRD	in	DIM, 415, 498
ZDRI	Array for storing DRI	in	DIM, 414, 497
ZDW	Array for storing DW	in	DIM, 355, 436
ZEPS	Used for storing EPS	—	374, 457
ZEPSM	Used for storing EPSM	—	404, 487
ZETCF	Used for storing ETCF	—	377, 460
ZETCS	Used for storing ETCS	—	375, 458
ZFAUX	Used for storing FAUX	lb	380, 463
ZFEN	Used for storing FEN	lb	381, 464

SUBROUTINE SETNOM NOMENCLATURE (cont.)

Symbol	Description	Units	Reference
ZHPAUX	Array for storing HPAUX	—	DIM, 358, 440
ZINDPR	Used for storing INDPR	—	371, 454
ZIPB	Array for storing IPB	—	DIM, 350, 431
ZKBAL	Used for storing KBAL	—	379, 462
ZPA	Used for storing PA	psia	380, 463
ZPNS	Used for storing PNS	psia	373, 455
ZPPI	Array for storing PPI	psia	DIM, 347, 428
ZPTI	Array for storing PTI	psia	DIM, 344, 441
ZPUCOF	Array for storing PUCOF	—	DIM, 408, 493
ZPUSEC	Array for storing PUSEC	—	DIM, 410, 490
ZPUVA	Used for storing PUVA	—	385, 468
ZR	Array for storing R	sec^2/ft^5	DIM, 417, 500
ZSG	Array for storing SG	—	DIM, 352, 433
ZTPIS	Array for storing TPIS	°F	DIM, 348, 429
ZTRSG	Array for storing TRSG	°F	DIM, 351, 432
ZTTI	Array for storing TTI	°R	DIM, 356, 438
ZWPESF	Used for storing WPRESF	lb/sec	420, 503
ZWPESO	Used for storing WPRESO	lb/sec	421, 504
ZXMREN	Used for storing XMREN	—	383, 466
ZXMRTC	Used for storing XMRTC	—	372, 455

```
$IBFTC SETNØM  FULIST,REF
      SUBRØUTINE  SETNØM(J,JJ)
      DIMENSIØN ZPUCØF(6,10) , ZPUSEC(10), ZDEL4(20), ZDEL5(20), ZDRI(20
     A ), ZDRD(20), ZDLTR(20), ZR(20)
      DIMENSIØN ZTTI(2), ZC3(6), ZC4(6),ZC5(6),ZC6(6), ZATI(2), ZPTI(2),
     1 ZDLTP(2), ZDPD(2), ZPPI(2), ZTPIS(2), ZDPI(2), ZTRSG(2),ZIPB(2),
     2 ZSG(2), ZDEL1(2),ZDEL2(2),ZDEL3(2), ZC1(6,2),ZC8(4,2), ZDW (2),
     3 ZATE(2)
      DIMENSIØN XPPI(2),XTPIR(2),XTRSG(2),XSG(2),XDPI(2),XDPD(2),
     1 XDEL2(2), XDEL1(2), XDEL3(2),  XDLTP(2),XPTI(2), XPTES(2),
     3 XATI(2), XDW(2), IIPB(2),XGR(2), XXNTST(2),XETAGR(2),XDEL6(2),
     5 XATE(2),XC3(6),XC4(6), XC5(6), XC6(6), XDRI(20), XDRD(20),
     7 XDEL4(20),XDEL5(20),XDLTR(20),XR(20), XWMIS(4), XC1(6,2),
     9 XC7(9,2),XHPAUX(2),XQ(2),XTPIS(2),XXNPN(2),XXNP(2),XRHPI(2)    ,
     F FPPI(2), FTPIR(2), FTRSG(2),FSG(2), FDPI(2), FDPD(2), FDEL1(2),
     G FDEL2(2), FDEL3(2), FDLTP(2), FPTI(2), FPTES(2),  FATI(2), FDW(2)
     H, KIPB(2), FGR(2), FXNTST(2), FETAGR(2), FDEL6(2), FATE(2),
     I FC3(6), FC4(6), FC5(6), FC6(6), FDRI(20),  FDRD(20 ), FDEL4(20),
     J FDEL5(20), FDLTR(20), FR(20),  FWMIS(4), FC1(6,2 ), FC7(9,2 ),
     L RPPI(2),RHPAUX(2)
      DIMENSIØN ZHPAUX(2)
      CØMMØN /CTURMP/ PP(2),C8(4,2)
      CØMMØN /CJ2IN/DEL20,DEL21,DEL22,DEL23,DEL24,DEL25,DEL26,CNST1,
     1CNST2,CNST3,CNST4,CNST5,CNST6,PUCØF(6,10),PUSEC(10),PUVA,
     2 EPSM,ABPNI, DMØØ, DMFØ  ,WPRESF, WPRESØ ,DEL27,DEL28,DEL29,CNST7
     3,DEL35
      CØMMØN /CJ2ØUT/ RPU,PV,WPUØ,PMAN,PMAN1,TMAN,TAVE,WBP,
     1PBPNI,FPL,FPR,XMBPNI,ABPN,DBPN,PTIS(2),TTIS(2)
      CØMMØN /PNSPRI/ PNSPR
C
      CØMMØN /MNRL10/ PCIEL,Q(2),XMRTCL,PII(2)
C
      CØMMØN /LXRL10/ TØL3(1),TØL5(1),CD2(1),CØN1(1),CØN2(1),CØN3(1),
     1CØN4(1),CØN5(1)
C
      CØMMØN /FLRL10/ CØN8(1),CØN6(1),TØL4(1),FLØPT(1),CØN7(1),TØJE(1),
     1WCF1(1),TØL6(1),TØL7(1),CØN10(1),AVT(1)
C
      CØMMØN /CHRL10/ TØL8(1),TØL9(1),FCL(1)
C
      CØMMØN /PPRL10/ XRATIØ(2), TØL1, TØL2, XNPN(2)
C
      CØMMØN /TBRL10/CØN9(1),ETGER(1),CØN11(1),CØN12(1),CØN13(1),CØN14(1
     1),CØN15(1),TØL10(1), HP1(1),HP2(1)
C
      CØMMØN /CPUMP/    PPI(2),TPIS(2),XNPSH(2),      PVPI(2),RHPI(2),
     1VHPI(2),WP(2),DPI(2),DEL1(2),TPDS(2),DLTP(2),PPDS(2),PPD(2),PVPD(2
     2),RHPD(2),VHPD(2),DPD(2),DEL2(2),RHP(2),QP(2),H(2),XNP(2),C1(6,2),
     3IPB(2),XNPP(2),DEL3(2),HPRIM(2),ETAP(2),XNS(2),XNSS(2),BHPP(2),
     4PPIS(2),IPBR
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM

```
C
      CØMMØN /CGPRØP/ C3(6),C4(6),C5(6),C6(6),CSG(1),GAMG(1),CPG(1),
     1XMWG(1),XMRG(1),RG(1),XØMG(1)
C
      CØMMØN /CRES/ PRI(20),TRIS(20),PVRI(20),RHRI(20),VHRI(20),WRD(20),
     1DRI(20),DEL4(20),TRDS(20),DLTR(20),RHRD(20),RHRA(20),PRD(20),
     2R(20),VHRD(20),DRD(20),PRDS(20),PVRD(20),DEL5(20),WRI(20),PRIS(20)
C
      CØMMØN /CTURB/   PTI(2),PTES(2),WT(2),ATI(2),CST(2),DELHT(2),
     1CPT(2),TTI(2),GAMT(2),UVRAT(2),DW(2),XNT(2),ETAT(2),C7(9,2),HPT(2)
     2,XNTST(2),TTDS(2),XMTE(2),ATE(2),RGT(2),PTE(2),TTD(2),PRT(2)
C
      CØMMØN /CTURBX/ TXD(1),TXI(1), DELTX(1),PXDS(1),PXIS(1),RX(1),
     1RGX(1),TXIS(1),WX(1),CSXD(1),XØMX(1),GAMX(1),XMXD(1),PRXD(1),
     2PXD(1),AXD(1),FX(1),TXDS(1),CPCX(1)
C
      CØMMØN /CCHAM/ INDPR(1),CF(1),CFTHE(1),ETCF(1),EPS(1),PNS(1),FC(1)
     1,ATC(1),FAUX(1),FEN(1),DEL7(1),CS(1),CSTHE(1),ETCS(1),WC(1),XMRTC(
     21),WCF(1),WCØ(1)
C
      CØMMØN /CBAL/ KBAL(1)  /CPA/ PA(1) /CRESGG/ RESØG(1),RESFG(1)
C
      CØMMØN /CMAIN/ WMIS(4),HPAUX(2),ETAGR(2),DEL6(2),GR(2),PPDA(2),
     1TN(1),ØRES(4),WG(1),WGØ(1),WGF(1),PCIE(1),XMRE(1),PGG(1),XISPC(1),
     2XISPE(1),FE(1),WEØ(1),WEF(1),TPIR(2)
C
      CØMMØN /CDEN/TRSG(2),SG(2)
C
      CØMMØN /H1F1J2/ RGG(1),DEL8(1),DEL9(1),XMREN(1)
C
      CØMMØN /INCØN/ CØN
C
      CØMMØN /STAGES/ISTG(6)
C
      CØMMØN /INFLTB/ YY(10,10,6),VINDA(10,6),VDEPA(10,6),NINDA(6),NIND,
     1NDA(6),Y(10,10),VDEP(10),VIND(10),ND,EAE(3,6)
      INTEGER ZINDPR, KBAL, IPB,ZIT
      IF (JJ .GT. 0) GØ TØ 110
      GØ TØ (30,32,98,100),J
   30 DØ 35ITI=1,2
      XTPIR(ITI)=TPIR(ITI)
      XPPI(ITI)=PPI(ITI)
      XHPAUX(ITI)=HPAUX(ITI)
      XTRSG(ITI)=TRSG(ITI)
      XSG(ITI) = SG(ITI)
      XDPI(ITI)= DPI(ITI)
      XDPD( ITI) =DPD(ITI)
      XDEL1(ITI)=DEL1(ITI)
      XDEL2(ITI)= DEL2(ITI)
      XDEL3(ITI)= DEL3(ITI)
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
   XDLTP(ITI)= DLTP(ITI)
   XPTI(ITI) = PTI(ITI)
   XPTES(ITI)= PTES(ITI)
   XATI(ITI) = ATI(ITI)
   XDW(ITI) = DW(ITI)
   IIPB(ITI)= IPB(ITI)
   XGR(ITI) = GR(ITI)
   XXNTST(ITI)= XNTST(ITI)
   XETAGR(ITI)= ETAGR(ITI)
   XDEL6(ITI)= DEL6(ITI)
35 XATE(ITI)= ATE(ITI)
40 DØ 45II=1,6
   XC3(II)= C3(II)
   XC4(II)= C4(II)
   XC5(II)= C5(II)
45 XC6(II)= C6(II)
50 DØ 55III=1,20
   XDRI(III)= DRI(III)
   XDRD(III)= DRD(III)
   XDEL4(III)=DEL4(III)
   XDEL5(III)=DEL5(III)
   XDLTR(III)= DLTR(III)
55 XR(III)= R(III)
60 DØ 65NI=1,4
65 XWMIS(NI)= WMIS(NI)
70 DØ 75IN=1,6
   DØ 75INT=1,2
75 XC1(IN,INT)= C1(IN,INT)
80 DØ 85NIT=1,9
   DØ 85NTI=1,2
85 XC7(NIT,NTI)=C7(NIT,NTI)
   XRGG= RGG
   XTN=TN
   XDEL9= DEL9
   XXMREN =XMREN
   XRX =RX
   XFEN= FEN
   XDEL7 =DEL7
   XPA =PA
   XDEL8 =DEL8
   XETCF=ETCF
   KKBAL =KBAL
   XATC =ATC
   XXMRTC= XMRTC
   XCPCX=CPCX
   XEPS= EPS
   XPNS= PNS
   IINDPR= INDPR
   XETCS= ETCS
   XCØN= CØN
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
      XDELTX=DELTX
      XAXD=AXD
      XFX=FX
      GO TO 99
   32 DO 36ITI=1,2
      FPPI(ITI)=PPI(ITI)
      FTPIR(ITI)=TPIR(ITI)
      FTRSG(ITI)=TRSG(ITI)
      FSG(ITI) = SG(ITI)
      FDPI(ITI)= DPI(ITI)
      FDPD( ITI) =DPD(ITI)
      FDEL1(ITI)=DEL1(ITI)
      FDEL2(ITI)= DEL2(ITI)
      FDEL3(ITI)= DEL3(ITI)
      FDLTP(ITI)= DLTP(ITI)
      FPTI(ITI) = PTI(ITI)
      FPTES(ITI)= PTES(ITI)
      FATI(ITI) = ATI(ITI)
      FDW(ITI) = DW(ITI)
      KIPB(ITI)= IPB(ITI)
      FGR(ITI) = GR(ITI)
      FXNTST(ITI)= XNTST(ITI)
      FETAGR(ITI)= ETAGR(ITI)
      FDEL6(ITI)= DEL6(ITI)
   36 FATE(ITI)= ATE(ITI)
   42 DO 46II=1,6
      FC3(II)= C3(II)
      FC4(II)= C4(II)
      FC5(II)= C5(II)
   46 FC6(II)= C6(II)
   52 DO 56III=1,20
      FDRI(III)= DRI(III)
      FDRD(III)= DRD(III)
      FDEL4(III)=DEL4(III)
      FDEL5(III)=DEL5(III)
      FDLTR(III)= DLTR(III)
   56 FR(III)= R(III)
   62 DO 66NI=1,4
   66 FWMIS(NI)= WMIS(NI)
   72 DO 76IN=1,6
      DO 76INT=1,2
   76 FC1(IN,INT)= C1(IN,INT)
   82 DO 86NIT=1,9
      DO 86NTI=1,2
   86 FC7(NIT,NTI)=C7(NIT,NTI)
   89 FRGG= RGG
      FTN=TN
      FDEL9= DEL9
      FXMREN =XMREN
      FRX =RX
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
      FFEN= FEN
      FDEL7 =DEL7
      FPA =PA
      FDEL8 =DEL8
      FETCF=ETCF
      FEPS= EPS
      FPNS= PNS
      KINDPR= INDPR
      FCPCX=CPCX
      FETCS= ETCS
      FCØN= CØN
      IKBAL =KBAL
      FATC =ATC
      FXMRTC= XMRTC
      GØ TØ 99
100   DØ 105NTT=1,2
      XQ(NTT) =Q(NTT)
      XTPIS(NTT)=TPIS(NTT)
      XXNPN(NTT)=XNPN(NTT)
      XXNP(NTT) =XNP(NTT)
      RHPAUX(NTT)=HPAUX(NTT)
      RPPI(NTT)= PPI(NTT)
105   XRHPI(NTT)=RHPI(NTT)
      XPCIE= PCIEL
      XTØL1= TØL1
      XTØL2= TØL2
      XCØN1= CØN1
      XCØN2= CØN2
      XCØN3= CØN3
      XCØN4= CØN4
      XCØN5=CØN5
      XCD2= CD2
      XTØL3= TØL3
      XTØL4= TØL4
      XTØL5= TØL5
      XCØN6= CØN6
      XCØN7= CØN7
      XCØN8= CØN8
      XTØL6= TØL6
      XTØL7= TØL7
      XTØL8= TØL8
      XTØL9= TØL9
      XFCL= FCL
      XAVT= AVT
      XETGER= ETGER
      XCØN9= CØN9
      XCØN10= CØN10
      XCØN11=CØN11
      XCØN12= CØN12
      XCØN13= CØN13
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
    XCØN14= CØN14
    XCØN15= CØN15
    XTØL10= TØL10
    REPS=EPS
    JKBAL =KBAL
    RATC =ATC
    RXMRTC= XMRTC
 99 RETURN
110 GØ TØ (120,122,196,190) , J
120 DØ 125INI=1,2
    HPAUX(INI)= XHPAUX(INI)
    PPI(INI)= XPPI(INI)
    TPIR(INI)= XTPIR(INI)
    TRSG(INI)= XTRSG(INI)
    SG(INI)=   XSG(INI)
    DPI(INI)= XDPI(INI)
    DPD(INI)= XDPD(INI)
    DEL1(INI)= XDEL1(INI)
    DEL2(INI)= XDEL2(INI)
    DEL3(INI)= XDEL3(INI)
    DLTP(INI)= XDLTP(INI)
    PTI (INI)= XPTI(INI)
    PTES(INI)= XPTES(INI)
    ATI(INI)=  XATI(INI)
    DW(INI)=   XDW(INI)
    GR(INI)=   XGR(INI)
    IPB(INI)=  IIPB(INI)
    XNTST(INI)=XXNTST(INI)
    ETAGR(INI)=XETAGR(INI)
    DEL6(INI)= XDEL6(INI)
125 ATE(INI)= XATE(INI)
130 DØ 135NNI=1,6
    C3(NNI)= XC3(NNI)
    C4(NNI)= XC4(NNI)
    C5(NNI)= XC5(NNI)
135 C6(NNI)= XC6(NNI)
    DØ 145IIK=1,20
    DRI(IIK)=  XDRI(IIK)
    DRD(IIK)=  XDRD(IIK)
    DEL4(IIK)= XDEL4(IIK)
    DEL5(IIK)= XDEL5(IIK)
    DLTR(IIK)  XDLTR(IIK)
145 R(IIK)= XR(IIK)
150 DØ 155NNN=1,4
155 WMIS(NNN)= XWMIS(NNN)
160 DØ 165NXT=1,6
    DØ 165NWT=1,2
165 C1(NXT,NWT)=XC1(NXT,NWT)
170 DØ 175NAA=1,9
    DØ 175NBB=1,2
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
175 C7(NAA,NBB)= XC7(NAA,NBB)
    RGG= XRGG
    TN=  XTN
    DEL9= XDEL9
    XMREN= XXMREN
    RX= XRX
    FEN= XFEN
    DEL7= XDEL7
    PA= XPA
    DEL8= XDEL8
    ETCF= XETCF
    EPS= XEPS
    PNS= XPNS
    INDPR= IINDPR
    ETCS= XETCS
    CØN= XCØN
    DELTX= XDELTX
    AXD= XAXD
    FX= XFX
    KBAL= KKBAL
    ATC= XATC
    XMRTC= XXMRTC
    CPCX=XCPCX
    GØ TØ 99
122 DØ 126INI=1,2
    PPI(INI)= FPPI(INI)
    TPIR(INI)= FTPIR(INI)
    TRSG(INI)= FTRSG(INI)
    SG(INI)=   FSG(INI)
    DPI(INI)= FDPI(INI)
    DPD(INI)= FDPD(INI)
    DEL1(INI)= FDEL1(INI)
    DEL2(INI)= FDEL2(INI)
    DEL3(INI)= FDEL3(INI)
    DLTP(INI)= FDLTP(INI)
    PTI (INI)= FPTI(INI)
    PTES(INI)= FPTES(INI)
    ATI(INI)=  FATI(INI)
    DW(INI)=   FDW(INI)
    GR(INI)=   FGR(INI)
    IPB(INI)=  KIPB(INI)
    XNTST(INI)=FXNTST(INI)
    ETAGR(INI)=FETAGR(INI)
    DEL6(INI)= FDEL6(INI)
126 ATE(INI)= FATE(INI)
132 DØ 136NNI=1,6
    C4(NNI)= FC4(NNI)
    C3(NNI)= FC3(NNI)
    C5(NNI)= FC5(NNI)
136 C6(NNI)= FC6(NNI)
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
142 D0 146IIK=1,20
    DRI(IIK)=  FDRI(IIK)
    DRD(IIK)=  FDRD(IIK)
    DEL4(IIK)= FDEL4(IIK)
    DEL5(IIK)= FDEL5(IIK)
    DLTR(IIK)= FDLTR(IIK)
146 R(IIK)= FR(IIK)
152 D0 156NNN=1,4
156 WMIS(NNN)= FWMIS(NNN)
162 D0 166NXT=1,6
    D0 166NWT=1,2
166 C1(NXT,NWT)=FC1(NXT,NWT)
172 D0 176NAA=1,9
    D0 176NBB=1,2
176 C7(NAA,NBB)= FC7(NAA,NBB)
    RGG= FRGG
    TN=  FTN
    DEL9= FDEL9
    XMREN= FXMREN
    RX= FRX
    FEN= FFEN
    DEL7= FDEL7
    PA= FPA
    DEL8= FDEL8
    ETCF= FETCF
    EPS= FEPS
    PNS= FPNS
    INDPR= KINDPR
    ETCS= FETCS
    KBAL= IKBAL
    ATC= FATC
    XMRTC= FXMRTC
    CPCX=FCPCX
    C0N= FC0N
    G0 T0 99
190 D0 195NDE=1,2
    Q(NDE)= XQ(NDE)
    TPIS(NDE)= XTPIS(NDE)
    XNPN(NDE)= XXNPN(NDE)
    XNP(NDE)= XXNP(NDE)
    HPAUX(N DE)= RHPAUX(NDE)
    PPI(NDE)= RPPI(NDE)
195 RHPI(NDE)= XRHPI(NDE)
    PCIEL = XPCIE
    T0L1= XT0L1
    T0L2= XT0L2
    C0N1= XC0N1
    C0N2= XC0N2
    C0N3= XC0N3
    C0N4= XC0N4
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
      CØN5= XCØN5
      CD2=XCD2
      TØL3= XTØL3
      TØL4= XTØL4
      TØL5= XTØL5
      CØN6= XCØN6
      CØN7= XCØN7
      CØN8= XCØN8
      TØL6= XTØL6
      TØL7= XTØL7
      TØL8= XTØL8
      TØL9= XTØL9
      FCL= XFCL
      AVT= XAVT
      ETGER= XETGER
      CØN9= XCØN9
      CØN10= XCØN10
      CØN11= XCØN11
      CØN12= XCØN12
      CØN13= XCØN13
      CØN14= XCØN14
      CØN15= XCØN15
      TØL10= XTØL10
      EPS=REPS
      KBAL=JKBAL
      ATC=RATC
      XMRTC=RXMRTC
      GØ TØ 99
   98 DØ 198II=1,2
      ZATI(II)=ATI(II)
      ZPTI(II) = PTI(II)
      ZDLTP(II)=DLTP(II)
      ZDPD(II)=DPD(II)
      ZPPI(II)=PPI(II)
      ZTPIS(II)=TPIS(II)
      ZDPI(II)=DPI(II)
      ZIPB(II)=IPB(II)
      ZTRSG(II)=TRSG(II)
      ZSG(II)=SG(II)
      ZDEL1(II)=DEL1(II)
      ZDEL2(II)=DEL2(II)
      ZDW(II)=DW(II)
      ZTTI(II) = TTI(II)
      ZDEL3(II) = DEL3(II)
      ZHPAUX(II) = HPAUX(II)
  198 ZATE(II)=ATE(II)
      DØ 200 I=1,6
      ZC3(I)=C3(I)
      ZC4(I)=C4(I)
      ZC5(I)=C5(I)
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
  200 ZC6(I)=C6(I)
      DO 205NN=1,4
      DO 205NNN=1,2
  205 ZC8(NN,NNN)=C8(NN,NNN)
      DO 210KK=1,6
      DO 210KKK=1,2
  210 ZC1(KK,KKK)=C1(KK,KKK)
      ZINDPR =INDPR
      ZXMRTC=XMRTC
      ZPNS= PNS
      ZEPS= EPS
      ZETCS=ETCS
      ZATC=ATC
      ZETCF=ETCF
      ZPA=PA
      ZKBAL=KBAL
      ZFAUX=FAUX
      ZFEN=FEN
      ZCNST1=CNST1
      ZXMREN=XMREN
      ZDEL20=DEL20
      ZPUVA=PUVA
      ZDEL7=DEL7
      ZDEL21=DEL21
      ZDEL22= DEL22
      ZDEL23 = DEL23
      ZDEL24 = DEL24
      ZDEL25 = DEL25
      ZDEL26 = DEL26
      ZDEL27 = DEL27
      ZDEL28 = DEL28
      ZDEL29 = DEL29
      ZDEL35 = DEL35
      ZCNST2 = CNST2
      ZCNST3 = CNST3
      ZCNST4 = CNST4
      ZCNST5 = CNST5
      ZCNST6 = CNST6
      ZCNST7 = CNST7
      ZEPSM = EPSM
      ZABPNI = ABPNI
      DO 214KY=1,6
      DO 214KYY=1,10
  214 ZPUCOF(KY,KYY)=PUCOF(KY,KYY)
      DO 212KO=1,10
  212 ZPUSEC(KO)=PUSEC(KO)
      DO 197 IG=1,20
      ZDEL4(IG)= DEL4(IG)
      ZDEL5(IG)= DEL5(IG)
      ZDRI(IG)= DRI(IG)
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
      ZDRD(IG)= DRD(IG)
      ZDLTR(IG)= DLTR(IG)
  197 ZR(IG)= R(IG)
      ZDMOO= DMOO
      ZDMFO= DMFO
      ZWPESF= WPRESF
      ZWPESO= WPRESO
      ZCON = CON
      GO TO 99
  196 DO 215II=1,2
      ATI(II)=ZATI(II)
      DLTP(II)=ZDLTP(II)
      DPD(II)=ZDPD(II)
      PPI(II)=ZPPI(II)
      TPIS(II)=ZTPIS(II)
      DPI(II)=ZDPI(II)
      IPB(II)=ZIPB(II)
      TRSG(II)=ZTRSG(II)
      SG(II)=ZSG(II)
      DEL1(II)=ZDEL1(II)
      DEL2(II)=ZDEL2(II)
      DW(II)=ZDW(II)
      TTI(II) = ZTTI(II)
      DEL3(II) = ZDEL3(II)
      HPAUX(II) = ZHPAUX(II)
      PTI(II) = ZPTI(II)
  215 ATE(II)=ZATE(II)
      DO 218 I=1,6
      C3(I)=ZC3(I)
      C4(I)=ZC4(I)
      C5(I)=ZC5(I)
  218 C6(I)=ZC6(I)
      DO 225NN=1,4
      DO 225NNN=1,2
  225 C8(NN,NNN)=ZC8(NN,NNN)
      DO 230KK=1,6
      DO 230KKK=1,2
  230 C1(KK,KKK)=ZC1(KK,KKK)
      INDPR=ZINDPR
      XMRTC=ZXMRTC
      PNS=ZPNS
      EPS=ZEPS
      ETCS=ZETCS
      ATC=ZATC
      ETCF=ZETCF
      PA=ZPA
      KBAL=ZKBAL
      FAUX=ZFAUX
      FEN=ZFEN
      CNST1=ZCNST1
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

```
      XMREN=ZXMREN
      DEL20= ZDEL20
      PUVA=ZPUVA
      DEL7=ZDEL7
      DEL21=ZDEL21
      DEL22=ZDEL22
      DEL23 = ZDEL23
      DEL24 = ZDEL24
      DEL25 = ZDEL25
      DEL26 = ZDEL26
      DEL27 = ZDEL27
      DEL28 = ZDEL28
      DEL29 = ZDEL29
      DEL35 = ZDEL35
      CNST2 = ZCNST2
      CNST3 = ZCNST3
      CNST4 = ZCNST4
      CNST5 = ZCNST5
      CNST6 = ZCNST6
      CNST7 = ZCNST7
      EPSM =ZEPSM
      ABPNI = ZABPNI
      DØ 232KØ=1,6
  232 PUSEC(KØ)=ZPUSEC(KØ)
      DØ 234KY=1,6
      DØ 234KYY=1,10
  234 PUCØF(KY,KYY)=ZPUCØF(KY,KYY)
      DØ 260 IG=1,20
      DEL4(IG)= ZDEL4(IG)
      DEL5(IG)= ZDEL5(IG)
      DRI(IG)= ZDRI(IG)
      DRD(IG)= ZDRD(IG)
      DLTR(IG)= ZDLTR(IG)
  260 R(IG)= ZR(IG)
      DMØØ= ZDMØØ
      DMFØ= ZDMFØ
      WPRESF= ZWPESF
      WPRESØ= ZWPESØ
      CØN = ZCØN
      GØ TØ 99
      END
```

FIGURE 109. SYMBOLIC LISTING OF SUBROUTINE SETNOM (cont.)

LXXXIX. SUBROUTINE BDATA

This subroutine "presets" certain data in the COMMON blocks used by both the Engine Balance/Rebalance Program (BLOCK I) and the trajectory simulation (BLOCK II).

CALLING SEQUENCE

This subroutine is not executable, it has no calling sequence. It is composed only of data values to be loaded with the rest of the program before execution.

USAGE

COMMON Block /STAGES/

ISTG(1) = 0

ISTG(2) = 0

ISTG(3) = 0

ISTG(4) = 0

ISTG(5) = 0

ISTG(6) = 0

SUBROUTINE BDATA NOMENCLATURE

Symbol	Description	Units	Reference
ISTG	Indicator for influence coefficients generated by engine balance/rebalance program	—	/STAGES/, DATA

```
$IBFTC BDATA   FULIST,REF
      BLØCK DATA
      CØMMØN /STAGES/ISTG(6)
      CØMMØN /INFLTB/ YY(10,10,6),VINDA(10,6),VDEPA(10,6),NINDA(6),NIND,
     1NDA(6),Y(10,10),VDEP(10),VIND(10),ND ,EAE(3,6)
      DATA  ISTG/6*0/
      END
```

FIGURE 110. SYMBOLIC LISTING OF SUBROUTINE BDATA

APPENDIX

SAMPLE INPUT DATA SHEETS

TENSOR INITIAL CONDITIONS, HEADERS, AND FLAGS

DATE ____________________

NAME ____________________

PROBLEM NO. ____________________

NO. OF CARDS ____________________

PAGE ______ OF ______

PRIORITY ____________________

KEYPUNCHED BY ____________________

VERIFIED BY ____________________

1	7	73
H13		
H14		
H15		

Note: Location field must be left justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
		TI			Initial time - sec
		VI			Initial velocity - ft/sec
		HI			Initial altitude - ft
		BETAI			Initial β - degrees
		AZI			Initial or launch azimuth - degrees
		BETAO			Kickover at end of first section vertical rise - degrees
		LATI			Initial or launch latitude - degrees
		LONGI			Initial or launch longitude - degrees
		WI			Initial weight - lb
		WPL			Payload weight - lb
		SIDET			Initial sidereal time if launch date not input
		FLTR			Flag for type of run
		FLSC			Number of sections
	B	FLPRT			Print option
year		DATE			Launch date for computing Julian Date and sidereal time
month		2			
day		3			
hour		4			
min		5			
sec		6			
zone		7			
	A	TARGT			Target body
		TRANS			Transfer time (days for planets, hours for moon)

TENSOR SECTION DATA INPUT

DATE ____________
NAME ____________
PROBLEM NO. ____________
NO. OF CARDS ____________

PAGE ____ OF ____
PRIORITY ____________
KEYPUNCHED BY ____________
VERIFIED BY ____________

1	7	73
* H		

Note: Location field must be left justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
* PARAM	A	SEX			Parameter to end section
VE		2			Value of parameter to end section
WIOWJ		3			Initial weight (+), or jettison weight (-) - lb
STIND		4			Stage indicator
THIND		5			Thrust indicator
ATIND		6			Attitude indicator
DTSBS		7			Integration interval - sec
DTSBP		8			Print interval - sec
PDOT		9			Pitch rate - deg/sec
YDOT		10			Yaw rate - deg/sec
PKIK		11			Pitch kick - deg
YKIK		12			Yaw kick - deg
VGX		13			X component of velocity gain for guidance system
VGY		14			Y component of velocity gain for guidance system
VGZ		15			Z component of velocity gain for guidance system
SECTNO		16			Section number
PADSK	A	17			Parameter for section skip
VESK		18			Value for section skip
PADRT1	A	19			Routine to execute before stage data moved
PADRT2	A	20			Routine to execute after stage data moved

* Fill in desired Section No. -
01 - 12 - for header and
location of section data.

TENSOR STAGE DATA INPUT

DATE ____________

NAME ____________

PROBLEM NO. ____________

NO. OF CARDS ____________

PAGE ____ OF ____

PRIORITY ____________

KEYPUNCHED BY ____________

VERIFIED BY ____________

1	7	73

Note: Location field must be left justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
* SREF		STAG			Aerodynamic reference area - ft^2
WIOWJS		2			Initial weight (+), or jettison weight (-) - lb
WOIS		3			Initial oxidizer weight - lb
WFIS		4			Initial fuel weight - lb
WPRPS		5			Monopropellant weight - lb
WBOS		6			Burn-out weight - lb
FVAC		7			Vacuum thrust - lb
WDOTV		8			Vacuum flow rate - lb/sec
EPSIS		9			Expansion ratio
ASUBTS		10			Throat area - in^2
ISP		11			Specific impulse - sec
KSUBF		12			Thrust multiplier
KSUBW		13			Flow rate multiplier
KSUBP		14			Pitch multiplier
KSUBY		15			Yaw multiplier
KSUBN		16			Normal force multiplier
KSUBA		17			Axial force multiplier

* Fill in desired Stage No. 1 - 6.

TENSOR MACH NUMBER INPUT DATA

DATE ______

NAME ______

PROBLEM NO. ______

NO. OF CARDS ______

PAGE ______ OF ______

PRIORITY ______

KEYPUNCHED BY ______

VERIFIED BY ______

1	7	73
* GMACH	②MACH①NUMBER①TABLE	

Note: Location field must be <u>left</u> justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.
		1		
		2		
		3		
		4		
		5		
		6		
		7		
		8		
		9		
		10		
		11		
		12		
		13		
		14		
		15		
		16		
		17		
		18		
		19		
		20		
		21		
		22		
		23		
		24		
		25		
		26		
		27		
		28		
		29		
		30		

* Be sure to fill in stage number.

TENSOR AERODYNAMIC AXIAL FORCE COEFFICIENT INPUT DATA

DATE ____________

NAME ____________

PROBLEM NO. ____________

NO. OF CARDS ____________

PAGE ______ OF ______

PRIORITY ____________

KEYPUNCHED BY ____________

VERIFIED BY ____________

1	7	73
* GCSBA	②AXIAL①FORCE①COEFFICIENT①AT①ZERO①ANGLE①OF①ATTACK	

Note: Location field must be left justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.
		1		
		2		
		3		
		4		
		5		
		6		
		7		
		8		
		9		
		10		
		11		
		12		
		13		
		14		
		15		
		16		
		17		
		18		
		19		
		20		
		21		
		22		
		23		
		24		
		25		
		26		
		27		
		28		
		29		
		30		

* Be sure to fill in stage No.

TENSOR AERODYNAMIC NORMAL FORCE COEFFICIENT INPUT DATA

DATE ____________

NAME ____________

PROBLEM NO. ____________

NO. OF CARDS ____________

PAGE ______ OF ______

PRIORITY ____________

KEYPUNCHED BY ____________

VERIFIED BY ____________

1	7	73
* GCSBN	②NORMAL①FORCE①COEFFICIENT②(CN/ALPHD)	

Note: Location field must be left justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.
		1		
		2		
		3		
		4		
		5		
		6		
		7		
		8		
		9		
		10		
		11		
		12		
		13		
		14		
		15		
		16		
		17		
		18		
		19		
		20		
		21		
		22		
		23		
		24		
		25		
		26		
		27		
		28		
		29		
		30		

* Be sure to fill in stage No.

TENSOR INFLUENCE COEFFICIENT DATA INPUT

DATE ______________
NAME ______________
PROBLEM NO. ______________
NO. OF CARDS ______________

PAGE ____ OF ____
PRIORITY ______________
KEYPUNCHED BY ______________
VERIFIED BY ______________

1	7	73

Note: Location field must be left justified.

	SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
*		G	YBBR	②SPECIFIC	①V	YBBRi i = stage No.
		A	LUES①	OF①DEPENDE	NT	
		①	ENGIN	E①PARAMETE	RS	
	$\bar{\bar{P}}_c$		1			Thrust chamber nozzle stagnation pressure, psia
	$\bar{\bar{W}}_{oc}$		2			Thrust chamber oxidizer flow rate, lb/sec
	$\bar{\bar{W}}_{fc}$		3			Thrust chamber fuel flow rate, lb/sec
	$\bar{\bar{W}}_{og}$		4			Gas generator oxidizer flow rate, lb/sec
	$\bar{\bar{W}}_{fg}$		5			Gas generator fuel flow rate, lb/sec
*		G	YBAR	②NOMINAL ①	①V	YBARi i = stage No.
		A	LUES①	OF ① DEPENDE	NT	
		①	ENGIN	E ① PARAMETE	RS	
	$\bar{P}_c$		1			Thrust chamber nozzle stagnation pressure, psia
	$\bar{W}_{oc}$		2			Thrust chamber oxidizer flow rate, lb/sec
	$\bar{W}_{fc}$		3			Thrust chamber fuel flow rate, lb/sec
	$\bar{W}_{og}$		4			Gas generator oxidizer flow rate, lb/sec
	$\bar{W}_{fg}$		5			Gas generator fuel flow rate, lb/sec
*		G	XNOM	② NOMINAL ①	①V	XNOMi i = stage No.
		A	LUES①	OF ① INDEPEN	DE	
		N	T①ENG	INE ① PARAME	TE	
		R	S			
	P_o		1			Oxidizer pump inlet pressure, psia
	P_f		2			Fuel pump inlet pressure, psia
	T_o		3			Oxidizer pump inlet temperature, °R
	T_f		4			Fuel pump inlet temperature, °R
	$\bar{F}$		5			F nom - for uprated thrust effect
	N_c		6			C-star correction factor
						* Be sure to fill in stage number desired.

TENSOR INFLUENCE COEFFICIENT DATA INPUT (cont.)

DATE ____________
NAME ____________
PROBLEM NO. ____________
NO. OF CARDS ____________

PAGE ______ OF ______
PRIORITY ____________
KEYPUNCHED BY ____________
VERIFIED BY ____________

1 | 7 | 73

Note: Location field must be left justified.

	SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
*		G	KPC	②PC①INFLU	EN	KPCi i = stage No.
		C	E①COE	FFICIENTS		
	P_o		1			
	P_f		2			
	T_o		3			
	T_f		4			
	$\bar{\bar{F}}$		5			
	N_c		6			
*		G	KWOC	②CHAMBER①	OX	KWOCi i = stage No.
		I	DIZER	①FLOW①RATE	①I	
		N	FLUEN	CE①COEFFIC	IE	
		N	TS			
	P_o		1			
	P_f		2			
	T_o		3			
	T_f		4			
	$\bar{\bar{F}}$		5			
	N_c		6			
*		G	KWFC	②CHAMBER①	FU	KWFCi i = stage No.
		E	L①FLO	W①RATE①INF	LU	
		E	NCE①C	OEFFICIENT	S	
	P_o		1			
	P_f		2			
	T_o		3			
	T_f		4			
	$\bar{\bar{F}}$		5			
	N_c		6			

* Be sure to fill in stage number desired.

TENSOR INFLUENCE COEFFICIENT DATA INPUT (cont.)

DATE ____________

NAME ____________

PROBLEM NO. ____________

NO. OF CARDS ____________

PAGE ____ OF ____

PRIORITY ____________

KEYPUNCHED BY ____________

VERIFIED BY ____________

1	7	73

Note: Location field must be <u>left</u> justified.

SYMBOL	PRE (1, 19, 37, 55)	LOC. (2, 20, 38, 56)	VALUE (7, 25, 43, 61)	EXP. (17, 35, 53, 71)	
*	G	KWOG	②GAS①GENE	RA	KWOGi i = stage No.
	T	OR①OX	IDIZER①FLO	W①	
	R	ATE①I	NFLUENCE①C	OE	
	F	FICIE	NTS		
P_o		1			
P_f		2			
T_o		3			
T_f		4			
$\bar{F}$		5			
N_c		6			
*	G	KWFG	②GAS①GENE	RA	KWFGi i = stage No.
	T	OR①FU	EL①FLOW①RA	TE	
	I	NFLUE	NCE①COEFFI	CI	
	E	NTS			
P_o		1			
P_f		2			
T_o		3			
T_f		4			
$\bar{F}$		5			
N_c		6			

* Be sure to fill in stage number desired.

TENSOR PROPELLANT TEMPERATURE POLYNOMIAL COEFFICIENTS

DATE ____________________
NAME ____________________
PROBLEM NO. ____________________
NO. OF CARDS ____________________

PAGE ______ OF ______
PRIORITY ____________________
KEYPUNCHED BY ____________________
VERIFIED BY ____________________

1	7	73

Note: Location field must be left justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
*	G	CTOX	②OXIDIZER	①T	CTOXi i = stage No.
	E	MPERA	TURE①POLYN	OM	
	I	AL①CO	EFFICIENTS		
$a_0 \cdot x^0$		1			
$a_1 \cdot x^1$		2			
$a_2 \cdot x^2$		3			
$a_3 \cdot x^3$		4			
$a_4 \cdot x^4$		5			
$a_5 \cdot x^5$		6			
*	G	CTFL	②FUEL①TEM	PE	CTFLi i = stage No.
	R	ATURE	①POLYNOMIA	L①	
	C	OEFFI	CIENTS		
$b_0 \cdot x^0$		1			
$b_1 \cdot x^1$		2			
$b_2 \cdot x^2$		3			
$b_3 \cdot x^3$		4			
$b_4 \cdot x^4$		5			
$b_5 \cdot x^5$		6			

* Be sure to fill in stage number.

TENSOR PROPELLANT TANK INPUT DATA

DATE ____________
NAME ____________
PROBLEM NO. ____________
NO. OF CARDS ____________

PAGE ____ OF ____
PRIORITY ____________
KEYPUNCHED BY ____________
VERIFIED BY ____________

1	7	73
* GTNKS	②PROPELLANT①TANK①INPUT①PARAMETERS	

Note: Location field must be left justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
PIND		1			Propellant indicator 1 = lox/RP1, 2 = lox/LH_2
TREF		2			Reference temperature - RP1
DREF		3			Reference SG - RP1
FPTPA		4			Tank pressure option, 0 = constant, ≠0 = stepped
PTOG		5			Constant oxidizer tank top pressure, psig
PTFG		6			Constant fuel tank top pressure, psig
TSO1		7			Stage time to step ox. tank pressure - sec
PTO1		8			Initial ox. tank pressure - psia
TSO2		9			Stage time to reach final ox. tank pressure - sec
PTO2		10			Final ox. tank pressure - psia
C1O		11			1st degree term oxidizer pressure polynomial coeff.
C2O		12			2nd degree " " " " "
TSF1		13			Stage time to step fuel tank pressure - sec
PTF1		14			Initial fuel tank pressure - psia
TSF2		15			Stage time to reach final fuel tank pressure - sec
PTF2		16			Final fuel tank pressure - psia
C1F		17			1st degree term fuel pressure polynomial coeff.
C2F		18			2nd degree " " " " "
LO		19			Length of cylindrical part of ox. tank - ft
RO		20			Radius of ox. tank - ft
UHO		21			Type of ox. tank upper header 1-4
LHO		22			Type of ox. tank lower header 1-4
LF		23			Length of cylindrical part of fuel tank - ft
RF		24			Radius of fuel tank - ft
UHF		25			Type of fuel tank upper header 1-4
LHF		26			Type of fuel tank lower header 1-4

* Fill in stage number.

TENSOR MISCELLANEOUS ENGINE INPUT DATA

DATE ____________

NAME ____________

PROBLEM NO. ____________

NO. OF CARDS ____________

PAGE ______ OF ______

PRIORITY ____________

KEYPUNCHED BY ____________

VERIFIED BY ____________

1	7	73
* GMISC	②MISCELLANEOUS①ENGINE①INPUT①PARAMETERS	

Note: Location field must be <u>left</u> justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
NOFE		1			Number of engines
FTURB		2			Turbine thrust - lb
FOPT		3			Optimum thrust - lb
KURF		4			Uprated thrust indicator
DTBUP		5			Δt for thrust buildup - sec
VOCOFF		6			Oxidizer shutdown volume - ft^3
VFCOFF		7			Fuel shutdown volume - ft^3
EPVOL		8			Convergence tolerance for shutdown volumes - ft^3
DTDCY		9			Δt for thrust decay - sec
WMSDT		10			Misc. stage lube, etc., flow rate - lb/sec
WMOSD		11			Misc. stage oxidizer flow rate - lb/sec
WMFSD		12			Misc. stage fuel flow rate - lb/sec
WMOED		13			Misc. engine oxidizer flow rate - bl/sec
WMFED		14			Misc. engine fuel flow rate - lb/sec
RESO		15			Oxidizer feed line hydraulic resistence - sec^2/ft^5
RESF		16			Fuel feed line hydraulic resistence - sec^2/ft^5
ASUBT		17			Thrust chamber nozzle throat area - in^2
EPSI		18			Thrust chamber nozzle expansion area ratio
ETACF		19			Nozzle efficiency
ETACS		20			Combustion efficiency
TOI		21			Initial oxidizer temperature - °R
CTO		22			Oxidizer temperature scale factor
TFI		23			Initial fuel temperature - °R
CTF		24			Fuel temperature scale factor
VOBB		25			Volume of oxidizer below bulkhead - ft^3
VFBB		26			Volume of fuel below bulkhead - ft^3
HOCOL		27			Height of oxidizer column - ft
HFCOL		28			Height of fuel column - ft
PARAML		29			Parameter for lift-off - (stage 1)
VPARL		30			Value of lift-off parameter
					* Fill in stage number 1-6

TENSOR PARAMETER STUDY AND GAINS ROUTINE INPUT

DATE

NAME

PROBLEM NO.

NO. OF CARDS

PAGE OF

PRIORITY

KEYPUNCHED BY

VERIFIED BY

1	7	73
H	PARAMETER①STUDY①INPUT	

Note: Location field must be left justified.

Note: FLTR must be negative to execute parameter study.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
	A	PSTDY			Parameter 1 — location 1
	A	2			Parameter 1 — location 2
	A	3			Parameter 1 — location 3
		4			Parameter 1 Delta
		5			Parameter 1 — number of increments
	A	6			Parameter 2 — location 1
	A	7			Parameter 2 — location 2
	A	8			Parameter 2 — location 3
		9			Parameter 2 Delta
		10			Parameter 2 — number of increments
	H		GAINS①ROUT	IN	
	E	①INPU	T		
		FLPAR			Non-zero to execute GAINS
	A	IVAR			Independent Variables
	A	2			
	A	3			
	A	4			
	A	5			
	A	6			
	A	7			
	A	8			
	A	9			
	A	10			
	A	DVAR			Dependent Variables
	A	2			
	A	3			
	A	4			
	A	5			
	A	6			
	A	7			
	A	8			
	A	9			
	A	10			

TENSOR 1 AND 2 PARAMETER SEARCH DATA INPUT

DATE ____________
NAME ____________
PROBLEM NO. ____________
NO. OF CARDS ____________

PAGE ______ OF ______
PRIORITY ____________
KEYPUNCHED BY ____________
VERIFIED BY ____________

1	7	73
H	ONE①PARAMETER①SEARCH①INPUT	

Note: Location field must be left justified.

*Be sure to fill in correct input block number (1 or 2).

SYMBOL	PRE 1 19 37 55	LOC. 2 20 38 56	VALUE 7 25 43 61	EXP. 17 35 53 71	
*	A	1PRS			Independent variable — location 1
	A	2			Independent variable — location 2
	A	3			Independent variable — location 3
	A	4			Dependent variable
		5			Desired value of dependent variable
		6			% Increment — independent variable
		7			Convergence tolerance
		8			Slope of dependent variable at solution
		9			Maximum number of iterations
		10			Maximum number of consecutive errors
		11			Maximum number of imaginary solutions
		12			Error scale factor
	H		TWO①PARAME	TE	
	R	①SEAR	CH①INPUT		
	A	2PRS1			Independent variable 1 — location 1
	A	2			Independent variable 1 — location 2
	A	3			Independent variable 1 — location 3
	A	4			Independent variable 2
	A	5			Dependent variable 1
	A	6			Dependent variable 2
		7			Desired value of dependent variable 1
		8			Desired value of dependent variable 2
		9			% Increment — independent variable 1
		10			% Increment — independent variable 2
		11			Convergence tolerance — variable 1
		12			Convergence tolerance — variable 2
		13			Jacobian counter
		14			Scale factor for point 0
		15			Scale factor for points 1 & 2
		16			Error stop for point 0
		17			Error stop for point 1
		18			Error stop for point 2
		19			Damping scale factor
		20			Number of times to damp
		21			Maximum number of iterations

TENSOR MIN-MAX SEARCH INPUT

PAGE ____ OF ____

DATE ____

NAME ____

PROBLEM NO. ____

NO. OF CARDS ____

PRIORITY ____

KEYPUNCHED BY ____

VERIFIED BY ____

1	7	73
H	MIN-MAX①SEARCH①INPUT	

Note: Location field must be left justified.

*Be sure to fill in correct input block number (1 or 2)

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
*	A	MNMX			Independent variable — location 1
	A	2			Independent variable — location 2
	A	3			Independent variable — location 3
	A	4			Dependent variable
		5			Min-max indicator, -1.0 = min, +1.0 = max
		6			% Increment
		7			Convergence tolerance
		8			Maximum number of iterations
		9			Maximum number of errors
		10			Error scale factor

TENSOR PROPELLANT LOADING OPTIMIZATION INPUT DATA

DATE ______

NAME ______

PROBLEM NO. ______

NO. OF CARDS ______

PAGE ____ OF ____

PRIORITY ______

KEYPUNCHED BY ______

VERIFIED BY ______

1	7	73
GPROPD	②PROPELLANT①LOADING①OPTIMIZATION①DATA	

Note: Location field must be left justified.

	SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
			PROPD			Maximum number of iterations
			2			Maximum number of consecutive errors
			3			Option indicator
			4			Number of stages for weight buildup
Stage 1			5			Stage optimization flag
			6			Stage structure weight
			7			Oxidizer ullage volume
			8			Fuel ullage volume
Stage 2			9			Stage optimization flag
			10			Stage structure weight
			11			Oxidizer ullage volume
			12			Fuel ullage volume
Stage 3			13			Stage optimization flag
			14			Stage structure weight
			15			Oxidizer ullage volume
			16			Fuel ullage volume
Stage 4			17			Stage optimization flag
			18			Stage structure weight
			19			Oxidizer ullage volume
			20			Fuel ullage volume
Stage 5			21			Stage optimization flag
			22			Stage structure weight
			23			Oxidizer ullage volume
			24			Fuel ullage volume
Stage 6			25			Stage optimization flag
			26			Stage structure weight
			27			Oxidizer ullage volume
			28			Fuel ullage volume

TENSOR MISCELLANEOUS INPUT DATA

DATE ____________

NAME ____________

PROBLEM NO. ____________

NO. OF CARDS ____________

PAGE ______ OF ______

PRIORITY ____________

KEYPUNCHED BY ____________

VERIFIED BY ____________

1	7	73

Note: Location field must be left justified.

SYMBOL	1 19 37 55 PRE	2 20 38 56 LOC.	7 25 43 61 VALUE	17 35 53 71 EXP.	
	E	ND			End of case card

REFERENCES

[1] S. Gill, A Process for the Step-by-Step Integration of Differential Equations in an Automatic Digital Computing Machine, published in Cambridge Philosophical Society Proceedings, Vol. 47, January 1951.

[2] R. S. Burington, Handbook of Mathematical Tables and Formulas, Handbook Publishers, Inc., 1954.

[3] Richard H. Battin, Astronautical Guidance, McGraw-Hill Book Company, 1964.