NASA TECHNICAL MEMORANDUM

A REAL-TIME DIGITAL COMPUTER PROGRAM FOR THE SIMULATION OF AUTOMATIC SPACECRAFT REENTRIES

Jack T. Kaylor, Lawrence F. Rowell, and Richard W. Powell

Langley Research Center
Hampton, Va. 23665

1. Report No. NASA TM X-3496	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle A REAL-TIME DIGITAL COMPUTER PROGRAM FOR THE SIMULATION OF AUTOMATIC SPACECRAFT REENTRIES		5. Report Date July 1977
		6. Performing Organization Code
7. Author(s) Jack T. Kaylor, Lawrence F. Rowell, and Richard W. Powell		8. Performing Organization Report No. L-11260
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665		10. Work Unit No. 506-26-30-04
		11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546		13. Type of Report and Period Covered Technical Memorandum
		14. Sponsoring Agency Code

15. Supplementary Notes
    Appendix D: Convolution Integration Technique by Lawrence E. Barker and Lawrence F. Rowell.

16. Abstract

The automatic reentry flight dynamics simulator (ARFDS), a nonlinear, six-degree-of-freedom simulation, digital computer program, has been developed at the NASA Langley Research Center (LaRC). The program includes a rotating, oblate Earth model for accurate navigation calculations and contains adjustable gains on the aerodynamic stability and control parameters. This program uses the Langley real-time simulation system (RTSS) available at LaRC and is designed to examine entries of vehicles which have constant mass properties (no main engine thrust), whose attitudes are controlled by both aerodynamic surfaces and reaction control thrusters, and which have automatic guidance and control systems (no pilot in the loop). The program has been used to study the space shuttle orbiter entry.

This report includes descriptions of the equations of motion used, the control and guidance schemes that were implemented, the program flow and operation, and the RTSS hardware involved.

17. Key Words (Suggested by Author(s)) Digital simulation of a reentry vehicle Automatic guidance and control system Real-time six-degree-of-freedom simulation	18. Distribution Statement Unclassified - Unlimited Subject Category 16
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	131	$6.00

CONTENTS

A REAL-TIME DIGITAL COMPUTER PROGRAM FOR THE SIMULATION

OF AUTOMATIC SPACECRAFT REENTRIES

Jack T. Kaylor, Lawrence F. Rowell,
and Richard W. Powell
Langley Research Center

SUMMARY

The automatic reentry flight dynamics simulator (ARFDS), a nonlinear, six-degree-of-freedom simulation, digital computer program, has been developed at the NASA Langley Research Center (LaRC). The program includes a rotating, oblate Earth model for accurate navigation calculations and contains adjustable gains on the aerodynamic stability and control parameters. This program uses the Langley real-time simulation system (RTSS) available at LaRC and is designed to examine entries of vehicles which have constant mass properties (no main engine thrust), whose attitudes are controlled by both aerodynamic surfaces and reaction control thrusters, and which have automatic guidance and control systems (no pilot in the loop). The program has been used to study the space shuttle orbiter entry.

This report includes descriptions of the equations of motion used, the control and guidance schemes that were implemented, the program flow and operation, and the RTSS hardware involved.

INTRODUCTION

The automatic reentry flight dynamics simulator (ARFDS), developed by the NASA Langley Research Center (LaRC), is a nonlinear, six-degree-of-freedom simulation of a vehicle which has constant mass properties (no main engine thrust) and whose attitudes are controlled by both aerodynamic surfaces and reaction control system thrusters. A rotating, oblate Earth was modeled to obtain more accurate gravitational calculations for long-duration navigation studies. The ARFDS is being used to evaluate the automatic entry guidance and control systems of the space shuttle orbiter from 121.9 km (400 000 ft) down to the initiation of the terminal area phase which begins at 21.3 km (70 000 ft). Of primary interest are studies of the effects that off-nominal aerodynamics have on the performance of the guidance and control system.

To facilitate a variety of studies, the program contains adjustable gains on the aerodynamic stability and control parameters and allows flight initialization at various points of the entry. To simulate the manner in which the automatic systems are used by the shuttle onboard computers, the sampling intervals for the guidance system and the control system can be chosen independently.

The ARFDS does not allow for a pilot in the loop since it is designed to simulate automatic flight systems, but a derivative of the ARFDS, the orbiter

flight dynamics simulator (OFDS), has been designed for use at LaRC with a simulator cockpit and pilot. Only the ARFDS will be discussed in this report.

The objectives of this report are to describe the simulation equations and special features in the ARFDS and to discuss briefly the Langley real-time simulation system (RTSS) facility and program operating procedures.

The entry guidance system is described in appendix A. Appendix B describes the entry digital autopilot. The equations of motion are presented in appendix C. A convolution integration technique developed by Lawrence E. Barker and Lawrence F. Rowell of Langley Research Center is presented in appendix D. The ARFDS program listing is given in appendix E.

SYMBOLS

Values are given in both SI and U.S. Customary Units. The measurements were made in U.S. Customary Units. Symbols used only in the appendixes are defined therein.

h	altitude, m (ft)
M	Mach number
p,q,r	inertial roll, pitch, and yaw rates about body axes, rad/sec
RCS	reaction control system
V_R	Earth relative velocity, m/sec (ft/sec)
V_X,V_Y,V_Z	velocities along inertial axes, m/sec (ft/sec)
x,y,z	components of position along inertial axes, m (ft)
α	angle of attack, deg
α_c	commanded angle of attack from guidance system, deg
β	angle of sideslip, deg
γ	flight-path angle, deg
δ_{BF}	body flap deflection, deg
δ_e	total elevon deflection, deg
δ_{el}	elevon deflection, left panel, deg
δ_{er}	elevon deflection, right panel, deg
δ_{SB}	speed-brake deflection, deg

θ_c longitude, deg

λ geocentric latitude, deg

ϕ,θ,ψ roll, pitch, and yaw angle about body axes, deg

ϕ_c commanded bank angle from guidance system, deg

ψ_e heading angle, deg

PROBLEM DESCRIPTION

The ARFDS was developed to examine the entry of the space shuttle orbiter from the atmospheric interface at 121.9 km (400 000 ft) with a velocity of 8118.04 m/sec (26 634 fps) to the beginning of the terminal area phase at 21.3 km (70 000 ft) with a velocity of 454.15 m/sec (1490 fps). Below 21.3 km the shuttle switches to different guidance and control systems, and these were not modeled. The physical characteristics of the space shuttle orbiter used in this program are described in table I. The aerodynamic data were obtained from the

TABLE I.- PHYSICAL CHARACTERISTICS OF SPACE SHUTTLE ORBITER

```
Mass properties:
    Mass, kg (lb) . . . . . . . . . . . . . . . . . . . . . . .   83 001    (182 986)
    Moments of inertia:
        I_XX, kg-m² (slug-ft²) . . . . . . . . . . . . . . . . 1 029 066    (759 000)
        I_YY, kg-m² (slug-ft²) . . . . . . . . . . . . . . . . 7 816 290  (5 765 000)
        I_ZZ, kg-m² (slug-ft²) . . . . . . . . . . . . . . . . 8 015 596  (5 912 000)
        I_XZ, kg-m² (slug-ft²) . . . . . . . . . . . . . . . .   177 612    (131 000)
        I_XY = I_YZ . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 0

Wing:
    Reference area, m² (ft²) . . . . . . . . . . . . . . . .   249.91      (2690.0)
    Chord, m (ft) . . . . . . . . . . . . . . . . . . . . . . .   12.06       (39.57)
    Span, m (ft) . . . . . . . . . . . . . . . . . . . . . . . .   23.79       (78.06)

Elevon:
    Reference area, m² (ft²) . . . . . . . . . . . . . . . . .  19.51       (210.0)
    Chord, m (ft) . . . . . . . . . . . . . . . . . . . . . . . .   2.30        (7.56)

Rudder:
    Reference area, m² (ft²) . . . . . . . . . . . . . . . . .   9.30      (100.15)
    Chord, m (ft) . . . . . . . . . . . . . . . . . . . . . . . .   1.86        (6.1)

Body flap:
    Reference area, m² (ft²) . . . . . . . . . . . . . . . . .  12.54       (135.0)
    Chord, m (ft) . . . . . . . . . . . . . . . . . . . . . . . .   2.06        (6.75)
```

June 1974 space shuttle orbiter aerodynamic data base. Throughout the entry, the orbiter is directed by onboard computers which process, in the automatic mode, both the guidance system (appendix A) and the control system (appendix B). Further information on the space shuttle systems and mission is given in reference 1.

The simulation equations of motion described in appendix C include the gravitational model, a rotating oblate Earth model, provisions for a wind model, and the viscous effects on the aeodynamic parameters. The aerodynamic interference with the reaction control system (RCS) thrusters is described in appendix B. The atmospheric model used in this program was the 1962 standard atmosphere.

The integration of the linear differential equations encountered in the control system is performed by a one-pass, semianalytic scheme known as the convolution method, and the difference equations resulting from this approach are discussed in appendix D. The function-look-up and interpolation algorithms were designed for simulations using the RTSS to reduce computer memory and execution time and are described in appendix B of reference 2.

PROGRAM DESCRIPTION

The ARFDS program was written to be operated from the Langley real-time simulation system (RTSS) facility, and, thus, required the inclusion of related supervisory software. The simulation presented in this paper was performed with the ICOPS operating system on CDC 6600 computers. The following discussion will briefly describe both the software and hardware components of the RTSS. In the fall of 1976 the Langley Research Center digital complex was upgraded with the acquisition of CDC CYBER 175 computers, a new operating system NOS (network operating system), and an expansion in the number of digital-to-analog converters (DAC's). The simulation has been updated to operate under the NOS.

RTSS Hardware

The RTSS links a Control Data Corporation (CDC) 6600 computer to a simulation control station through the use of two peripheral processors. The control station, shown in figure 1, is used to direct the digital simulation program being processed on the CDC 6600 computer and to monitor the data being recorded on the various devices.

The devices shown from the right to left in figure 1 are an on-line typewriter for message printout, the control console with digital decimal display, an X-Y plotter, a strip chart time-history recorder, and a cathode-ray tube for both real-time plotting and text editing. Not shown is a high-speed printer for data recording.

The RTSS can be tied to any of several remote sites (fig. 2) where a pilot is brought into the loop through the use of simulation hardware such as cockpits, scene generators, and motion bases. Three elements, the computer, the control station, and the remote sites, must be capable of exchanging signals. The analog signals from either the remote sites or the control station that are to be used by the computer must be converted to digital bit patterns. Likewise, the digital values sent from the computer to either the remote sites or the control station must be converted to analog signals. Therefore, analog-to-digital conversion (ADC) and digital-to-analog conversion (DAC) must be performed whenever computer input and output (I/O) are requested. The conversions of both the continuous and discrete analog signals to computer words are performed by the

4

L-69-8762
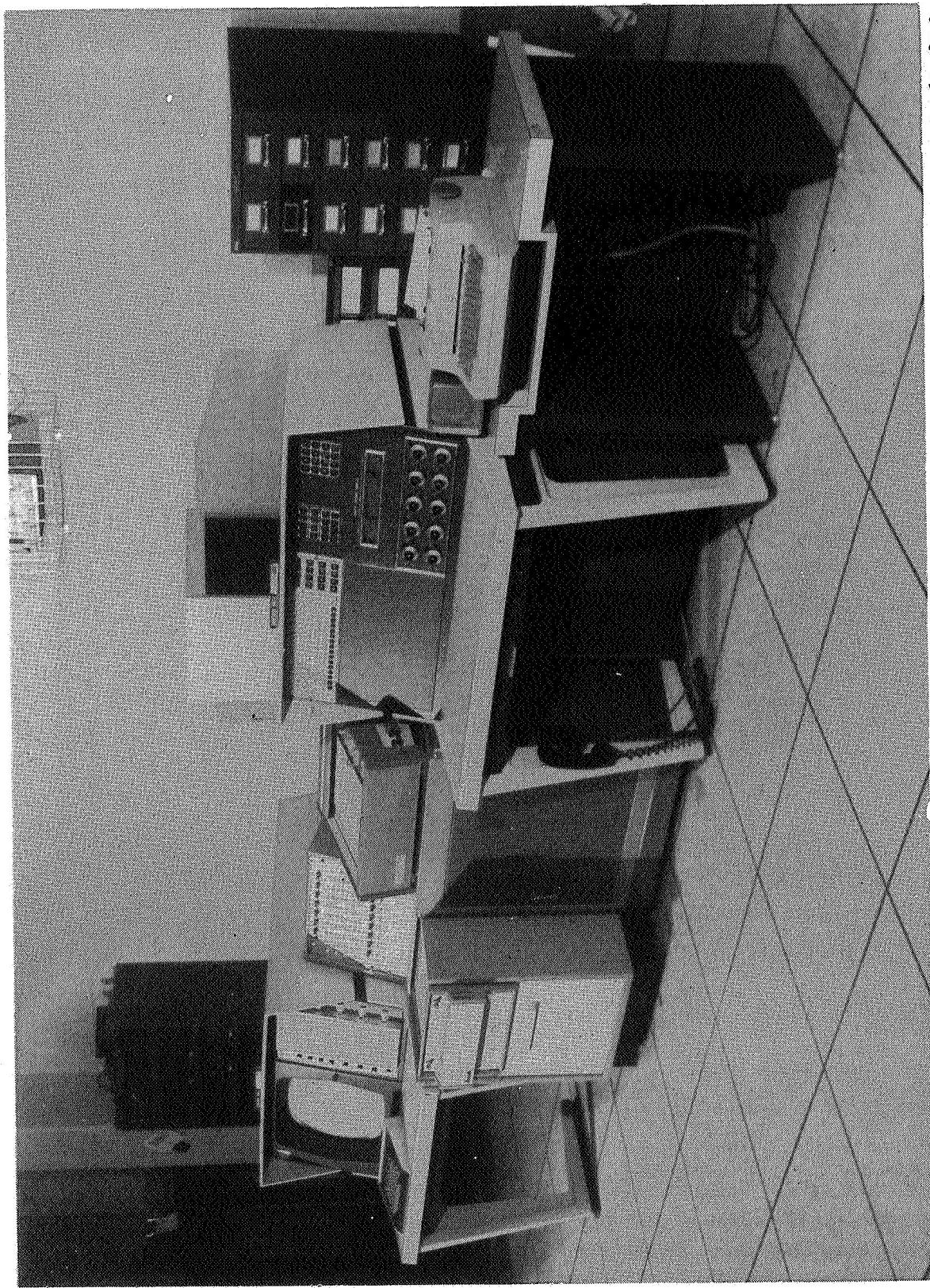
Figure 1.- Typical program control station.

5

Figure 2.- Langley real-time simulation system (RTSS) hardware, computer, and remote site interface.

6
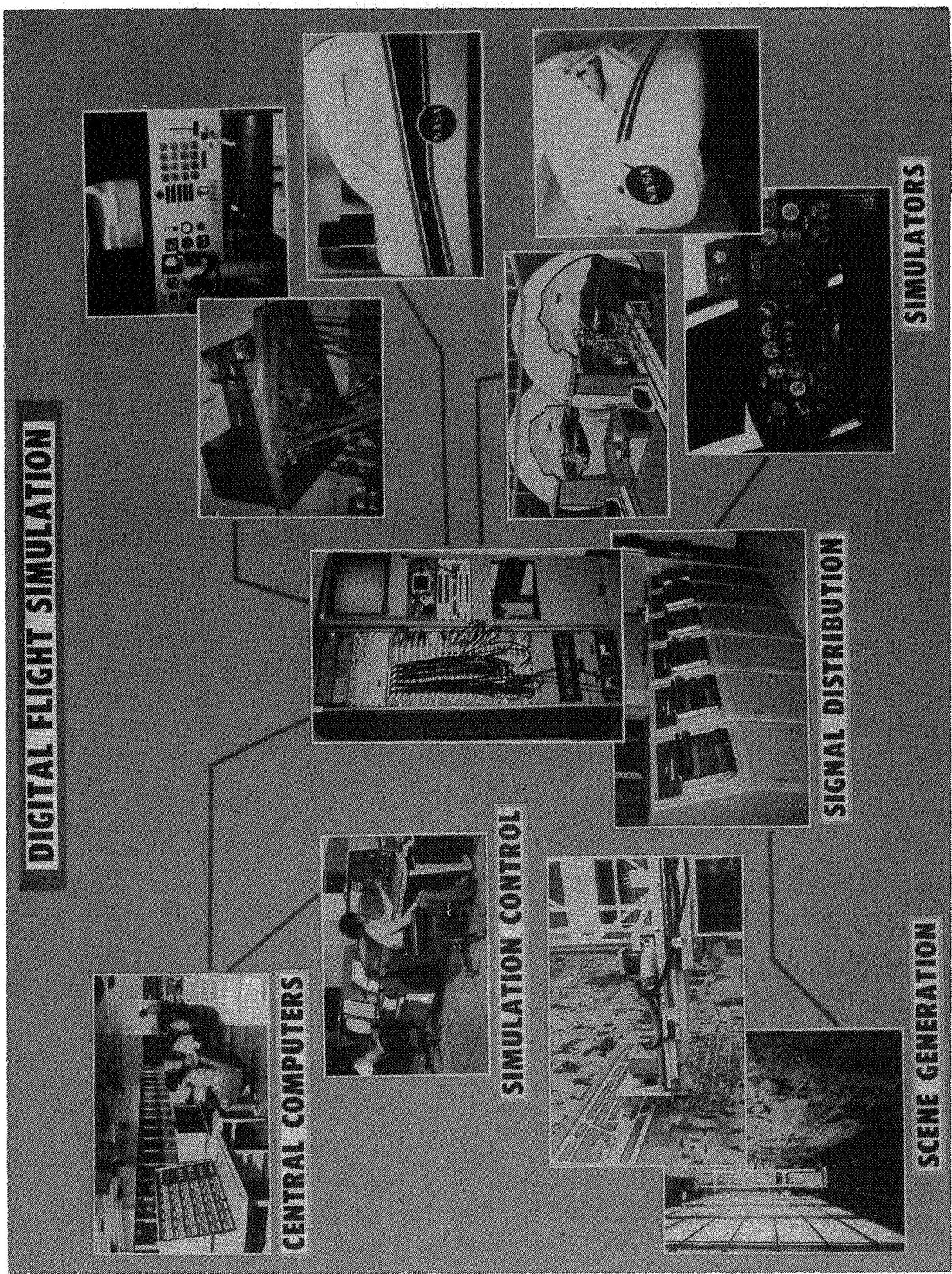
analog-to-digital and discrete-input system (ADDIS), whereas the conversions from computer words to analog signals are performed by the digital-to-analog and discrete-output system (DADOS).

The I/O is performed when called for periodically by the real-time clock. This transfer of information must be performed at a rate high enough to insure that the signals used to drive scene generators, cockpit instruments, and other simulation hardware appear to be continuous. Experience has shown in most cases that if I/O is performed at a rate of 16 times a second or greater, no discontinuities will be observed by a pilot performing simulation flight tasks. Programs, then, will normally use an integration step size of between 1/16 and 1/32 of a second, and the new information produced with each iteration can be synchronized with real time by using the clock to call for I/O at the appropriate time after the completion of each integration step.

RTSS Software

The simulation program contains two arrays with the FORTRAN names ADC and DAC. The digital values computed by the program are stored in the DAC array to await conversion by DADOS, whereas analog signals, such as controller deflections from remote sites, have their values converted by ADDIS and placed in the ADC array for use within the program. These two arrays are used by the supervisory software as the storage locations for values transferred between computers and hardware. In a similar manner, logical program variables that must be sent to hardware (logical discretes out) for purposes of activating lights or switches are stored in the FORTRAN array designated LDISO. Hardware items, such as push buttons or switches, needed to select program options (logical discretes in) have their signals converted to digital true-or-false bit patterns and stored in the array LDISI.

Control Console

Figure 3 shows a closeup of the control console. There are three blocks of switches (discretes into the program, LDISI) with 16 push buttons each and a large group of numbered white and red lights (discretes out of the program, LDISO). Switches in the block on the far left are used as logical variables to select options programed into the simulation. The numbered lights are turned on by program logical variables and are used to mark the occurrence of selected events.

The group of switches on the right is used to address locations in the computer memory and to either change or display their values on the digital decimal display unit located immediately below the switches. The center group of switches is used to control the mode of program operation. The mode controls were designed to be similar to the analog computer, that is, initial condition or reset, hold, and operate. This group also contains special purpose buttons to print or erase data, to read data, and to take the program out of real-time operation. The 10 potentiometers (brought into the program through the ADC's) can be used as additional inputs to the program.

Figure 3.- Closeup of control panel on program control console.

L-68-10517

PROGRAM CONTROL STATION

8
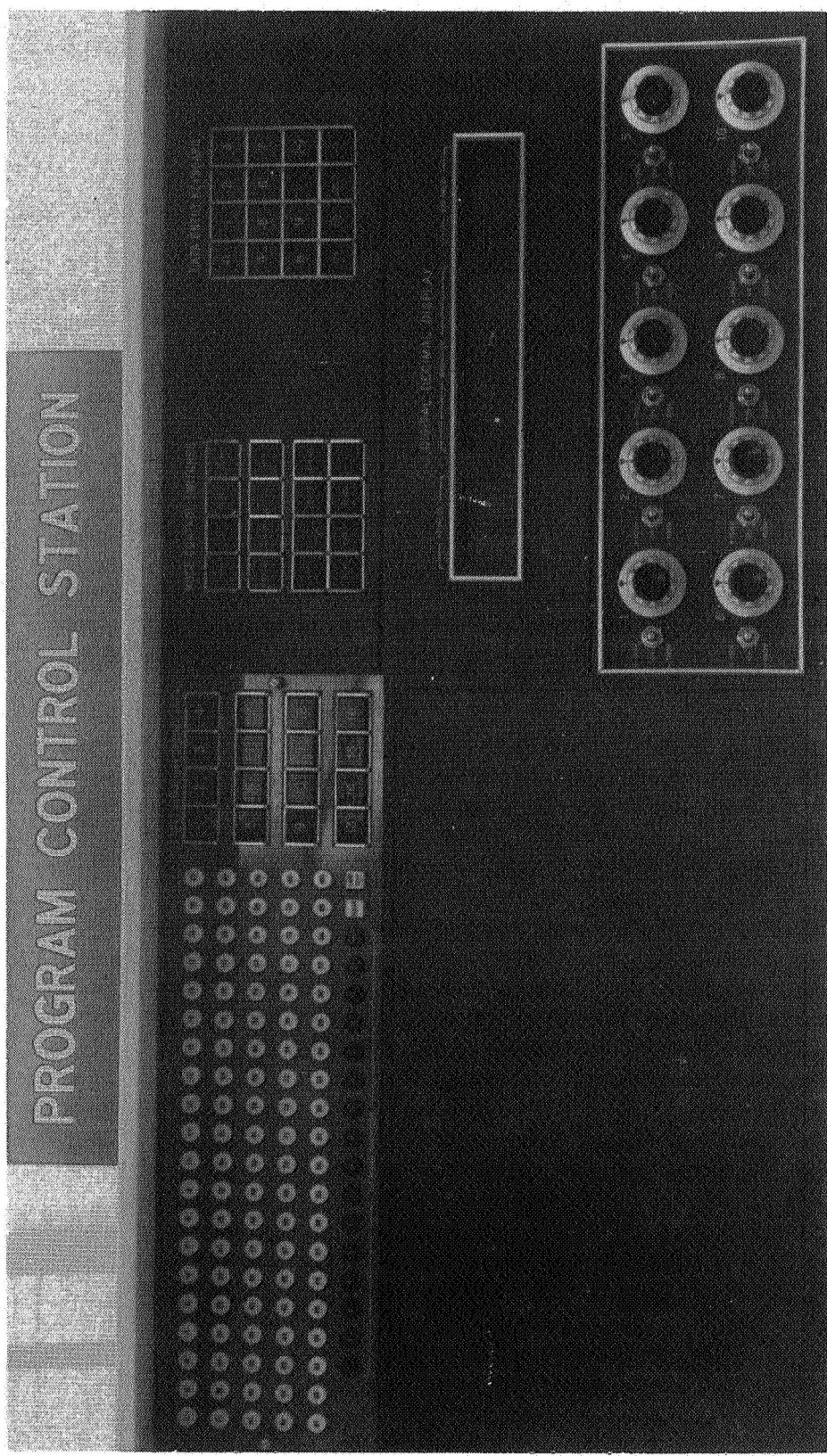
Program Flow

The FORTRAN program code must be divided functionally to utilize the mode switches just described. Figure 4 is a simplified flow diagram indicating the functions performed by each of the three principal modes (reset, hold, and operate) that can be selected at the control console. The following discussion of figure 4 defines each of the major calls to RTSS software shown in the figure, as well as the purpose of each mode.

The first block containing "constants and initialization" represents any code that must be performed only once when the program is first submitted to the computer. The call to READY begins clock synchronization of both the iterative computations and the input and output through ADDIS and DADOS. From this point, all calculations must be performed fast enough to complete one integration step before the next clock pulse calls for I/O.

Selecting the RESET button at the control console will place the flow in a loop beginning with the block containing "initial conditions" and extending to the call to RTMODE. RTMODE checks the switch setting of the control console and directs the flow according to the mode switch that is depressed. In RESET, the value of INT is zero. A HOLD mode selected by pressing the HOLD button is available which will loop from the block of "hold equations" to the call to RTMODE. The HOLD mode, although not used in this simulation, normally consists of any special equations, such as trim calculation, that are to be performed before the simulation proceeds with the integration process.

The OPERATE mode extends from the block containing "ADC's and LDISI's" to the call to RECYCLE. This mode performs the integration as well as data storage. The call to DSPLAY is made once per iteration during all three modes to update the digital decimal display on the control console. The subroutine RECORD is called only once per iteration in the operate mode to save for printing the current values of any variables previously specified by calls made to system subroutines in the initialization section.

The subroutine IGRATE1 is a standard subroutine which performs the numerical integration of all nonlinear differential equations by one of the following methods:

(1) Runge-Kutta, second order

(2) Runge-Kutta, fourth order

(3) Adams-Moulton, second order

(4) Adams-Moulton, fourth order

For these multipass schemes, the flag INT is needed to control the looping and to insure that data are recorded and I/O performed only when an integration step is completed (indicated by INT = 1). Predictor-corrector methods (3) and (4) can be reduced to single-pass, predictor-only methods when appropriate. ARFDS uses the second order Runge-Kutta integration scheme for the numerical integration of the equations of motion.
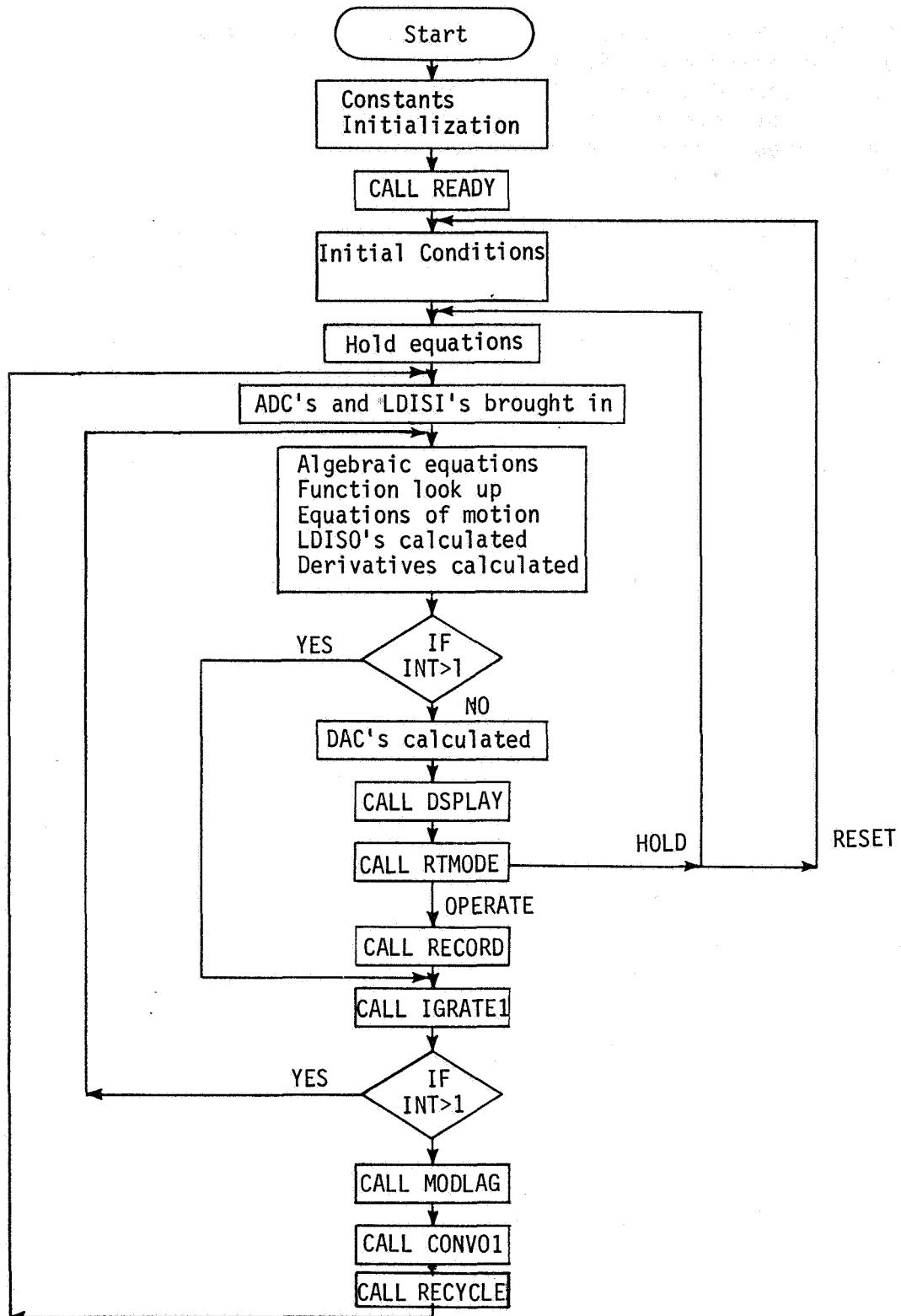
Figure 4.- Block diagram of real-time program modes.

The calls to CONVO1 and MODLAG represent special integration routines which have been designed for the solution of linear differential equations and are used for the solution of the linear control system equations of ARFDS. Both of these routines use a convolution integration technique (appendix D) which requires one pass and can be used with the multipass methods in IGRATE1. The MODLAG routine includes provisions for rate and position limiting.

RECYCLE is called when one iteration (all passes of an integration step) of the problem is completed. At this point the program will cease calculations until a clock pulse signals for I/O and the latest inputs are available for the next iteration in OPERATE.

There are other calls to RTSS software used primarily in the initialization of the program, but they will not be discussed here. Figure 5 is a more detailed diagram of the ARFDS program indicating where the primary components of the simulation are computed. Copies of the main program, its variable cross-reference map, and source listings of the major mathematical subroutines are supplied in appendix E. Most of the routines are in FORTRAN, but the COMPASS language routine was also included for completeness. Figure 6 is a block diagram that describes the flow of variables through the guidance system, control system, and equations of motion for the ARFDS program.

```
                    ┌──────────┐
                    │  ARFDS   │
                    └──────────┘
                         │
                         ▼
        ┌─────────────────────────────┐
        │ COMMON statements           │
        │ REAL   statements           │
        │ DIMENSION statements        │
        │ LOGICAL statements          │
        │ EQUIVALENCE statements      │
        └─────────────────────────────┘
                         │
                         ▼
      ┌──────────────────────────────────────────────┐
      │ CALL DATAIC - read and pack aerodynamic       │
      │ function data. DATA statements                │
      └──────────────────────────────────────────────┘
                         │
                         ▼
       ┌─────────────────────────────────────┐
       │ Calls to set up typewriter, DAC and │
       │ discrete channels, display addresses,│
       │ and statement numbers for mode      │
       │ controls.  Specify the variables to │
       │ be recorded.                        │
       └─────────────────────────────────────┘
                         │
  CONSTANTS SECTION      ▼
       ┌─────────────────────────────────────┐
       │ Constants and initial calculations  │
       │ that need to be done only once.     │
       └─────────────────────────────────────┘
                         │
                         ▼
      ┌──────────────────────────────────────────────┐
      │ CALL READY - from this point, calculations    │
      │ and I/O will be synchronized with a real-time │
      │ clock.                                         │
      └──────────────────────────────────────────────┘
        (A)──────────────────────▶│
  RESET SECTION                    ▼
       ┌─────────────────────────────────────┐
       │ Transformations to get initial      │
       │ conditions for variables of         │
       │ integration.  Set variables to      │
       │ their values at  Time = 0.          │
       └─────────────────────────────────────┘
        (B)──────────────────────▶│
  HOLD SECTION                     ▼
      ┌──────────────────────────────────────────────┐
      │ Any special calculations to be performed when │
      │ the simulation is put into hold.  None in     │
      │ ARFDS.                                        │
      └──────────────────────────────────────────────┘
                         │
                         ▼
                        (C)
```

Figure 5.- Block diagram of simulation program functions.

C

OPERATE SECTION

Calculations for latitude, longitude, direction cosines, and body velocities.

CALL ATMOS3 - calculate atmospheric quantities.

CALL DATAFUN(1) - do table look-up of aerodynamic functions of M and α.

Calculate viscous interaction equations.

GUIDANCE SCHEME

Calculated ϕ_c and α_c every GUIDDT seconds.

CONTROL SYSTEM

Calculate aerodynamic interference, control deflections, aerodynamic functions of control deflections (CALL DATAFUN(2)), and reaction control system firings.

Calculate forces, moments, and derivatives.

Have all passes of a multipass integration scheme been completed (INT$\leq$1)?

NO → D

YES

Calculate values for DAC's to output to strip chart recorders. Output latest values to the digital decimal display unit.

RESET Ⓐ
HOLD Ⓑ

CALL RTMODE - route program flow according to mode switches set at the control console.

TERMINATE

OPERATE
Ⓔ

Figure 5.- Continued.

13

Figure 5.- Concluded.

Entry Digital Autopilot

Entry Guidance System

```
┌─────────────────────┐                    ┌────────────────────────────────────┐
│ A.  Reaction        │                    │ A.  Constant Attitude Phase        │
│     Control         │                    │ B.  Constant-Heat-Rate Phase       │
│     System (RCS)    │   Φ_C , α_C        │ C.  Equilibrium Glide Phase        │
│                     │ ◄────────────────  │ D.  Constant Drag Phase            │
│ B.  Aerodynamic     │                    │ E.  Transition Phase               │
│     Control         │ ◄────────────────  │                                    │
│     System          │                    │                                    │
└─────────────────────┘                    └────────────────────────────────────┘
```

RCS Firings
and Control
Surface
Deflections

State Variables

Equations of Motion

```
┌──────────────────────────┐
│ A.  Force                │
│ B.  Moment               │
│ C.  Euler Angle Rates    │
│ D.  Direction Cosines    │
│ E.  Gravitational        │
│ F.  Auxiliary            │
└──────────────────────────┘
```

Initialization

Figure 6.- Block diagram showing variable flow between guidance and control
systems and equations of motion.

When a simulation program is first submitted to the computer for processing, the user should have the RESET switch selected at the control console. Thus, when the program begins execution, it will perform all initialization tasks and then loop in the RESET mode until some change is made at the console. Some of the changes that might be made include altering the value of a program parameter, selecting program options using the logical switches, reading data, or going to the hold or operate modes.

The ARFDS requires initialization of the variables p, q, r, α, β, γ, ϕ, V_R, ψ_e, θ_c, λ, and h. These variables are used in transformations to get the initial conditions for the state variables of integration: p, q, r, ϕ, θ, ψ, V_X, V_Y, V_Z, x, y, and z. The velocities are integrated in the inertial frame. Body-axis system velocities are obtained by transformation from the inertial axis.

For the purpose of changing values, any computer location in the program can be addressed from the console. However, the variables most frequently altered have been set up in three FORTRAN arrays called TABLE for floating point numbers, INTEG for integer numbers, and LOGIC for logical variables. These arrays are the most easily accessed from the console. In the ARFDS, TABLE contains, by a FORTRAN EQUIVALENCE, the following: the orbiter physical properties, the center-of-gravity offsets, the data recording rate, initial conditions, the variables reserved for future use, the initial control deflections, the constant wind components, the aerodynamic-stability and control parameter gains, and the guidance system sampling interval. The variable gains on each of the aerodynamic stability and control parameters are to be used in sensitivity studies.

The INTEG array contains, by a FORTRAN EQUIVALENCE, integers such as run number, the number of integration steps per second, the integration scheme selected, and the initial guidance phase. The first two variables in the LOGIC array, when set true, bypass the viscous interaction calculations and replace the oblate Earth model with a spherical Earth model.

The 16 switches in the block used for mode control are the logical discretes LDISI(17) to LDISI(32) which are numbered from left to right by rows starting with the top row. The function of each of these switches has previously been established and is incorporated in supervisory software.

Function Switches

The 16 function switches, logical discretes LDISI(33) to LDISI(48), have been assigned uses in ARFDS as follows:

Switch	Function
1	False – Calculate guidance equations each pass of the integration step.

Switch	Function
	True – Calculate guidance equations only at intervals of GUIDDT seconds.
2	False – Variables to be integrated by IGRATE1 will use the normal scheme as selected by the integer ISCHEME.
	True – If the scheme is a predictor-corrector method, use only the predictor pass.
3	False – Perform calculations in synchronization with the real-time clock.
	True – Process one iteration each time function switch 8 is pressed. Function switch 12 will cause the value in the digital decimal display to be typed out.
5	False – Evaluate aerodynamic functions for each pass of a multipass integration step.
	True – Evaluate aerodynamic functions during the first pass only of a multipass method.
8, 12	Used in conjunction with function switch 3.
15	False – No effect.
	True – Type out the address and value of the variable in the digital decimal display.
16	False – Variables brought into the digital decimal display must be addressed by their specific locations in computer memory.
	True – Variables in the arrays TABLE, INTEG, LOGIC, ADC, DAC, LDISI, and LDISO are brought into the digital decimal display in the addresses from 1 to 900.
4, 6, 7, 9, 10, 11, 13, 14	Not used.

The function switches are the inputs to the simulation program that can be made from the control console. This program does not use any of the 10 potentiometers on the console or any remote hardware site.

The principal outputs from the program to the control station are 16 time-history recorder channels which are driven by DAC(1) to DAC(16). In addition, the first five white lights, LDISO(61) to LDISO(65), are used to indicate the current guidance phase. The first three red lights, LDISO(111) to LDISO(113), are used to indicate the intersection with the terminal area phase of the

17

reentry. The additional white lights used, LDISO(81) to LDISO(88), indicate
RCS thruster firings, and the discrete event markers, LDISO(103) to LDISO(108),
located between time-history channels record the same information. The remain-
ing event markers, LDISO(109) and LDISO(46) to LDISO(48), record the guidance
(appendix A) phase changes on the time-history recorders.

Labeled COMMON

The labeled COMMON arrays used to communicate variables between the main
applications program and the subroutines are as follows:

REALTIM   Contains the arrays ACD, DAC, LDISI, and LDISO and the addresses
which control the looping for the various modes of operation.

INTCOMM   Carries information between the main program and the integration
subroutine, IGRATE1, which is used to solve the nonlinear differ-
ential equations. The variables are current time, step size,
multipass control flag, number of equations (NEQ) to be inte-
grated, integration scheme selected, and a 2 × NEQ array contain-
ing the integrals and derivatives.

INTINTR   Contains a 5 × NEQ buffer used by IGRATE1.

CCONA1   Carries information between the main program and the integration
subroutine, CONVO1, which is used to solve the linear differen-
tial equations of the control system. The variables are step
size, number of equations (NEQ1) to be integrated, and a
10 × NEQ1 array containing the inputs into and the solution
states from the transfer function.

TABLES   Carries the variables expressed as equivalents to TABLE and
INTEG into the subroutines that need these values.

DRIVER   Carries the independent variables of the aerodynamic functions
into subroutine DATAFUN. These variables are α, M, δ_{SB}, δ_e,
δ_{BF}, δ_{el}, and δ_{er}.

FUNCNAM   Returns to the main program the results of the aerodynamic func-
tion table look up performed by the subroutine DATAFUN.

SIMULATION VALIDATION

Because of the discontinuous derivatives produced in the equations of
motion by the RCS, a Runge-Kutta integration was used in preference to a
predictor-corrector method. The transfer functions of the control system were
solved by difference equations resulting from the convolution method of appen-
dix D. At the time ARFDS was developed, the specific integration method to be
used by the onboard control system software was not known. In later versions of
ARFDS, the difference equations used for the control system will reflect the
actual techniques used onboard the shuttle.

18

Results from the completed real-time program were compared with results from an independent dynamic check. Results were compared for both a second- and a fourth-order Runge-Kutta scheme with various step sizes. In each case, complete entries were flown to insure that all guidance and control phases were exercised. The integration step size was chosen to be 1/25 of a second to match the actual minimum pulse width of the RCS thrusters. With this step size, the second-order integration scheme provided accurate results for the frequencies of the system studied.

PROGRAM APPLICATIONS

The ARFDS has been used in the evaluation of the space shuttle orbiter entry guidance and control systems. From these evaluations, suggestions have been made for modifications to the guidance and control systems which maintain vehicle performance while reducing the amount of RCS fuel required for the entry (ref. 3). For one study, the guidance system was replaced with one that issued step commands at selected times in order to evaluate the control system responses. In addition, the orbiter sensitivity to off-nominal aerodynamics has been studied, and modifications to the control system to make the orbiter more tolerant of the off-nominal aerodynamics have been identified.

CONCLUDING REMARKS

A nonlinear, six-degree-of-freedom simulation known as the automatic reentry flight dynamics simulator (ARFDS) has been developed at the Langley Research Center. The ARFDS model is for a vehicle which has constant mass properties (no main engine thrust) and whose attitudes are controlled by both aerodynamic surfaces and reaction control system thrusters.

The simulation was used in the evaluation of the space shuttle orbiter entry guidance and control systems. To date, it has been used to suggest modifications to the guidance and control systems that maintained vehicle performance while reducing the amount of required reaction control system fuel. Also, the sensitivity of the control system to off-nominal aerodynamics has been examined.

Langley Research Center
National Aeronautics and Space Administration
Hampton, VA 23665
February 10, 1977

APPENDIX A

ENTRY GUIDANCE SYSTEM

The baseline guidance scheme controls the entry by roll modulation while the space shuttle orbiter is flying a preselected angle-of-attack profile. The downrange is controlled by the magnitude of the roll angle, and the crossrange is controlled by multiple bank reversals. The guidance system outputs to the control system are commanded roll angle and commanded angle of attack.

The entry guidance system was developed by the NASA Johnson Space Center to follow a previously determined entry profile. This profile is achieved by dividing the entry into five major phases as illustrated in figure A1:

(1) Constant-attitude phase

(2) Constant-heat-rate phase

(3) Equilibrium glide phase

(4) Constant-drag phase

(5) Transition phase

The space shuttle orbiter is commanded to fly a constant-attitude trajectory until a specified total acceleration is attained. At this point, a constant stagnation heat-rate trajectory is flown through pullout to a relative velocity of 6248.4 m/sec (20 500 fps) or until the reference drag level becomes larger than that required to reach the target. If the latter condition is reached, the guidance scheme jumps to the constant-drag phase. If this condition is not met, an equilibrium glide profile is flown either until the reference drag level intersects the constant-drag profile required to reach the target and then jumps to the constant-drag phase or until the velocity drops off to 2743.2 m/sec (9000 fps). Whenever the velocity drops to 2743.2 m/sec (9000 fps), the transition phase is entered. During the transition phase, the commanded angle of attack is decreased to the value required at the terminal-area phase intersection, which occurs at a velocity of 457.2 m/sec (1500 fps) and at an altitude of approximately 21.3 km (70 000 ft).

Table AI shows the input constants that were used, and figure A2 shows the block diagram of the guidance laws.

SYMBOLS

A,B   parameters used in FORTRAN SIGN function

AK dD/dV for constant-heat-rate phase, used to define C23, sec^{-1}

ALDREF $(L/D)_{ref}$, used in controller

20

ALFM   reference equilibrium glide drag, m/sec^2 (ft/sec^2)

ALMN1   minimum roll command outside of lateral deadband YB, rad

ALMN2   minimum roll command inside of lateral deadband YB, rad

ALPCMD   angle-of-attack command, α_c, deg

AMAXI   maximum value function

ARC   distance from intersection with alinement circle to target, m (ft)

ARG $(L/D)_V/(L/D)$ used in roll-command equations, rad

ATK   radius of Earth, m (ft)

BA   equilibrium glide roll angle used in iteration loop, rad

BAD   final equilibrium glide roll angle, deg

BA1   first iteration equilibrium glide roll angle, deg

BA2   second iteration equilibrium glide roll angle, deg

CAGI   temporary calculation used in transition phase to calculate ALDREF and RDTREF, sec^2/m^2 (sec^2/ft^2)

CIGAR   transformation matrix from Earth-centered inertial (ECI) axes to geocentric axes

COSBADD   temporary calculation in equilibrium glide ranging phase used to calculate DREFP

CTH   great circle range from orbiter to target, rad

C4   coefficient used to calculate RDTREF, m/sec (ft/sec)

C5   coefficient used to calculate RDTREF, m/sec (ft/sec)

C11   parameter used to calculate RER1 and RDTREF, m^{-1} (ft^{-1})

C16   coefficient used to calculate LOD1, sec^2/m (sec^2/ft)

C17   coefficient used to calculate LOD1, sec/m (sec/ft)

C21   parameter used to calculate DREFP, RDTREF, SQ, and TT11, m/sec^2 (ft/sec^2)

C22   parameter used to calculate DREFP, E1, E2, RDTREF, SQ, TT11, and TT22, sec^{-1}

C23   parameter used to calculate C22, DREFP, E1, E2, SQ, TT11, and TT22, m^{-1} (ft^{-1})

D   total drag force, N (lb)

DBAR   distance from runway to alinement circle, m (ft)

DBB   increment in roll angle in equilibrium glide phase, deg

DELAZ   azimuth error, rad

DF   final drag level in transition phase, m/sec^2 (ft/sec^2)

DLIM   control system limit drag level in transition phase, m/sec^2 (ft/sec^2)

DRAG   current drag acceleration level, m/sec^2 (ft/sec^2)

DREFP   drag reference used in controller, m/sec^2 (ft/sec^2)

DT   planar range to target, m (ft)

DTH   angle between alinement circle center and tangency point, rad

DTR $= \pi/180$, rad/deg

DVHEAD   azimuth between runway and heading to tangency point of alinement circle, rad

D23   parameter used to calculate AK, m/sec^2 (ft/sec^2)

EEF   current energy level, m^2/sec^2 (ft^2/sec^2)

EEF4   reference energy level used in transition phase, m^2/sec^2 (ft^2/sec^2)

E1,E2   parameters used to calculate TT22

GAMMA   flight-path angle, rad

GCLAT   orbiter geocentric latitude, rad

GCLATT   target geocentric latitude, rad

GS   acceleration of gravity at sea level, m/sec^2 (ft/sec^2)

GSTART   acceleration required to initiate constant-heat-rate phase, g units

HA   current altitude, m (ft)

HADOT $= d(HA)/dt$, m/sec (ft/sec)

HDSER   parameter in oblate Earth correction term to RDTREF, m^3/sec (ft^3/sec)

22

HS	altitude scale height, m (ft)
IDFG2	switching flag in constant-drag phase
IDFG3	switching flag in transition phase
IFT	initialization flag in equilibrium glide phase
ISLECT	phase selector
ISTART	initialization flag
ISTP	iteration flag in equilibrium glide phase
ISTRT	flag indicating acceleration level equal to GSTART has been reached
ITR	iteration flag in transition phase
L/D	lift-drag ratio
$(L/D)_{ref}$	reference lift-drag ratio
$(L/D)_V$	lift-drag ratio in vertical plane
LMN	minimum value of LOD1
LN	natural logarithm function
LOD1	desired $(L/D)_V$
PSIE	current heading of orbiter, rad
PSIET	current heading to target, rad
RAZ	runway azimuth, rad
RCG	predicted range in constant-drag phase, m (ft)
RDC	parameter used in RDTREF calculation
RDTOLD	final RDTREF in equilibrium glide phase, m/sec (ft/sec)
RDTOL2	final RDTREF in constant-drag phase, m/sec (ft/sec)
RDTREF	altitude rate reference, m/sec (ft/sec)
REC	vector defining runway coordinate system
REC1	$= (REC)^{-1}$
REH	distance from center of Earth to vehicle, m (ft)

REQ predicted equilibrium glide phase range, m (ft)

RER1 parameter in range prediction for transition phase, m (ft)

RFF predicted range in constant-heat-rate phase, m (ft)

RG vector from orbiter to runway center, m (ft)

RGP vector from orbiter to alinement circle center, m (ft)

RK2ROL roll direction (+ right, - left)

RLON longitude of orbiter, rad

RLONT target longitude, rad

ROLLC roll-angle command, ϕ_c, rad

RPT desired range in transition phase, m (ft)

RPT1 range bias below velocity of 456.2 m/sec (1500 fps), m (ft)

RTE radius of Earth at runway, m (ft)

RTURN radius of alinement circle, m (ft)

R11 first iteration of range prediction in equilibrium glide and transition phases, m (ft)

R12 second iteration of range prediction in equilibrium glide and transition phases, m (ft)

SIGN(A,B) function which gives to value of A the algebraic sign of variable B

SQ parameter used in constant-heat-rate range prediction, sec^{-2}

SQQ parameter used in constant-heat-rate range prediction, sec^{-1}

TA vector from alinement circle tangency point to vehicle, m (ft)

TAP vector TA in geocentric coordinates, m (ft)

TARE target vector from alinement circle center to runway, m (ft)

TDREF parameter used in DREFP calculation in equilibrium glide phase, m/sec^2 (ft/sec^2)

TEMP temporary calculation in equilibrium glide phase, m (ft)

TRANGE great circle range from orbiter to target, m (ft)

T1   parameter used in calculation of ALDREF, m/sec^2 (ft/sec^2)

T2   constant drag level required to reach target, m/sec^2 (ft/sec^2)

TT11,TT22   parameters used in range prediction in constant-heat-rate phase, m (ft)

U   = $\left| DVHEAD \right|$, rad

UTARE   TARE unit vector

UXYZE   RG unit vector

V   Earth relative velocity, m/sec (ft/sec)

VBB   intersection velocity between constant-heat-rate phase and equilibrium glide phase, m/sec (ft/sec)

VCG   predicted intersection velocity between constant-drag phase and equilibrium glide phase, m/sec (ft/sec)

VINERT   inertial velocity, m/sec (ft/sec)

VOLD   final velocity in equilibrium glide phase, m/sec (ft/sec)

VOLD2   final velocity in constant-drag phase, m/sec (ft/sec)

VQ   predicted final velocity for constant-drag phase, m/sec (ft/sec)

VSAT   reference circular orbit velocity, m/sec (ft/sec)

V1OLD   value of VOLD-152.4, m/sec (ft/sec)

V2OLD   value of VOLD2-152.4, m/sec (ft/sec)

XLFAC   total acceleration, m/sec^2 (ft/sec^2)

XLOD   L/D of vehicle with undeflected control surfaces including viscous effects

XYZE   geocentric position vector, m (ft)

YB   lateral deadband (amount of overshoot that guidance system will allow before commanding roll reversal), rad

TABLE AI.- ENTRY GUIDANCE INPUT CONSTANTS

Parameter	Value	Unit
ALFM	7.62 (25)	m/sec^2 (ft/sec^2)
ALMN1	0.7986355	rad
ALMN2	0.9659258262	rad
ATK	6 366 707.02 (2.08881464×10^7)	m (ft)
DBAR	14 360.4 (48 000)	m (ft)
DF	5.819 (19.09)	m/sec^2 (ft/sec^2)
EEF4	185 806.08 (2.0×10^6)	m^2/sec^2 (ft^2/sec^2)
GCLATT	34.55577617	deg
GS	9.815 (32.2)	m/sec^2 (ft/sec^2)
GSTART	0.05	g units
RAZ	-0.7679448709	rad
RLONT	-120.5338	deg
RPT	421 885.6 (1.3841391×10^6)	m (ft)
RPT1	23 150 (75 951.4)	m (ft)
RTE	637 329 8.953 (2.090977347×10^7)	m (ft)
RTURN	4 632.96 (15 200)	m (ft)
VQ	2 133.6 (7 000)	m/sec (ft/sec)
VSAT	7 853.54 (25 766.2)	m/sec (ft/sec)

START

↓

TARGETING

↓

*ISLECT = 0 — NØ → (right) → A,B,C,D;ISLECT

YES ↓

XLFAC ≧ GSTART*GS — YES → ISLECT = 1

NØ ← CØNSTANT ATTITUDE PHASE ← E

(A)

↓

V<6248.4 — NØ → DREFP>T2 — NØ → CØNSTANT-HEAT-RATE PHASE → E

YES ↓   YES ↓

ISLECT = 2   ISLECT = 3

↓   ↓

B   C

GØ TØ

ENTRY POINT
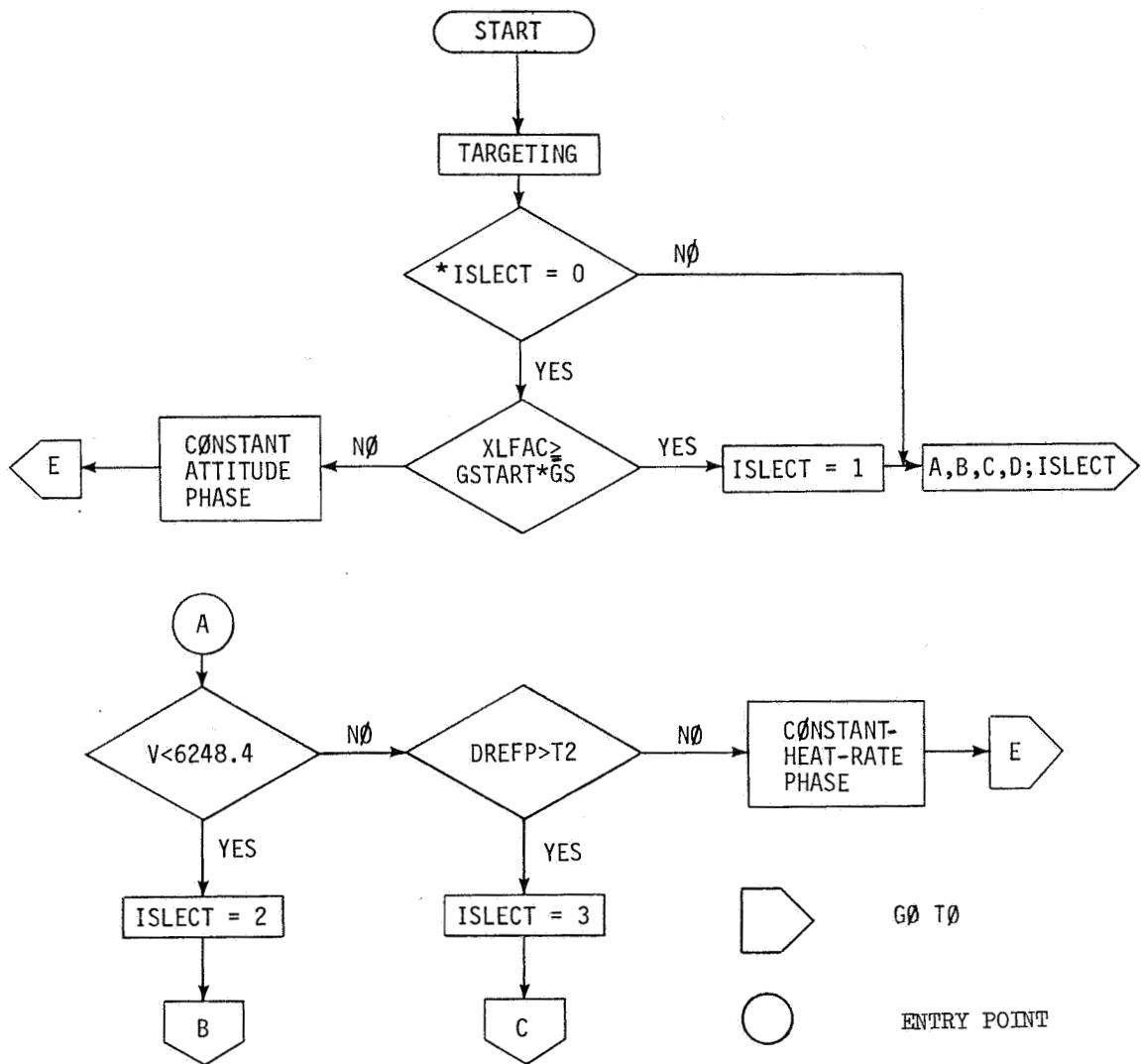
*ISLECT = 0 initially

Figure A1.- Entry guidance system flow diagram.

Figure A1.- Concluded.

```
┌─────────────────┐
│   INITIALIZE    │
└─────────────────┘
         │
         ▼
┌──────────────────────────────┐
│  DATA   GS,ATK,GSTART,RPT,    │
│         VSAT,VQ,ALMN1,ALFM,   │
│         RPT1,RAZ,RTE,DBAR,    │
│         RTURN,ALMN2,EEF4,DF,  │
│         RLØNT,GCLATT          │
└──────────────────────────────┘
         │
         ▼
```

$$REC(1,1) = -CØS(RAZ)*SIN(GCLATT)*CØS(RLØNT)-SIN(RAZ)*SIN(RLØNT)$$
$$REC(1,2) = -CØS(RAZ)*SIN(GCLATT)*SIN(RLØNT)+SIN(RAZ)*CØS(RLØNT)$$
$$REC(1,3) = CØS(RAZ)*CØS(GCLATT)$$
$$REC(2,1) = SIN(RAZ)*SIN(GCLATT)*CØS(RLØNT)-CØS(RAZ)*SIN(RLØNT)$$
$$REC(2,2) = SIN(RAZ)*SIN(GCLATT)*SIN(RLØNT)+CØS(RAZ)*CØS(RLØNT)$$
$$REC(2,3) = -SIN(RAZ)*CØS(GCLATT)$$
$$REC(3,1) = -CØS(GCLATT)*CØS(RLØNT)$$
$$REC(3,2) = -CØS(GCLATT)*SIN(RLØNT)$$
$$REC(3,3) = -SIN(GCLATT)$$

```
         │
         ▼
┌──────────────────────────────┐
│  ISTRT = 0                    │
│  ISTART = 0                   │
│  ISLECT = 0                   │
│  IFT = 0                      │
│  IDFG2 = IDFG3 = 0            │
│  ISTP = 0                     │
└──────────────────────────────┘
         │
         ▼
┌──────────────────────────────┐
│  BA1 = 0                      │
│  VBB = 6248.4                 │
│  YB  = .3054                  │
└──────────────────────────────┘
         │
         ▼
┌──────────────────────────────┐
│     GUIDANCE EQUATIØNS        │
└──────────────────────────────┘
         │
         ▼
        ╱╲
       ╱  ╲
      ╱ TIME ╲       NØ     ┌────┐
     ╱ TØ SAMPLE ╲ ───────▶ │ DD ⟩
      ╲GUIDANCE?╱           └────┘
       ╲  ╱
        ╲╱
         │ YES
         ▼
        ┌──┐
        │ A⟩
        └──┘
```

Figure A2.– Analytic drag control entry guidance system block diagram.
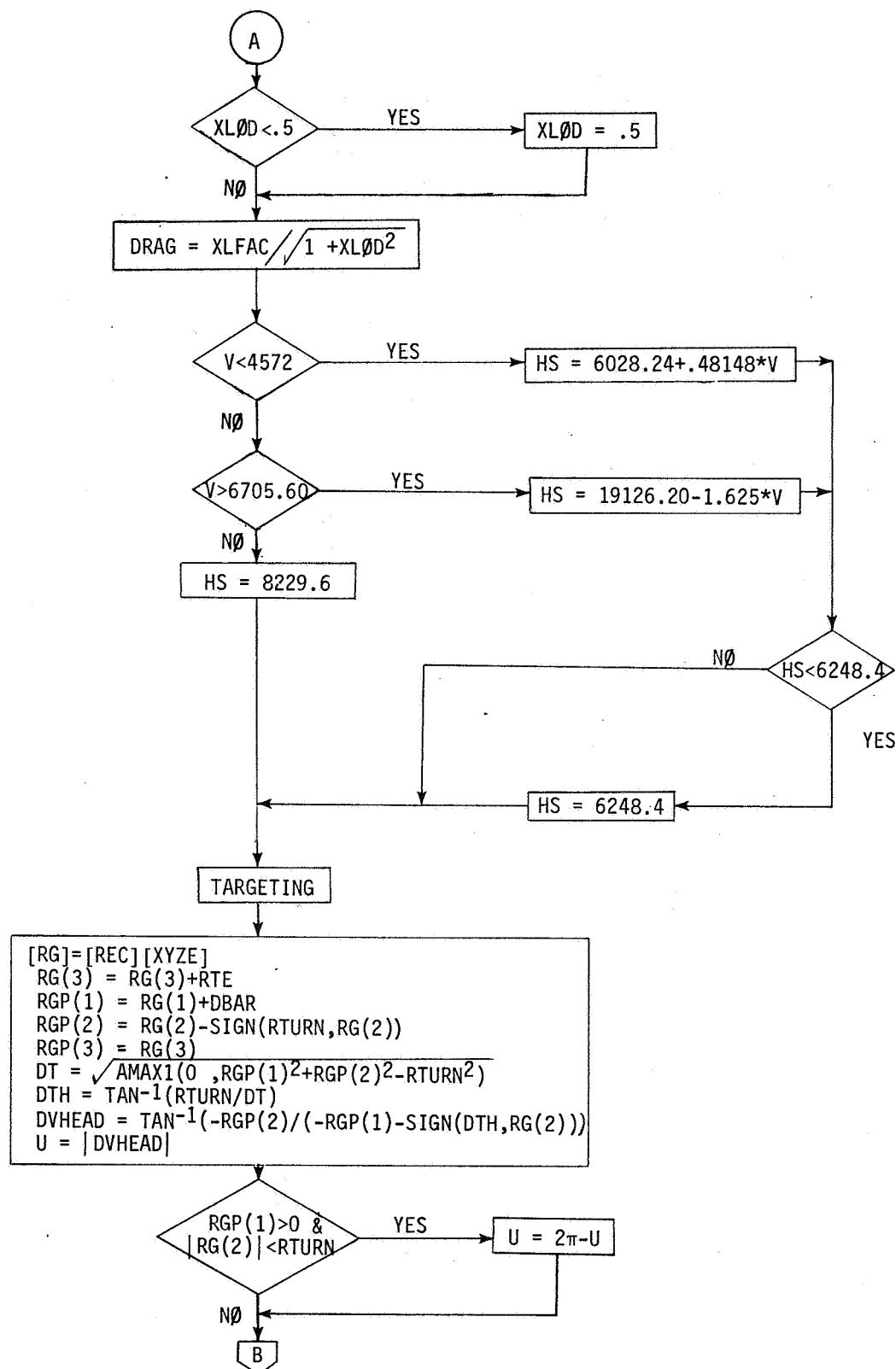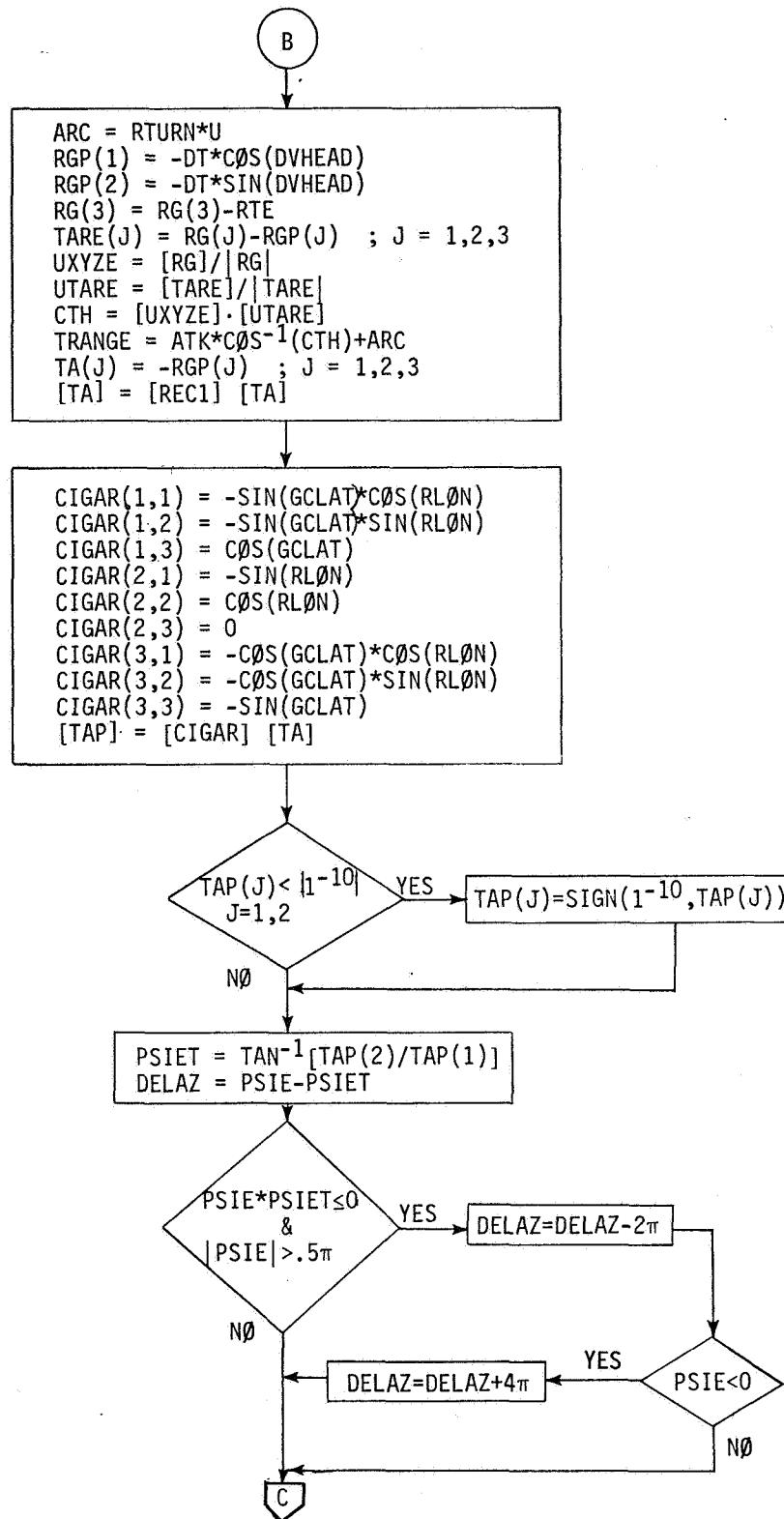
Figure A2.- Continued.

B

```
ARC = RTURN*U
RGP(1) = -DT*CØS(DVHEAD)
RGP(2) = -DT*SIN(DVHEAD)
RG(3) = RG(3)-RTE
TARE(J) = RG(J)-RGP(J)   ; J = 1,2,3
UXYZE = [RG]/|RG|
UTARE = [TARE]/|TARE|
CTH = [UXYZE]·[UTARE]
TRANGE = ATK*CØS⁻¹(CTH)+ARC
TA(J) = -RGP(J)   ; J = 1,2,3
[TA] = [REC1] [TA]
```

$$TRANGE = ATK*CØS^{-1}(CTH)+ARC$$

```
CIGAR(1,1) = -SIN(GCLAT)*CØS(RLØN)
CIGAR(1,2) = -SIN(GCLAT)*SIN(RLØN)
CIGAR(1,3) = CØS(GCLAT)
CIGAR(2,1) = -SIN(RLØN)
CIGAR(2,2) = CØS(RLØN)
CIGAR(2,3) = 0
CIGAR(3,1) = -CØS(GCLAT)*CØS(RLØN)
CIGAR(3,2) = -CØS(GCLAT)*SIN(RLØN)
CIGAR(3,3) = -SIN(GCLAT)
[TAP] = [CIGAR] [TA]
```

TAP(J)< |1⁻¹⁰|
J=1,2 — YES → TAP(J)=SIGN(1⁻¹⁰,TAP(J))

NØ

```
PSIET = TAN⁻¹[TAP(2)/TAP(1)]
DELAZ = PSIE-PSIET
```

$$PSIET = TAN^{-1}[TAP(2)/TAP(1)]$$

PSIE*PSIET≤0
&
|PSIE|>.5π — YES → DELAZ=DELAZ-2π

NØ

DELAZ=DELAZ+4π ← YES — PSIE<0

NØ

C

Figure A2.- Continued.

Figure A2.- Continued.

Figure A2.- Continued.

Figure A2.- Continued.

Figure A2.- Continued.

Figure A2.- Continued.

Figure A2.- Continued.

$\boxed{Q}$

$$D23 = -GS*(VBB^2/VSAT^2 - 1)/(1.4*C\emptyset S(BA))$$
$$AK = -3.5*D23/VBB$$
$$C23 = -AK/3352.8$$
$$C22 = 15849.6*C23$$
$$C21 = D23 - C22*VBB - C23*VBB^2$$
$$TT11 = -.5*LN((C21 + C22*VBB + C23*VBB^2)/(C21 + C22*V + C23*V^2))/C23$$
$$SQ = 4.*C23*C21 - C22^2$$

SQ < 0 — YES

N∅

$$SQQ = \sqrt{SQ}$$
$$TT22 = C22*(TAN^{-1}(2*C23*VBB/SQQ + C22/SQQ)$$
$$-TAN^{-1}(2*C23*V/SQQ + C22/SQQ))/(C23*SQQ)$$

$$SQQ = \sqrt{-SQ}$$
$$E1 = (2*C23*VBB + C22 - SQQ)/(2*C23*VBB + C22 + SQQ)$$
$$E2 = (2*C23*V + C22 - SQQ)/(2*C23*V + C22 + SQQ)$$
$$TT22 = C22*LN(E1/E2)/(2*C23*SQQ)$$

$\boxed{R}$

$$RFF = TT11 + TT22$$

VCG > V — YES → ∅

N∅

V < 6248.4 — YES → $RFF = 0$

N∅

$$REQ = .5*TEMP*LN((VCG^2 - VSAT^2)/(VBB^2 - VSAT^2))$$
$$RCG = (VCG^2 - VQ^2)/(2*ALFM)$$

S, T, ∅; ISTP
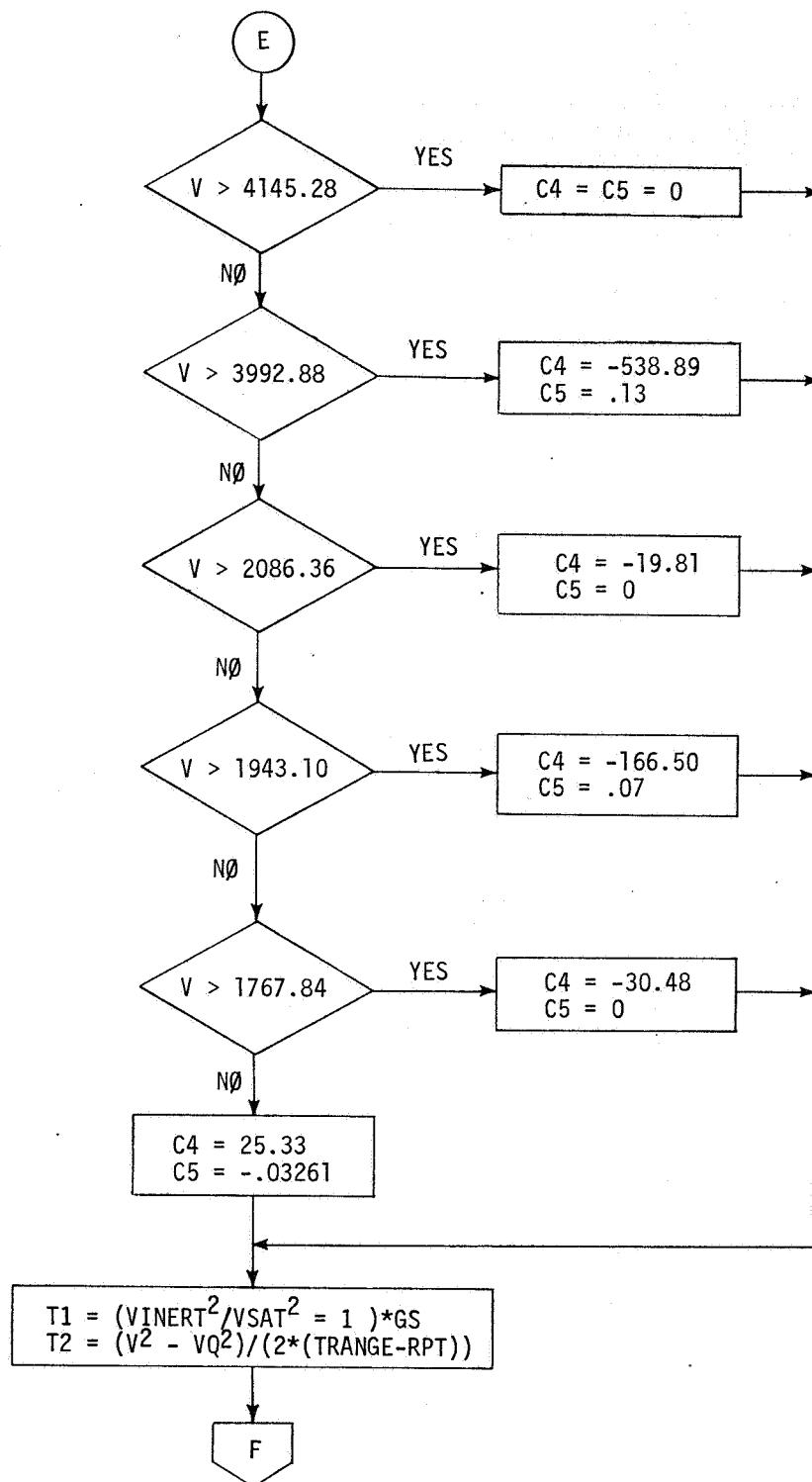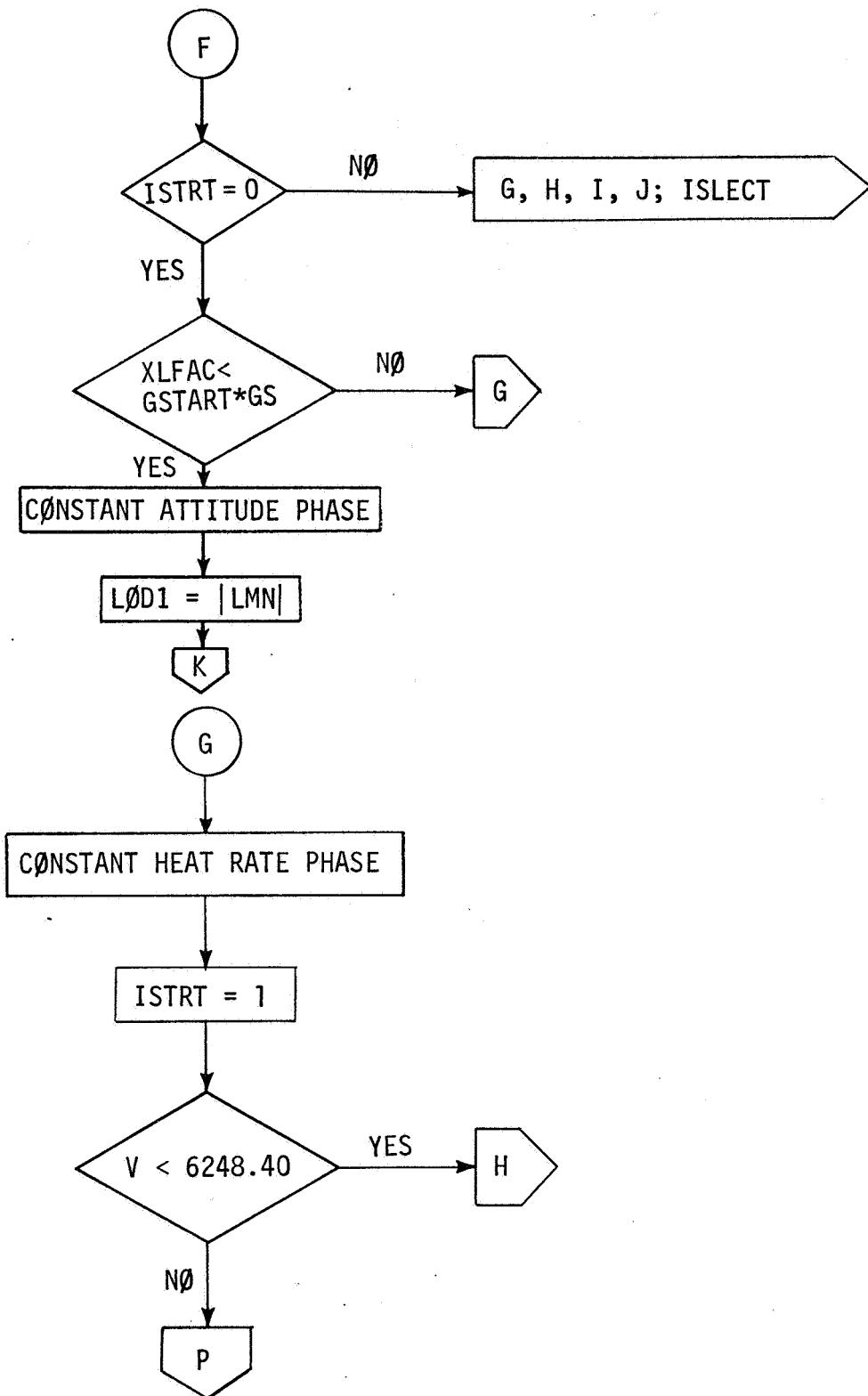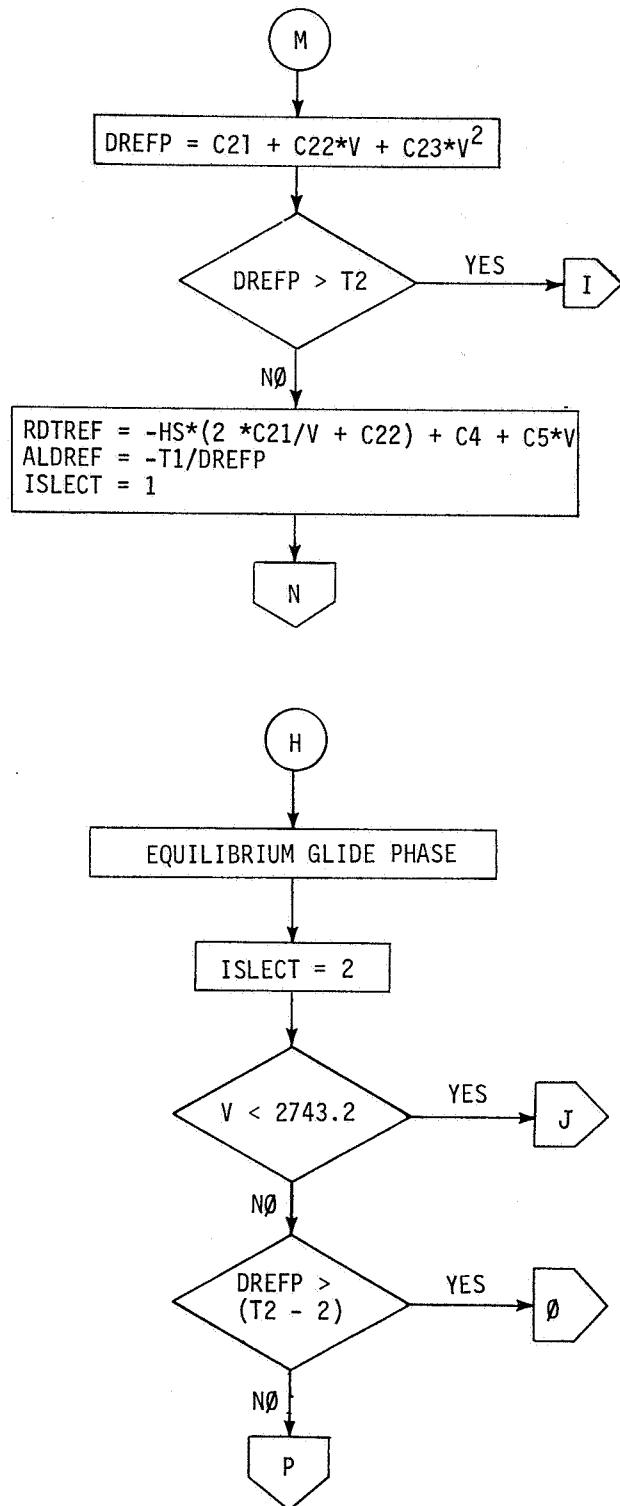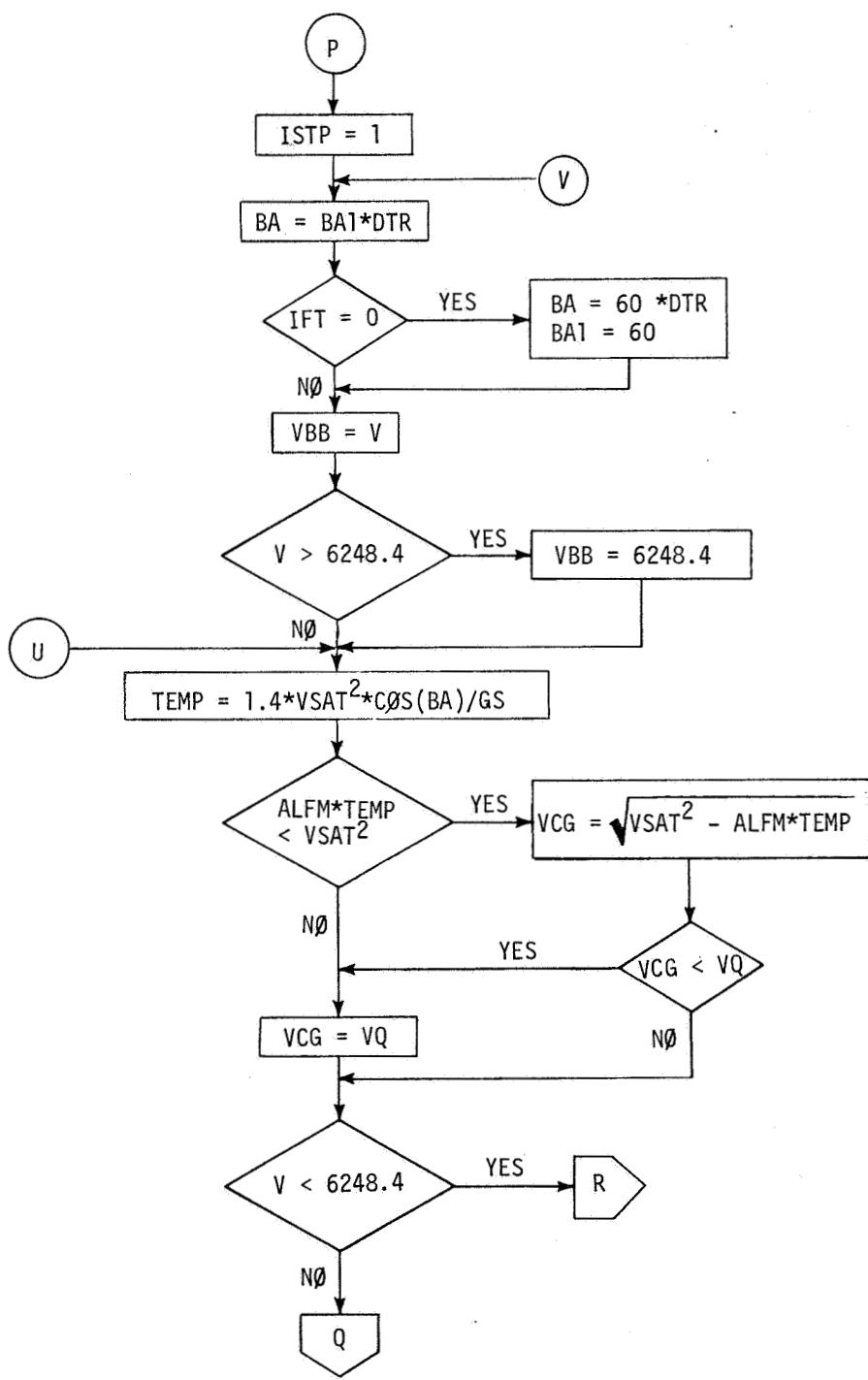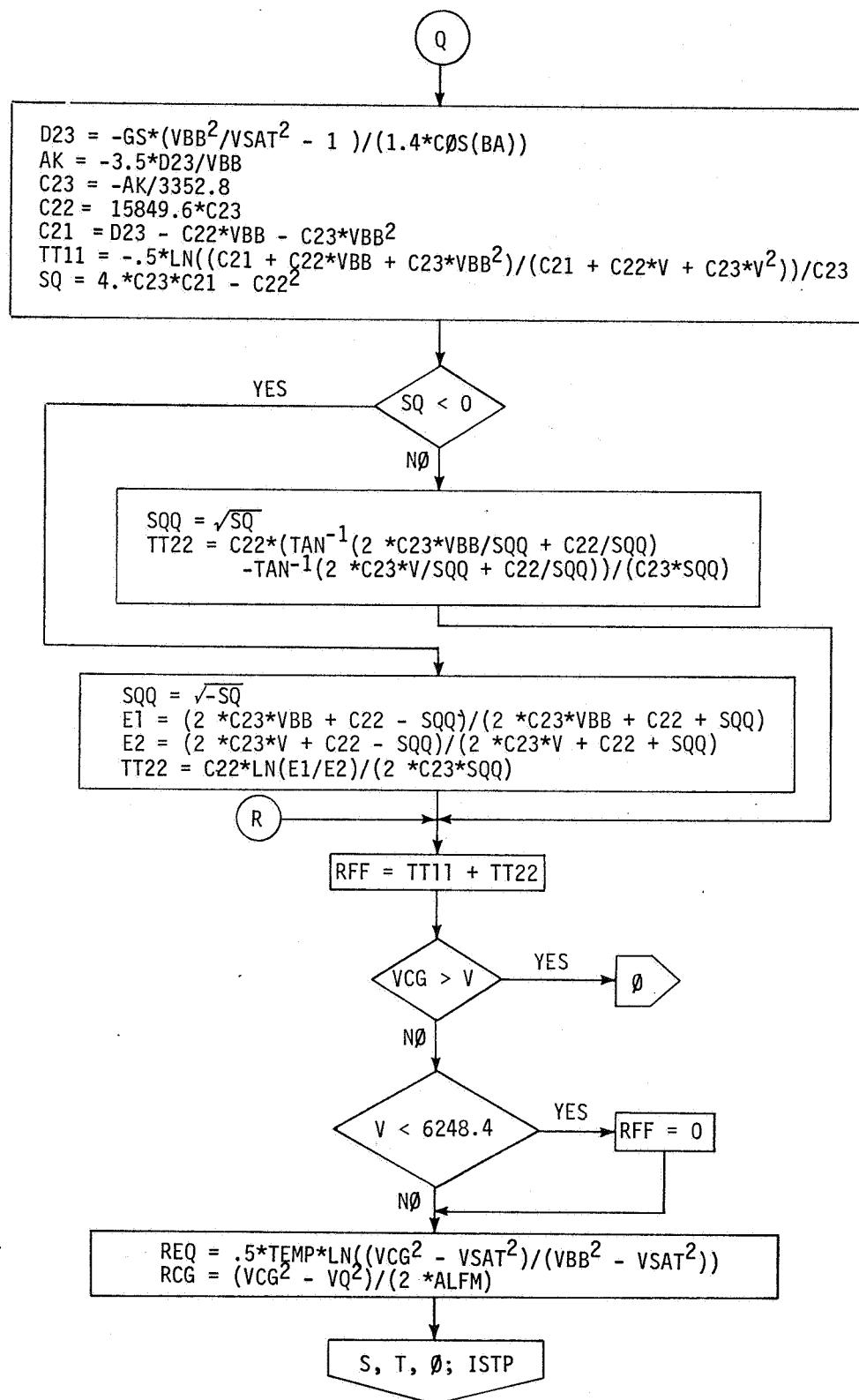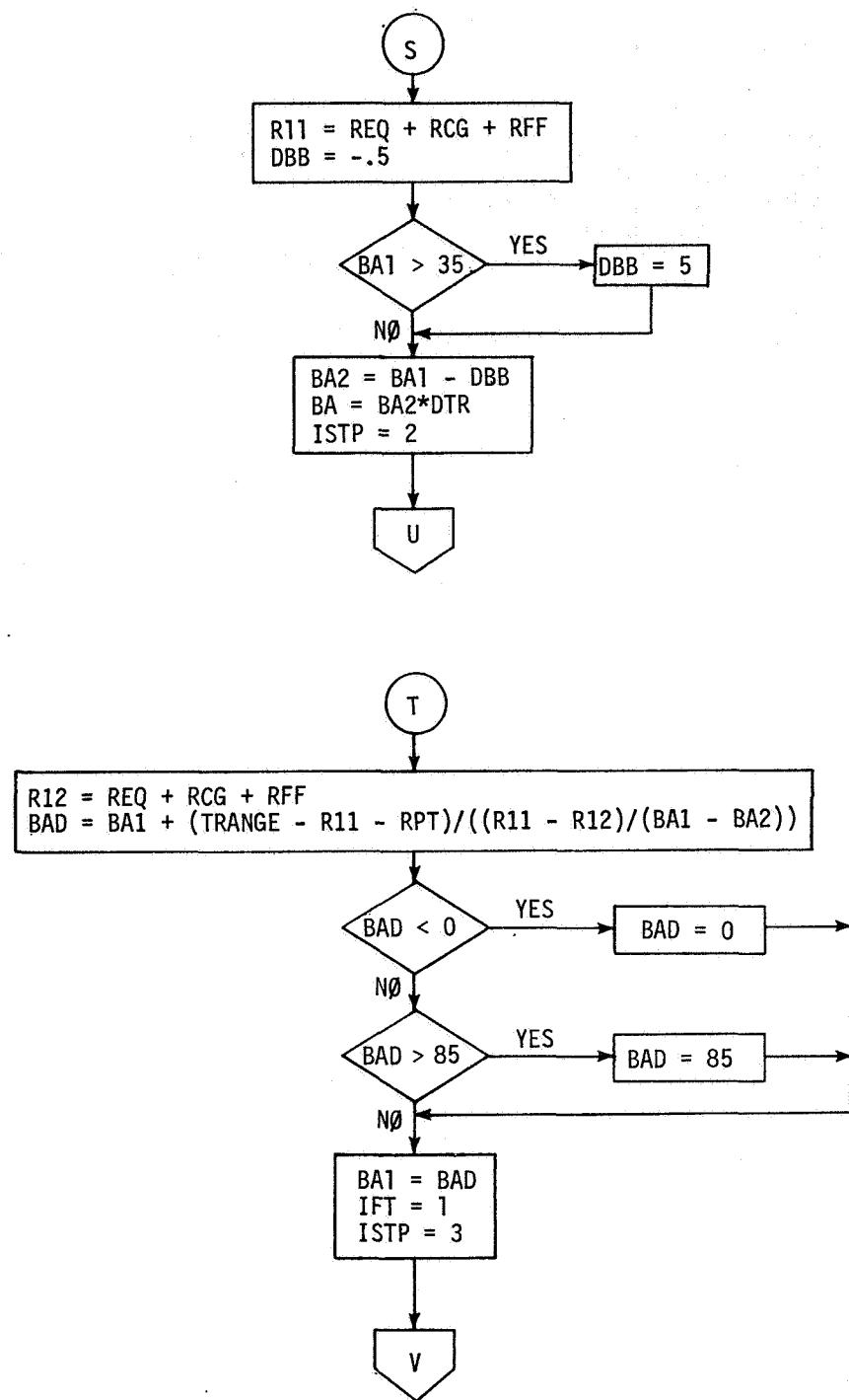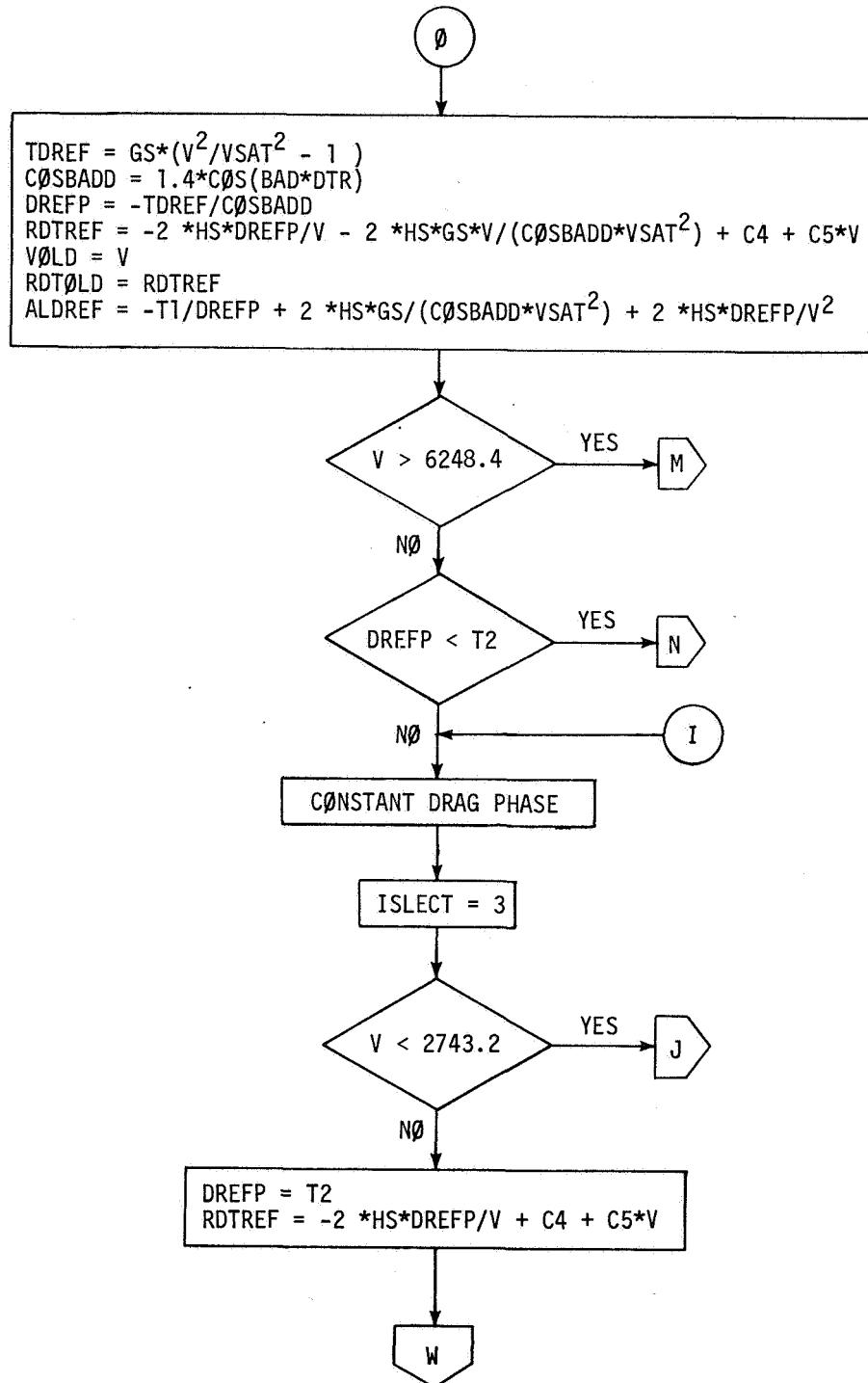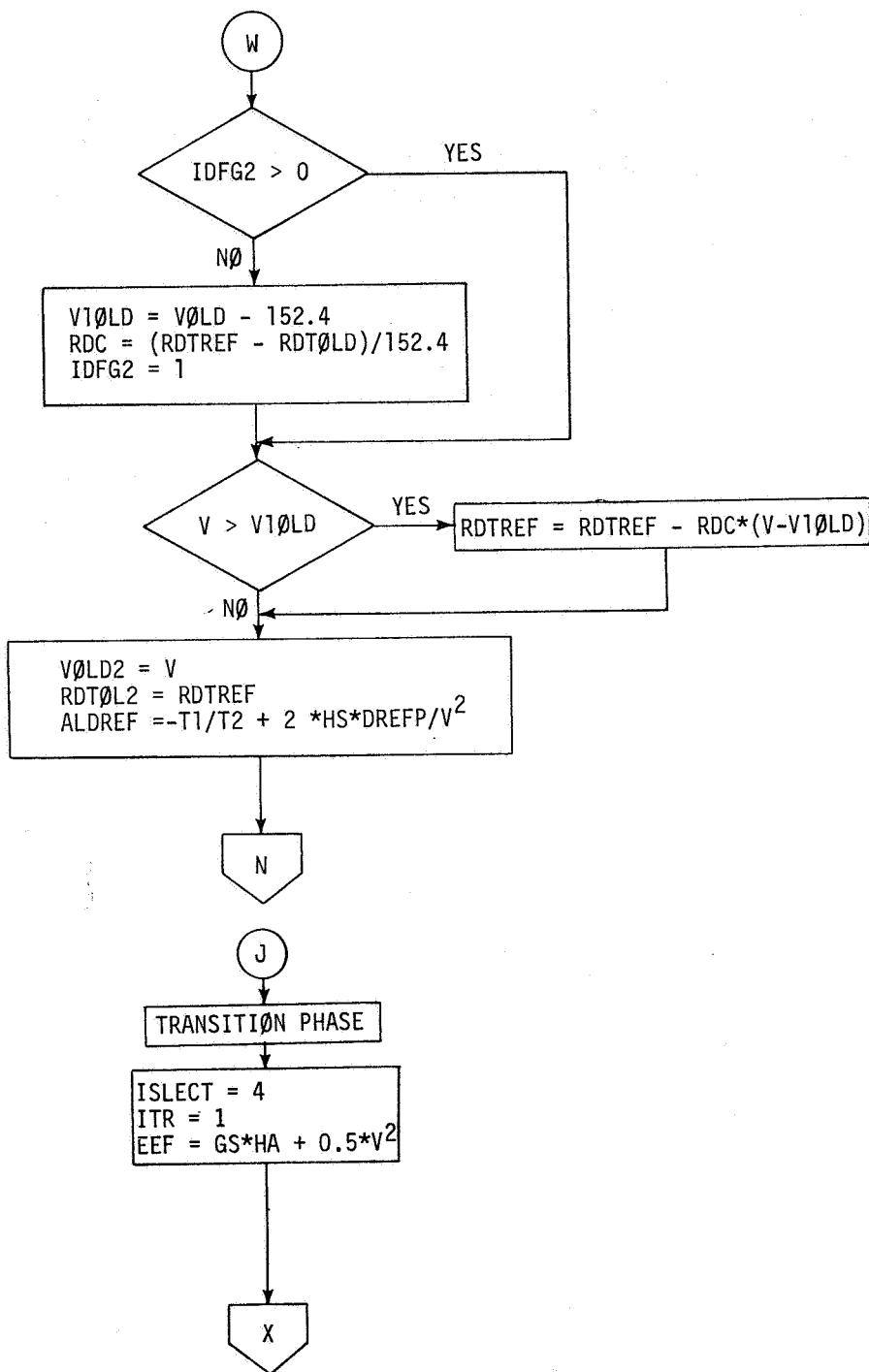
Figure A2.- Continued.

Figure A2.- Continued.

Figure A2.- Continued.
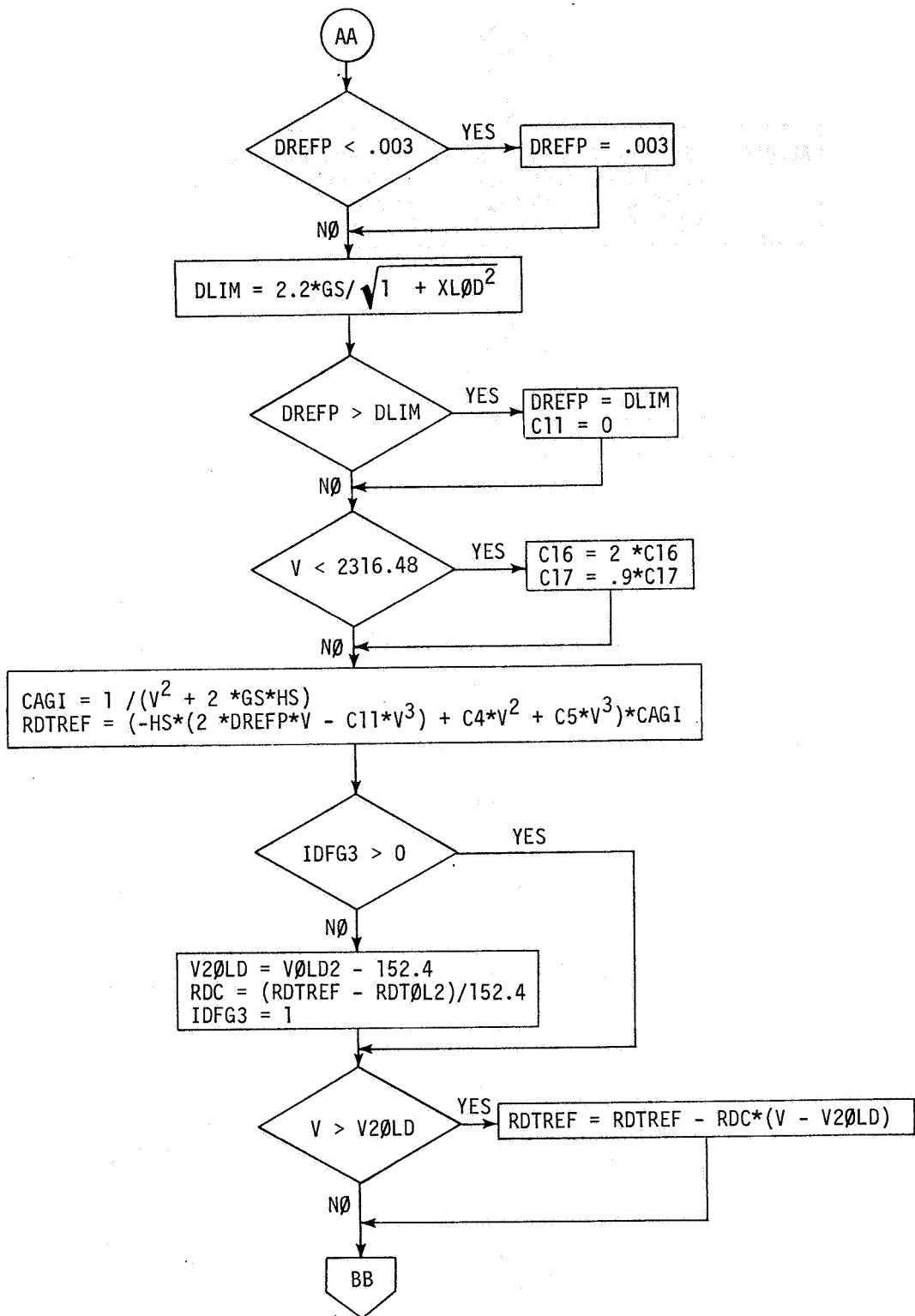
Figure A2.- Continued.

Figure A2.- Continued.
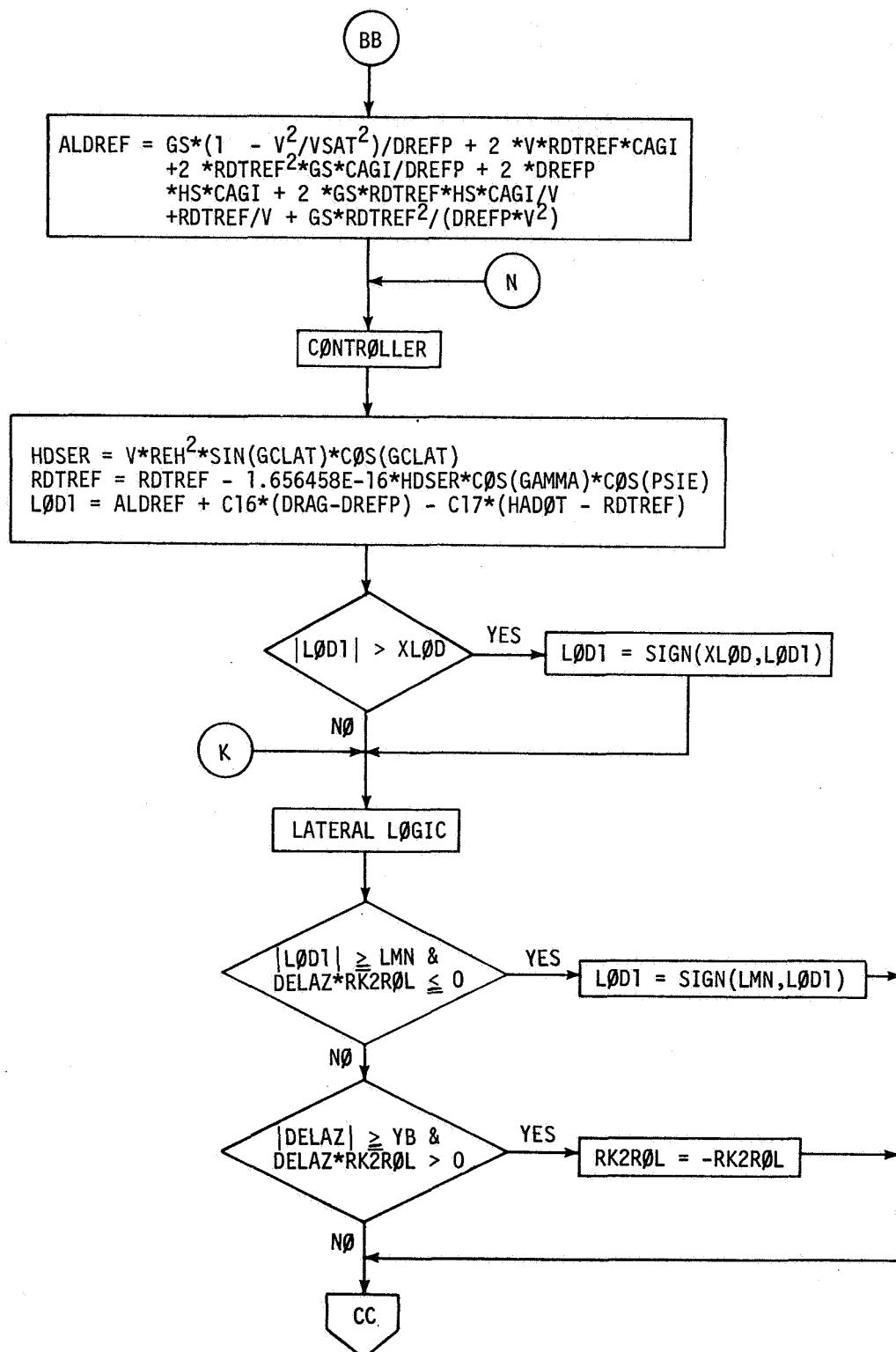
Figure A2.- Continued.

Figure A2.- Continued.
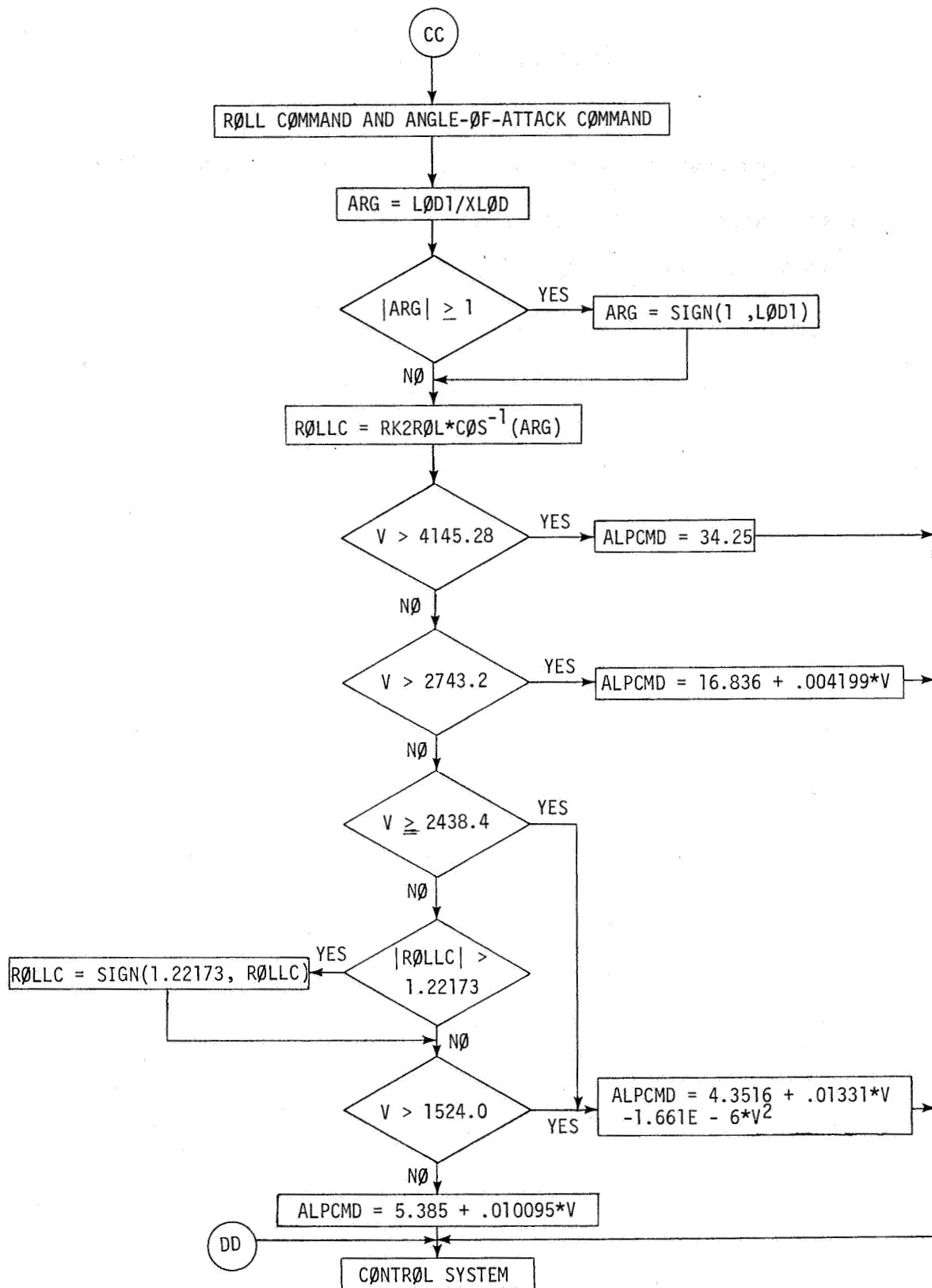
Figure A2.- Concluded.

APPENDIX B

ENTRY DIGITAL AUTOPILOT

Symbols

a_y	side acceleration at center of gravity, m/sec^2 (ft/sec^2)
C_{he}	elevon hinge-moment coefficient
$C_{h\beta}$	$= \partial(\text{rudder hinge moment})/\partial\beta$
$C_{h\delta_r}$	$= \partial(\text{rudder hinge moment})/\partial\delta_r$
C_{LN}	rolling-moment coefficient due to yaw RCS
C_{ML}	pitching-moment coefficient due to roll RCS
C_{MN}	pitching-moment coefficient due to yaw RCS
C_{NL}	yawing-moment coefficient due to roll RCS
c_e	elevon reference chord, m (ft)
c_r	rudder reference chord, m (ft)
DEMX	maximum elevon rate, deg/sec
DRMX	maximum rudder rate, deg/sec
E_P	pitch RCS error signal
E_R	roll RCS error signal
E_Y	yaw RCS error signal
$f(\delta e)$	function of δ_e used to limit $\delta_{a,c}$, deg
g	acceleration due to gravity, m/sec^2 (ft/sec^2)
Hm_e	elevon hinge moment, N-m (lbf-ft)
Hm_r	rudder hinge moment, N-m (lbf-ft)
K_L	rolling-moment RCS amplification factor
K_{MD}	pitching-moment RCS amplification factor due to down-firing jets
K_{MU}	pitching-moment RCS amplification factor due to up-firing jets
K_N	yawing-moment RCS amplification factor

46

K_p	aileron gain
K_α	elevator gain
K_{δ_r}	rudder gain
L_{RCS}	rolling moment due to RCS, N-m (lbf-ft)
L_{RJ}	ideal rolling moment due to firing of one roll jet, N-m (lbf-ft)
M_{PJ}	ideal pitching moment due to firing of one pitch jet, N-m (lbf-ft)
M_{RCS}	pitching moment due to RCS, N-m (lbf-ft)
N_{RCS}	yawing moment due to RCS, N-m (lbf-ft)
N_{YJ}	ideal yawing moment due to firing of one yaw jet, N-m (lbf-ft)
PJN	number of negative pitch jets firing
PJP	number of positive pitch jets firing
p	roll rate, deg/sec
q	pitch rate, deg/sec
$\bar{q}$	dynamic pressure, Pa (psf)
RJN	number of negative roll jets firing
RJP	number of positive roll jets firing
r	yaw rate, deg/sec
r'	$= r - (180g \sin \phi \cos \theta)/\pi V_R$, deg/sec
S_e	elevon reference area, m^2 (ft^2)
S_r	rudder reference area, m^2 (ft^2)
s	Laplace operator
YJN	number of negative yawing jets firing
YJP	number of positive yawing jets firing
δ_a	aileron deflection, deg
$\delta_{a,c}$	commanded aileron deflection, deg
$\delta_{a,UD}$	aileron deflection at up-down counter, deg

$\delta_{e,c}$ commanded elevon deflection, deg

$\delta_{el,c}$ commanded left elevon panel deflection, deg

$\delta_{e,lm}$ maximum change in elevon command that can be realized in one control cycle, deg

δ_{er} right elevon panel deflection, deg

$\delta_{er,c}$ commanded right elevon panel deflection, deg

$\delta_{e,t}$ initial elevator setting, deg

δ_r rudder deflection, deg

$\delta_{r,c}$ commanded rudder deflection, deg

$\delta_{r,lm}$ maximum change in rudder command that can be realized in one control cycle, deg

Entry Digital Autopilot Filters and Controls

The entry digital autopilot (EDAP) is designed to fly the space shuttle orbiter automatically from deorbit to the terminal area interface which occurs at an altitude of approximately 21.3 km (70 000 ft) with a velocity of 457.2 m/sec (1500 ft/sec). The EDAP directs both the reaction control system (RCS) and the aerodynamic control surfaces (δ_e, δ_a, δ_r, δ_{BF}, and δ_{SB}). Two types of signal limiting filters are used in this autopilot. The first type is illustrated in sketch (a):
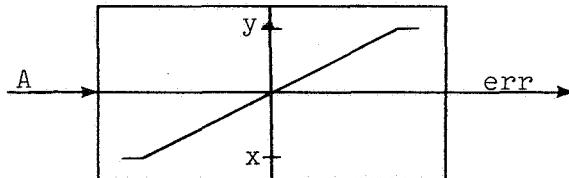
Sketch (a)

This filter limits the value of the quantity A to be between x and y. The second type, called a "hysteresis filter," can appear in one of two ways (see sketch (b)):

Sketch (b)

48
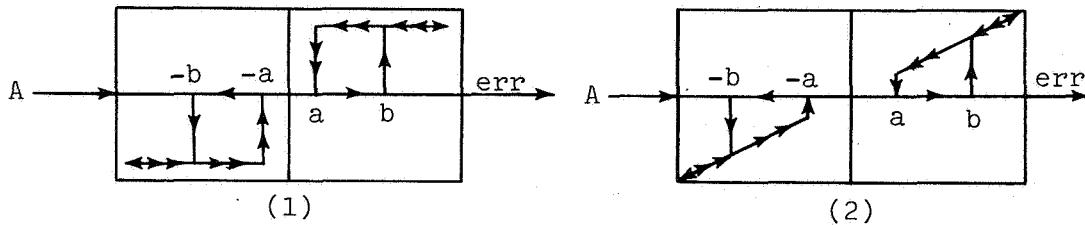
As A increases from zero, "err" remains zero until point "b" is reached. At this time, "err" becomes the value indicated (either a constant value if filter is type 1 or equal to A if filter is type 2) and continues as A increases. As A starts to decrease, it remains the value indicated until point "a" is reached, where "err" becomes zero again. A similar situation would exist for an A decreasing from zero.

Aerodynamic Control

The speed-brake δ_{SB} and body-flap δ_{BF} deflection schedules are shown in figure B1, where δ_{SB} is determined from a preset velocity schedule and δ_{BF} is dependent on the center-of-gravity location. Figures B2 to B10 are block diagrams of the various elements of the EDAP.

The elevons are used for both elevator δ_e and aileron δ_a functions. The elevator-command block diagram is shown in figure B2. The aileron functions in one of two ways depending on the flight regime. For $\alpha \leq 18^\circ$ and $M \leq 5$, the aileron is used for roll attitude ϕ control (fig. B3(a)); when these conditions are not present, the ailerons are used for turn coordination (fig. B3(b)).

If the orbiter has a lateral center-of-gravity offset, the number of positive yaw and roll thruster firings will not equal the number of negative yaw and roll thruster firings because of the induced sideslip. By counting the number of positive and negative yaw and roll thruster firings, it is possible to establish the "steady-state" aileron deflection required to offset this induced sideslip. This is the role of the "up-down counter" shown in figure B4. The numbers in parentheses in the block diagrams are the expressed values in U.S. Customary Units.

Figure B5 shows that the commanded left and right elevon deflections are functions of δ_e, $\delta_{a,c}$, and $\delta_{e,c}$. The rudder deflection δ_r (fig. B6) is used for turn coordination when the aileron is used for roll control. If the ailerons are being used for turn coordination, the rudder is inoperative.

Reaction Control System

The pitch reaction control system (RCS) (fig. B7) is operative for $\bar{q}$ less than 958 Pa (20 psf). In this regime it is used, along with the elevator control system, for longitudinal control.

The roll RCS (fig. B8) is operative for $\bar{q}$ less than 479 Pa (10 psf) and is used, along with the aileron control system, for turn coordination.

The yaw RCS (fig. B9(a)) aids the rudder control system in maintaining turn coordination. If the conditions are such that the ailerons are used for turn coordination, the yaw RCS (fig. B9(b)) is used for roll attitude ϕ control.

The RCS error signals command control jet firings, and the particular jets fired are a function of the sign and magnitude of the error signals. The sign

of the error signal determines vehicle motion direction; that is, a positive signal fires rockets that yield a positive (+) roll, pitch, or yaw motion, and negative signals command negative (-) motions. The number of jets fired is proportional to the magnitude of the error signals. (See figs. B7 to B9.)

Aerodynamic Interference Equations

The RCS model uses the following equations to account for aerodynamic interference:

$$L_{RCS} = L_{RJ}\left[(RJP - RJN)K_L + (YJP - YJN)C_{LN}\right]$$

$$M_{RCS} = M_{PJ}\left[(PJP)K_{MU} - (PJN)K_{MD} + (YJP + YJN)C_{MN} + (RJP + RJN)C_{ML}\right]$$

$$N_{RCS} = N_{YJ}\left[(YJP - YJN)K_N + (RJP - RJN)C_{NL}\right]$$

The values for the coefficients are shown in table BI.

Integration Schemes

To integrate the linear first-order differential equations in the control system, the convolution technique of appendix D is used. This is a one-pass scheme that has demonstrated a high degree of accuracy in other real-time simulations, including piloted simulations. (For example, see ref. 2.) The control actuators (fig. B10) are integrated the same way, except that provisions are made for both position and rate limits.

TABLE BI.- INTERFERENCE RCS VALUES

Jet moment	Value, N-m (lbf-ft)
L_{RJ}	11 185.5 (8 250.0)
M_{PJ}	38 325.6 (28 267.5)
N_{YJ}	38 878.8 (28 675.5)

$\bar{q}$, Pa (psf)	K_L	K_{MU}	K_{MD}	K_N	C_{LN}	C_{MN}	C_{ML}	C_{NL}
0	0.746	1.0	0.740	1.02	-0.624	0	0.130	-0.141
119.7 (2.5)	.688	1.0	.678	1.02	-.953	.038	.161	-.115
239.4 (5.0)	.630	1.0	.616	1.02	-1.069	.076	.192	-.111
478.8 (10.0)	.533	1.0	.541	1.02	-1.069	.114	.230	-.111
718.2 (15.0)	.475	1.0	.512	1.02	-1.069	.133	.244	-.111
957.6 (20.0)	.436	1.0	.493	1.02	-1.069	.152	.253	-.111

M	K_N	C_{LN}	C_{MN}
$\bar{q} > 957.6$ Pa (20 psf)			
2	1.02	-0.701	0.076
5	1.02	-.934	.076
10	1.02	-1.166	.076
30	1.02	-1.069	.152

Body-flap δ_{BF} schedule

Forward center of gravity $\quad \delta_{BF} = -11.7^0$

Aft center of gravity $\qquad \delta_{BF} = 16.3^0$

Speed-brake δ_{SB} schedule
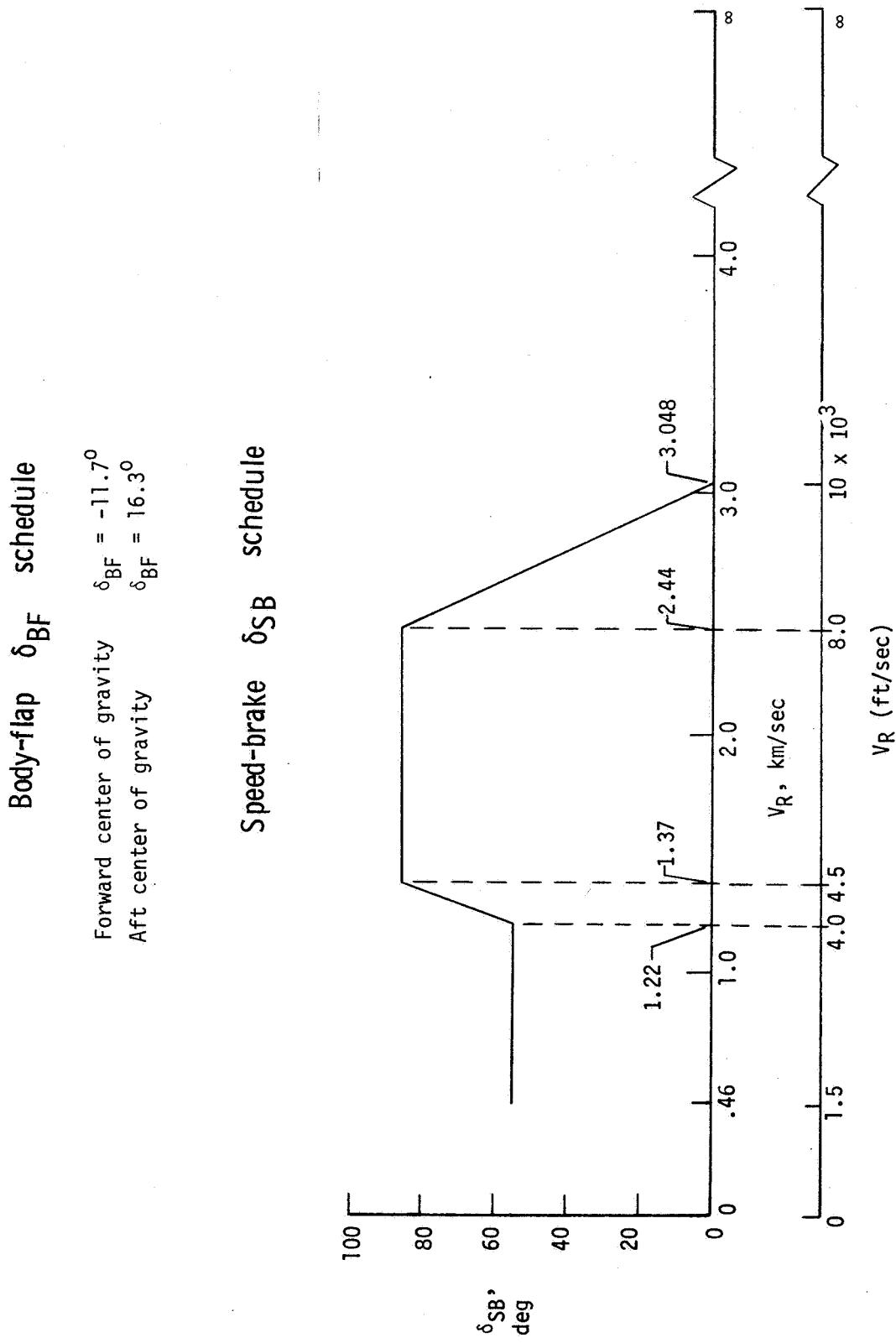
Figure B1.- Body-flap and speed-brake schedules.
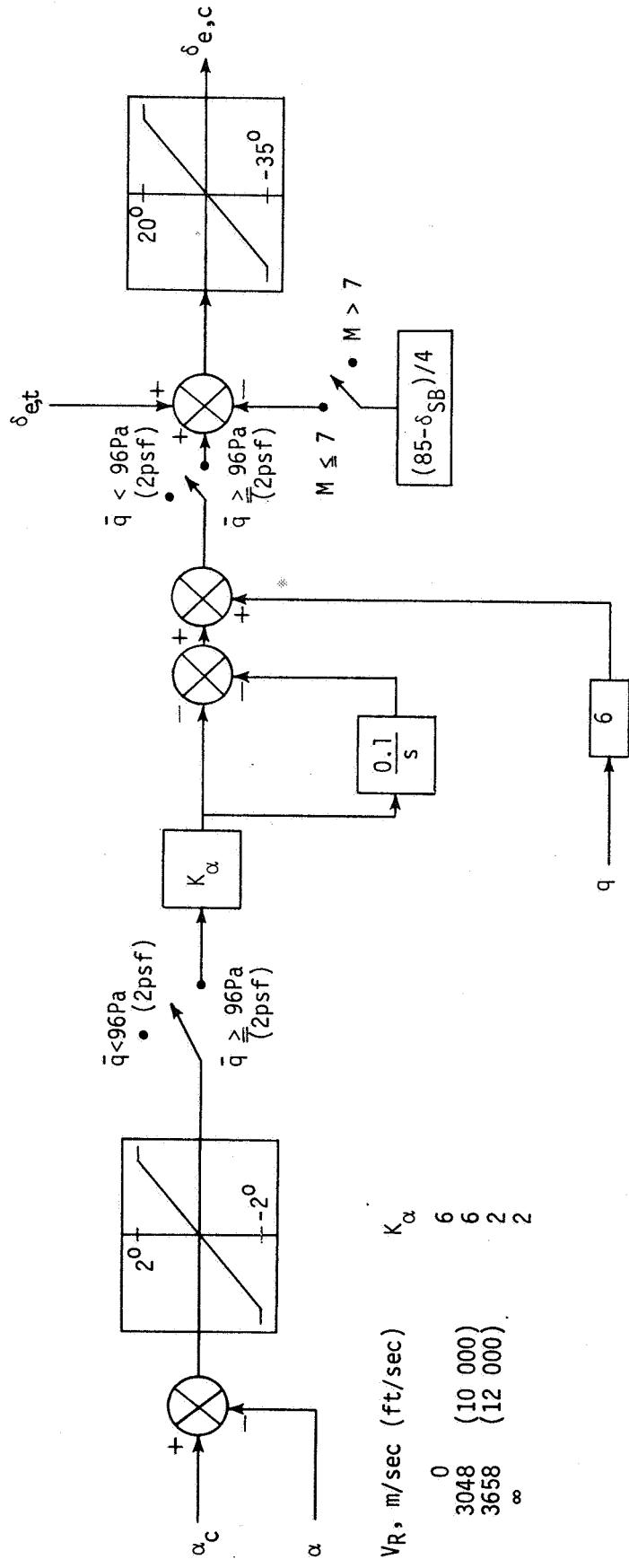
Figure B2.- Elevator-command block diagram. K_α is linearly varied between indicated points.

V_R, m/sec (ft/sec)	K_α
0	6
3048 (10 000)	6
3658 (12 000)	2
∞	2
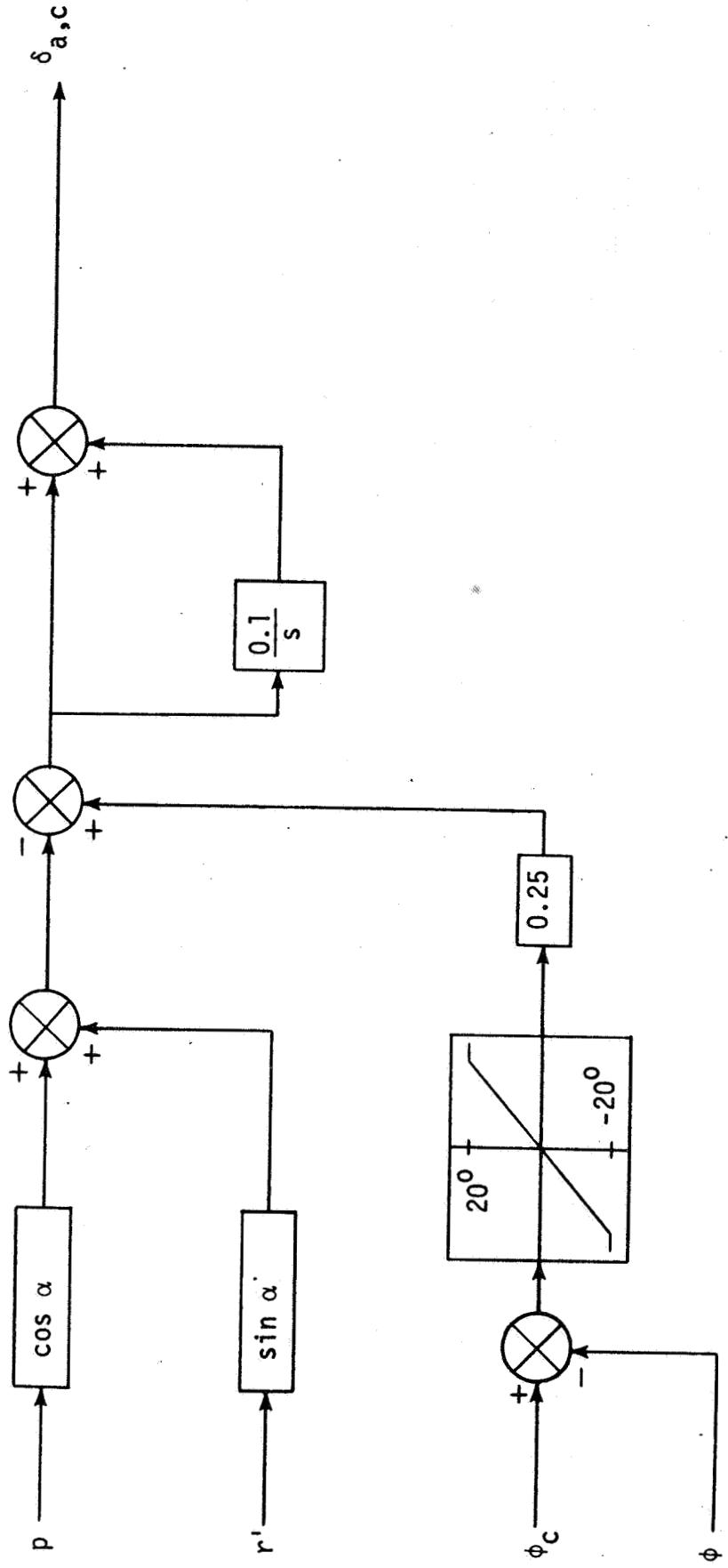
(a) $\alpha \leqq 18^{\circ}$ and $M \leqq 5$.

Figure B3.- Aileron-command block diagram for $\bar{q} < 96$ Pa (2 psf), $\delta_{a,c} = 0$.
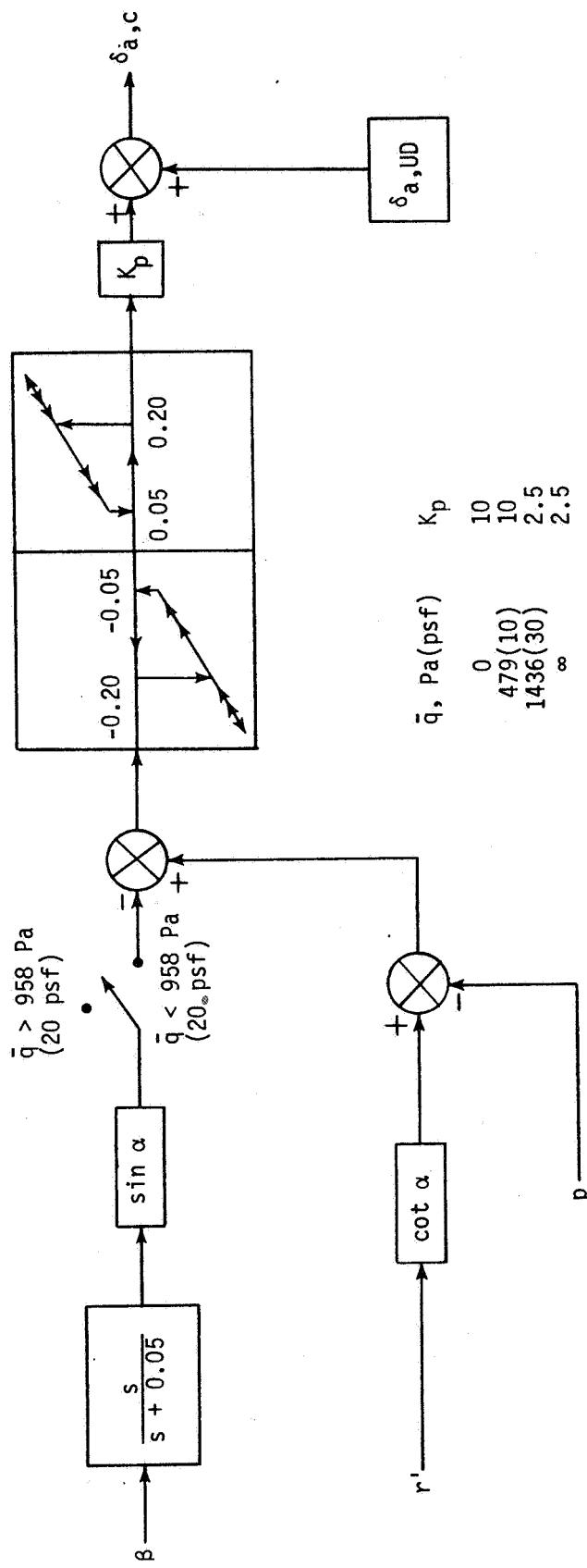
(b) $\alpha > 18^\circ$ or M > 5. For $\bar{q} < 96$ Pa (2 psf), $\delta_{a,c} = 0$; K_p is linearly varied between indicated points.

Figure B3.- Concluded.
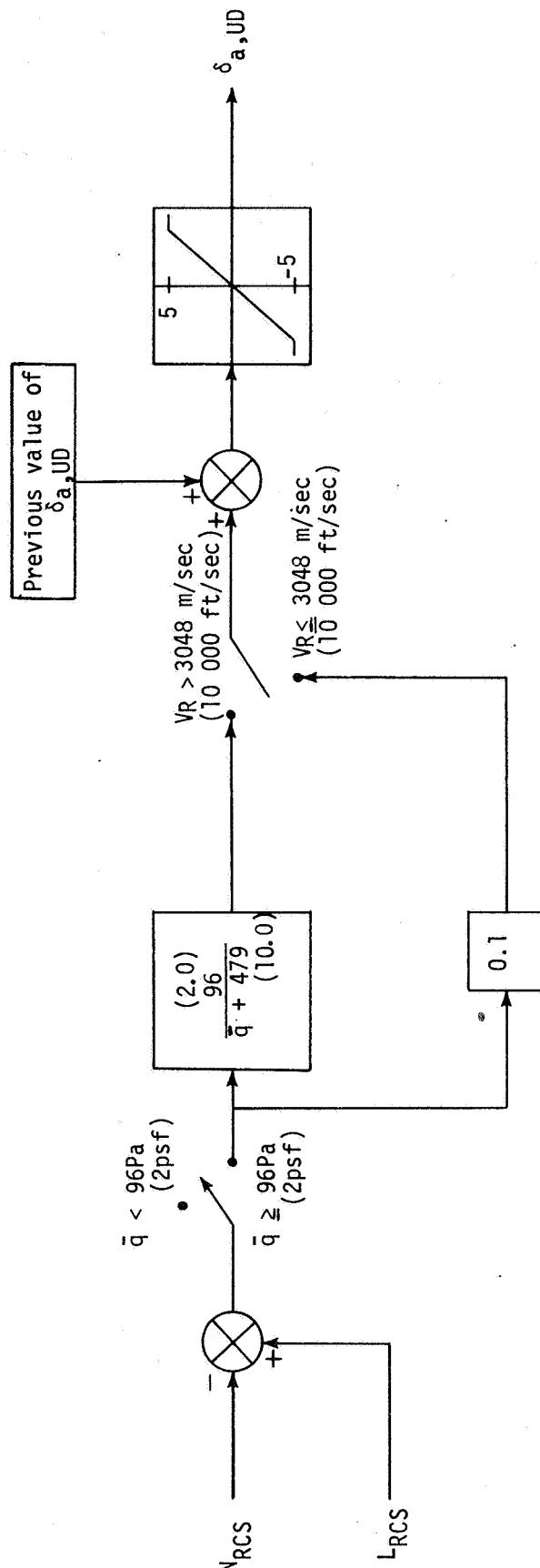
Figure B4.- Up-down counter block diagram. N_{RCS} indicates number of yaw jets that came on (+ for positive jet, − for negative jet). L_{RCS} indicates number of roll jets that came on (+ for positive jet, − for negative jet).
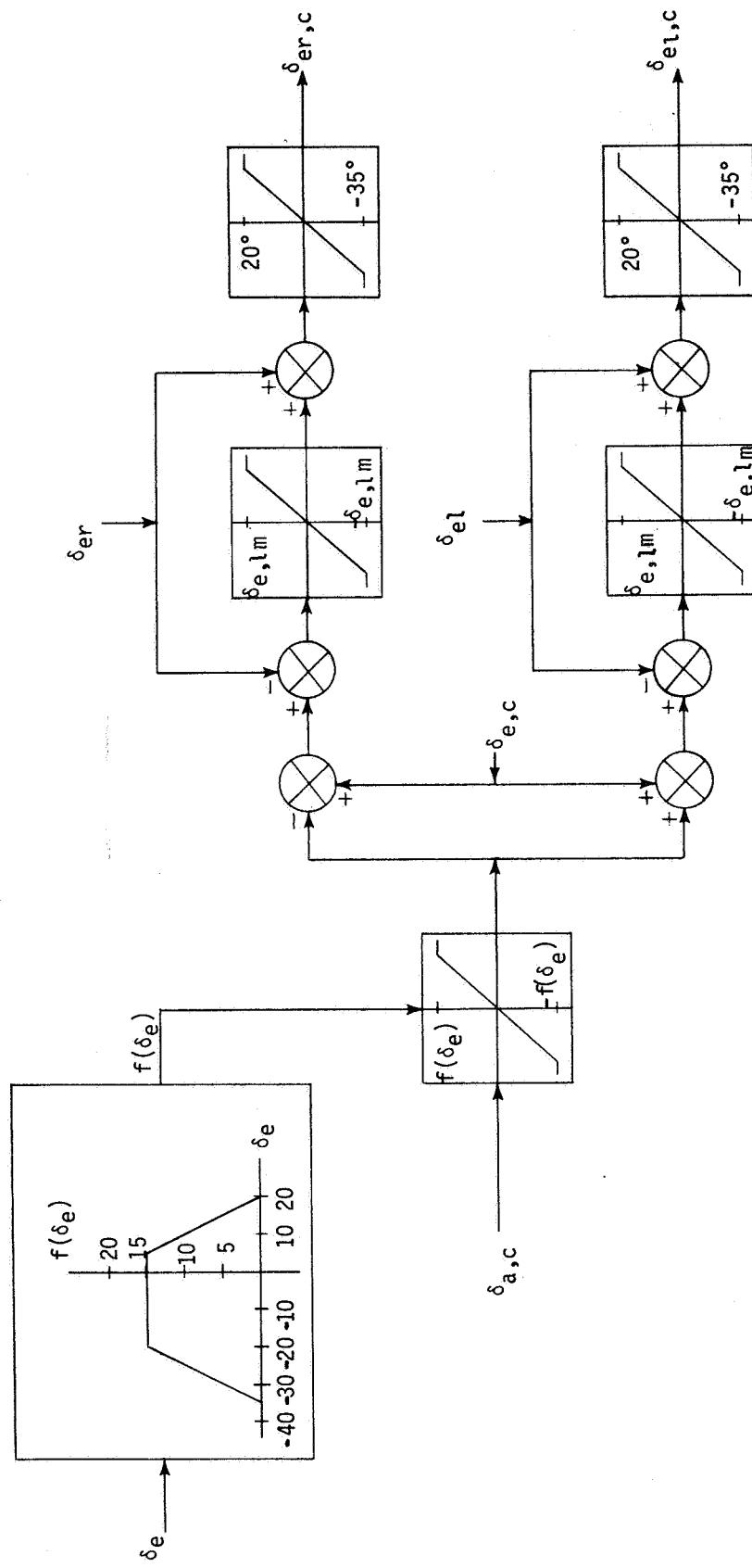
Figure B5.- Right and left elevon panel commands.
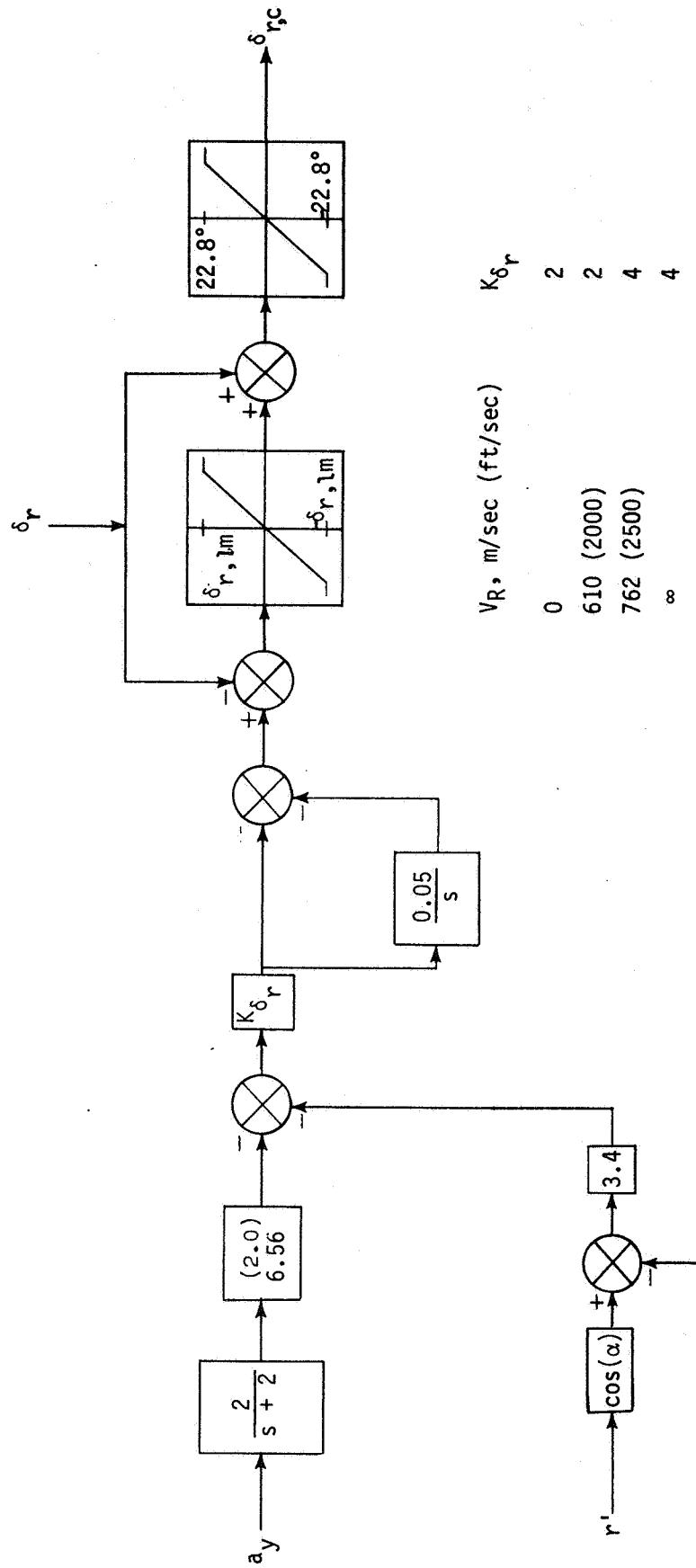
Figure B6.- Rudder command block diagram. For $\alpha > 18^{\circ}$ or $M > 5$, $\delta_{r,c} = 0$. K_{δ_r} is linearly varied between indicated points.

V_R, m/sec (ft/sec)	K_{δ_r}
0	2
610 (2000)	2
762 (2500)	4
∞	4
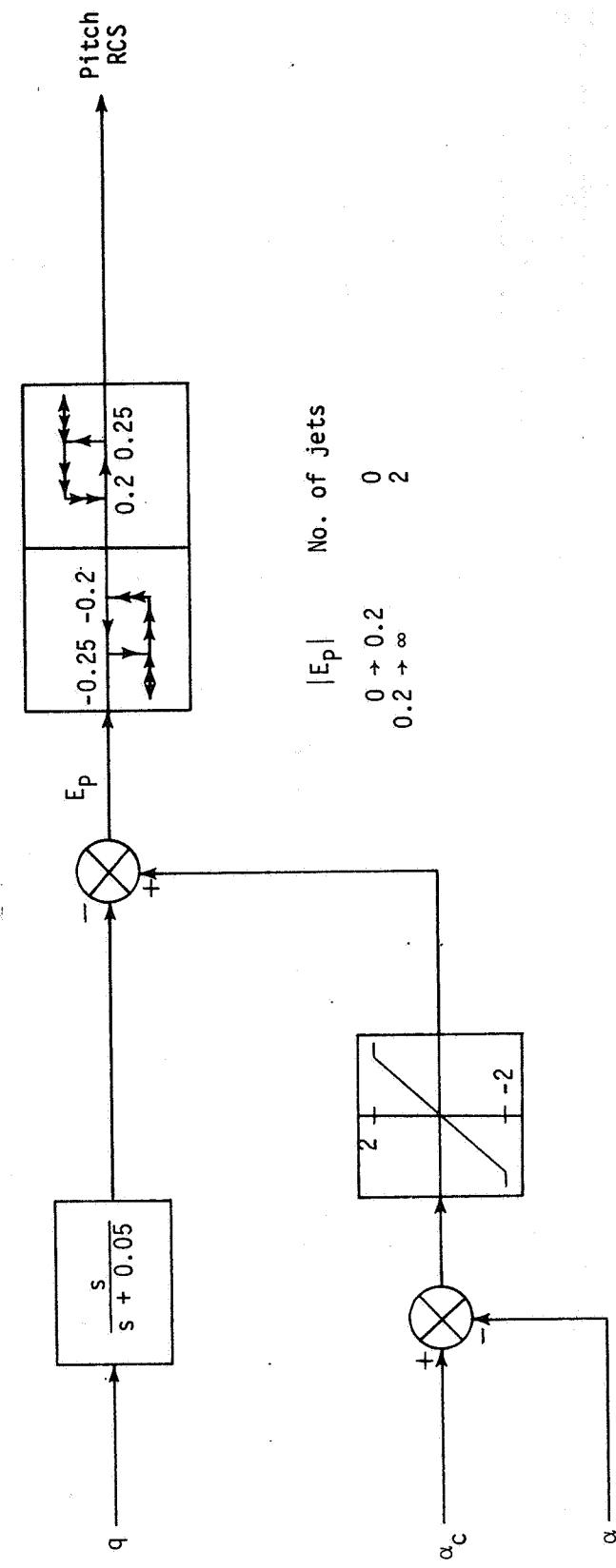
Figure B7.- Pitch RCS error-signal block diagram.
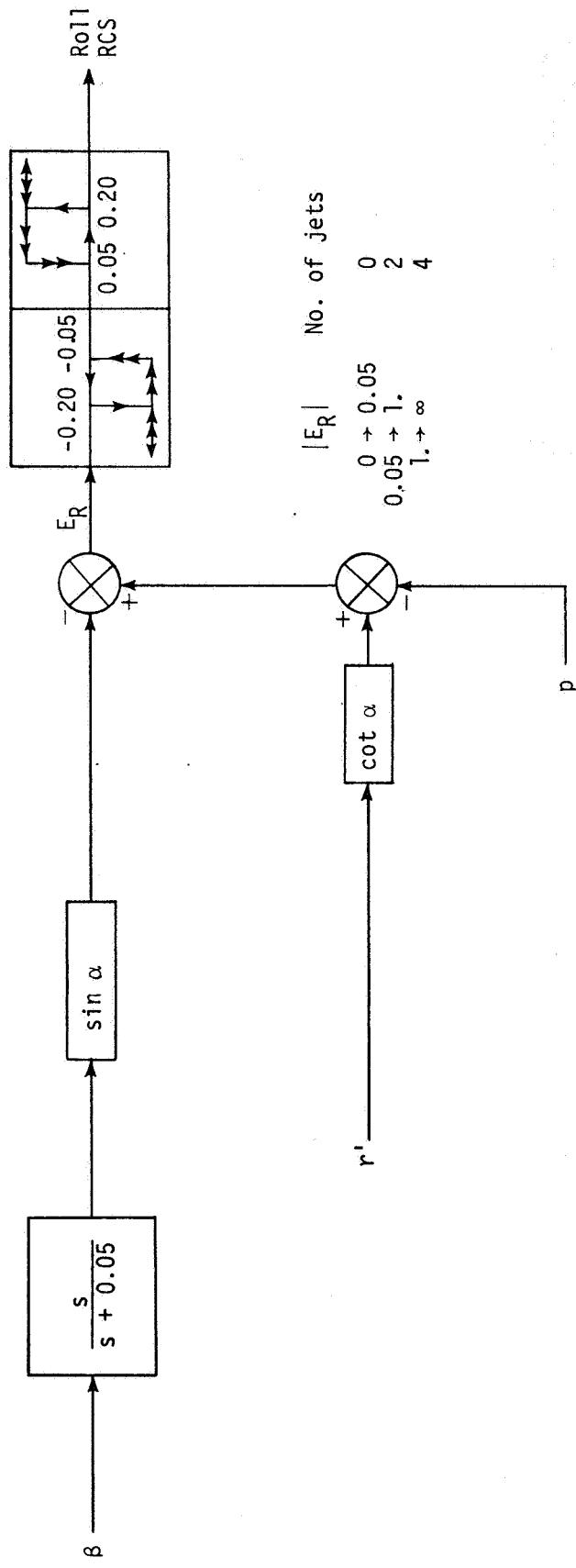
Figure B8.- Roll RCS error-signal block diagram.
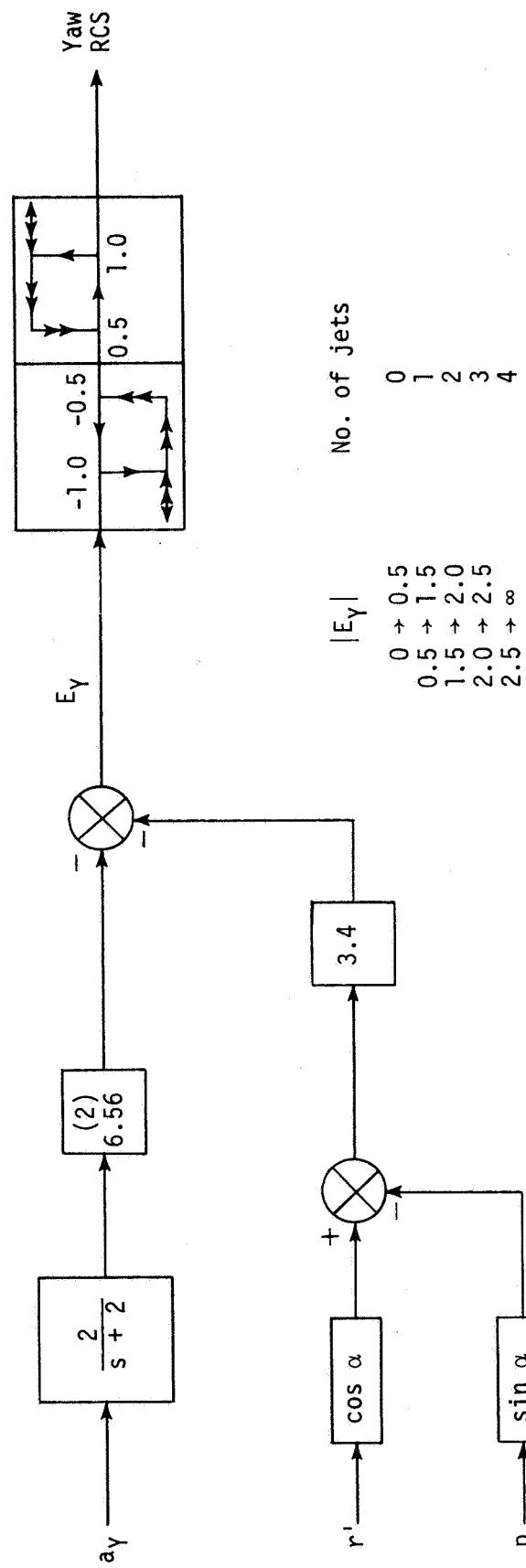
(a) $\alpha \leqq 18^\circ$ and $M \leqq 5$.

Figure B9.- Yaw RCS error-signal block diagram.
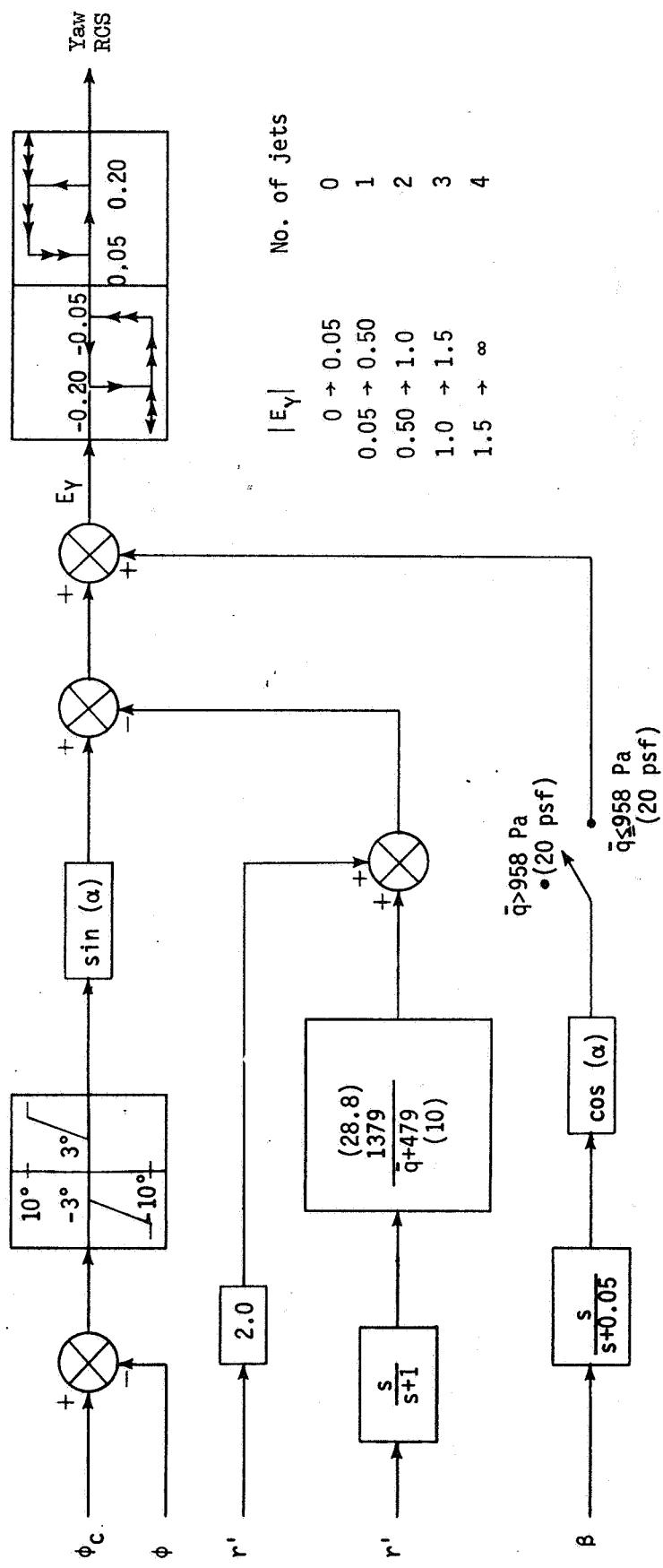
(b) $\alpha > 18°$ or M > 5. For $\bar{q} < 958$ Pa (20 psf), the number of jets is limited to 2.

Figure B9.- Concluded.
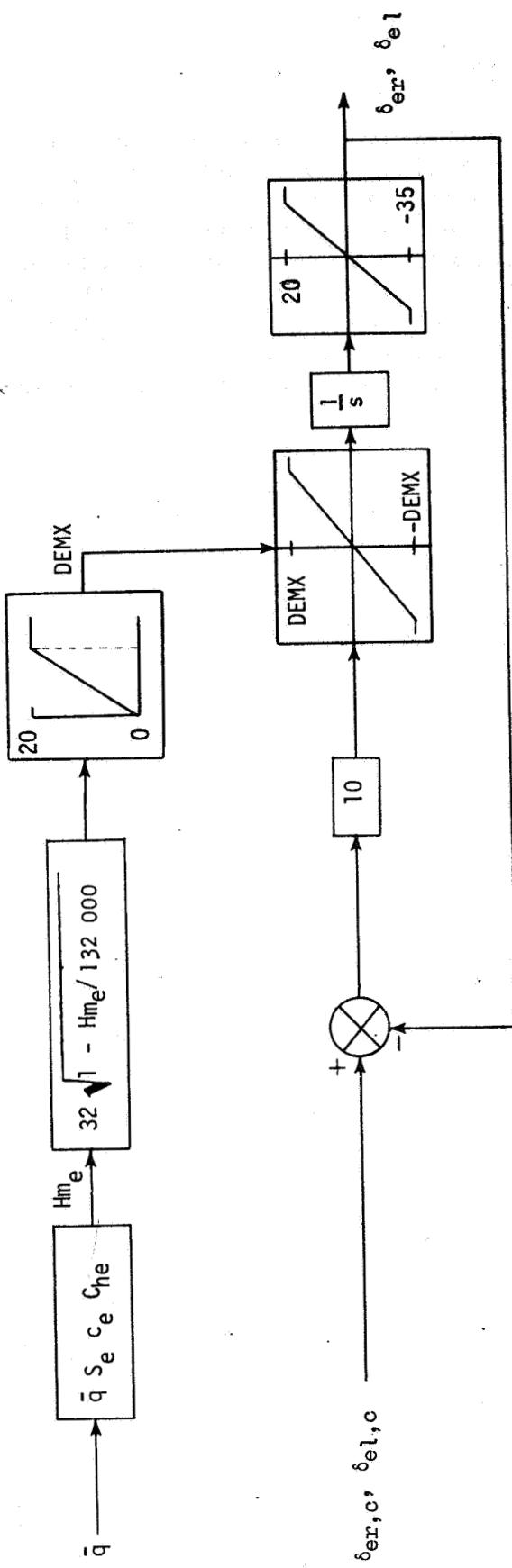
(a) Elevon.
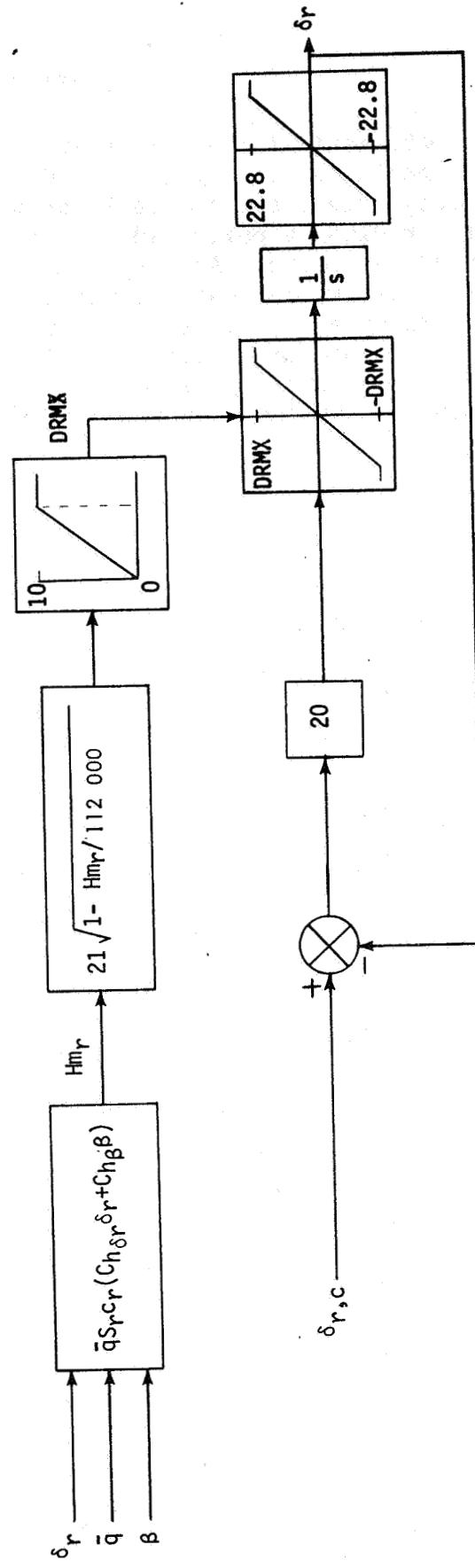
(b) Rudder.

Figure B10.- Actuator block diagrams.

APPENDIX C

EQUATIONS OF MOTION

The mathematical model is that of a nonlinear six-degree-of-freedom rigid body with nonlinear aerodynamics. Forces and moments are calculated in the vehicle body axis system. The Earth model is an oblate spheroid defined by an equatorial radius, a polar radius, rotational rate, and a gravitational model with second harmonics. Axis systems used in the mathematical model include body, inertial, relative geocentric, and inertial geocentric systems. The latitude and longitude rates are computed from relative geocentric velocities. Wind components can be programmatically varied or deleted. Figure C1 is a detailed block diagram showing state variable flow for the equations of motion and the independent variables for function generation.

Symbols

$\dot{A}$ longitudinal rate, rad/sec

a_x, a_y, a_z accelerations along body axes, m/sec^2 (ft/sec^2)

b wing span, m (ft)

C' temperature term for $\bar{v}$ equation

C_{AF} axial-force coefficient

C_D drag coefficient due to vehicle with undeflected control surfaces

C_{DT} total drag coefficient

C_{DV} increment of drag coefficient due to viscous interaction

C_H hinge-moment coefficient

C_L lift coefficient due to vehicle with undeflected control surfaces

C_{LT} total lift coefficient

C_{LV} increment of lift coefficient due to viscous interaction

C_m pitching-moment coefficient due to vehicle with undeflected controls

C_{mV} increment of pitching-moment coefficient due to viscous interaction

C_n yawing-moment coefficient due to vehicle with undeflected controls

C_Y side-force coefficient

$\bar{c}$ mean aerodynamic chord, m (ft)

64

APPENDIX C

c_p specific heat for air, J/kg-K (Btu/lbm-°R)

c_s speed of sound, m/sec (ft/sec)

DC_{AF} increment in axial force due to viscous effects

EA viscous axial exponent

F term used for iterative process for viscous calculations, K (°R)

F_p term used for iterative process for viscous calculations

F_X, F_Y, F_Z forces along body axes, N (lbf)

G_X, G_Y, G_Z gravitational components along inertial axes, m/sec^2 (ft/sec^2)

H_o enthalpy, J/kg (Btu/lbm)

h altitude, m (ft)

I_{XX}, I_{YY}, I_{ZZ} moments of inertia about body axes, kg-m^2 (slug-ft^2)

I_{XZ}, I_{YZ}, I_{XY} products of inertia about body axes, kg-m^2 (slug-ft^2)

J_2 second gravitational harmonic

l_b body reference length, m (ft)

l_i, m_i, n_i direction cosines, i = 1, 2, 3

M Mach number

M_X, M_Y, M_Z rolling, pitching, and yawing moments about body axes, N-m (lbf-ft)

m mass of vehicle, kg (lbm)

m_X, m_Y, m_Z moments about body axes due to RCS thrusters, N-m (lbf-ft)

N_{Re} Reynolds number

P, Q, R relative roll, pitch, and yaw rates about body axes, rad/sec

p, q, r inertial roll, pitch, and yaw rates about body axes, rad/sec

$\dot{Q}$ stagnation heating rate, J/m^2-sec (Btu/ft^2-sec)

$\bar{q}$ dynamic pressure, Pa (psf)

REH distance from center of Earth to vehicle, m (ft)

R_E equatorial radius of Earth, m (ft)

R_P polar radius of Earth, m (ft)

R_R ratio of R_E/REH

R_S radius of oblate Earth, m (ft)

S wing area, m^2 (ft^2)

T_{amb} ambient temperature, K (^{O}R)

T_w,T_{w1} iterative term used in viscous calculations, K (^{O}R)

T_X,T_Y,T_Z thrust components along body axes, N (lbf)

t time, sec

u_b,v_b,w_b atmospheric velocities along body axes, m/sec (ft/sec)

V atmospheric relative velocity, m/sec (ft/sec)

V_{INERT} inertial velocity, m/sec (ft/sec)

V_R Earth relative velocity, m/sec (ft/sec)

$V_{WX,RG},V_{WY,RG},V_{WZ,RG}$ atmospheric relative velocities along relative geocentric axes, m/sec (ft/sec)

V_X,V_Y,V_Z velocities along inertial axes, m/sec (ft/sec)

$V_{X,IG},V_{Y,IG},V_{Z,IG}$ velocities along inertial geocentric axes, m/sec (ft/sec)

$V_{X,RG},V_{Y,RG},V_{Z,RG}$ Earth relative velocities along relative geocentric axes, m/sec (ft/sec)

$\bar{v}$ hypersonic viscous parameter

$\bar{v}_c$ viscous effect ignored when hypersonic viscous parameter below $\bar{v}_c$

$W_{X,RG},W_{Y,RG},W_{Z,RG}$ wind components along relative geocentric axes, m/sec (ft/sec)

X,Y,Z components of position along inertial axes, m (ft)

X_b,Y_b,Z_b components of position along body axes, m (ft)

X_{IG},Y_{IG},Z_{IG} components of position along inertial geocentric axes, m (ft)

X_{off},Y_{off},Z_{off} center-of-gravity offsets along body axis such that X_{off} is positive forward, Y_{off} is positive right, and Z_{off} is positive downward, m (ft)

X_{RG},Y_{RG},Z_{RG} components of position along relative geocentric axes, m (ft)

Z_R	ratio of Z/REH
α	angle of attack, rad
β	angle of sideslip, rad
γ	flight-path angle, rad
δ_a	aileron deflection, $(\delta_{el} - \delta_{er})/2$, deg
δ_{BF}	body-flap deflection (positive downward), deg
δ_e	total elevon deflection, $(\delta_{el} + \delta_{er})/2$, deg
δ_{el}	elevon deflection, left panel (positive downward), deg
δ_{er}	elevon deflection, right panel (positive downward), deg
δ_r	rudder deflection (positive trailing edge left), deg
δ_{SB}	speed-brake deflection, deg
ϵ	emissivity of material
θ_c	longitude, rad
θ_I	inertial longitude, rad
λ	geocentric latitude, rad
μ	gravitational constant, m^3/sec^2 (ft^3/sec^2)
μ_A	viscosity, kg/m-sec (slugs/ft-sec)
ρ	atmospheric density, kg/m^3 ($slugs/ft^3$)
σ	Stephan-Boltzmann constant, $J/sec\text{-}m^2\text{-}K^4$ ($Btu/sec\text{-}ft^2\text{-}{}^\circ R^4$)
ϕ, θ, ψ	roll, pitch, and yaw angle about body axes, deg
ϕ_c	commanded roll angle, rad
ψ_e	heading angle, rad
ω	Earth rotational rate, rad/sec

Aerodynamic parameters:

$C_{D,\delta BF}$	increment of C_D due to body flap
$C_{D,\delta e}$	increment of C_D due to elevon

$C_{D,\delta SB}$ increment of C_D due to speed brakes

$C_{L,\delta BF}$ increment of C_L due to body flap

$C_{L,\delta e}$ increment of C_L due to elevon

$C_{L,\delta SB}$ increment of C_L due to speed brakes

$$C_{l_p} = \frac{\partial C_l}{\partial(pb/2V)}, \text{ rad}^{-1}$$

$$C_{l_r} = \frac{\partial C_l}{\partial(rb/2V)}, \text{ rad}^{-1}$$

$$C_{l_\beta} = \frac{\partial C_l}{\partial\beta}, \text{ deg}^{-1}$$

$C_{l_\beta,\delta SB}$ increment of C_{l_β} due to δ_{SB}, deg^{-1}

$$C_{l_{\delta a}} = \frac{\partial C_l}{\partial\delta_a}, \text{ deg}^{-1}$$

$$C_{l_{\delta r}} = \frac{\partial C_l}{\partial\delta_r}, \text{ deg}^{-1}$$

$$C_{m_q} = \frac{\partial C_m}{\partial(q\bar{c}/2V)}, \text{ rad}^{-1}$$

$C_{m,\delta BF}$ increment of C_m due to δ_{BF}

$C_{m,\delta e}$ increment of C_m due to δ_e

$C_{m,\delta SB}$ increment of C_m due to δ_{SB}

$$C_{n_p} = \frac{\partial C_n}{\partial(pb/2V)}, \text{ rad}^{-1}$$

$$C_{n_r} = \frac{\partial C_n}{\partial(rb/2V)}, \text{ rad}^{-1}$$

$$C_{n_\beta} = \frac{\partial C_n}{\partial\beta}, \text{ deg}^{-1}$$

68

$C_{n_\beta, \delta SB}$ increment of C_{n_β} due to speed brakes, $\deg^{-1}$

$C_{n_{\delta a}} = \dfrac{\partial C_n}{\partial \delta_a}$, $\deg^{-1}$

$C_{n_{\delta r}} = \dfrac{\partial C_n}{\partial \delta_r}$, $\deg^{-1}$

$C_{Y_\beta} = \dfrac{\partial C_Y}{\partial \beta}$, $\deg^{-1}$

$C_{Y_\beta, \delta SB}$ increment of C_{Y_β} due to speed brakes, $\deg^{-1}$

$C_{Y_{\delta a}} = \dfrac{\partial C_Y}{\partial \delta_a}$, $\deg^{-1}$

$C_{Y_{\delta r}} = \dfrac{\partial C_Y}{\partial \delta_r}$, $\deg^{-1}$

    A dot over a symbol indicates a derivative with respect to time t. A G before a stability or control derivative represents a multiplying factor for perturbation studies of that derivative.

Total Lift and Drag Coefficients

$$C_{LT} = GC_L C_L + C_{LV} + GC_{L,\delta e} C_{L,\delta e} + GC_{L,\delta SB} C_{L,\delta SB} + GC_{L,\delta BF} C_{L,\delta BF}$$

$$C_{DT} = GC_D C_D + C_{DV} + GC_{D,\delta e} C_{D,\delta e} + GC_{D,\delta SB} C_{D,\delta SB} + GC_{D,\delta BF} C_{D,\delta BF}$$

External Force Equations (Body)

$$a_x = \bar{q}S(C_{LT} \sin \alpha - C_{DT} \cos \alpha)\big/m + T_X/m$$

$$a_y = \bar{q}S\left[\left(GC_{Y_\beta} C_{Y_\beta} + GC_{Y_\beta, \delta SB} C_{Y_\beta, \delta SB}\right)\beta + GC_{Y_{\delta r}} C_{Y_{\delta r}} \delta r + GC_{Y_{\delta a}} C_{Y_{\delta a}} \delta a\right]\Big/m + T_Y/m$$

$$a_z = -\bar{q}S(C_{LT} \cos \alpha + C_{DT} \sin \alpha)\big/m + T_Z/m$$

$$\left.\begin{array}{l} F_X = ma_x - T_X \\ F_Y = ma_y - T_Y \\ F_Z = ma_z - T_Z \end{array}\right\} \quad T_X = T_Y = T_Z = 0 \text{ for current simulation}$$

Moment Equations (Body)

$$M_X = \bar{q}Sb\left[\frac{b}{2V}\left(GC_{l_p}C_{l_p}p + GC_{l_r}C_{l_r}r\right) + \left(GC_{l\beta}C_{l\beta} + GC_{l\beta,\delta SB}C_{l\beta,\delta SB}\right)\beta + GC_{l\delta a}C_{l\delta a}\delta_a\right.$$

$$\left. + GC_{l\delta r}C_{l\delta r}\delta_r\right] + m_X + Z_{off}F_Y - Y_{off}F_Z$$

$$M_Y = \bar{q}S\bar{c}\left(GC_m C_m + C_m V + \frac{\bar{c}}{2V}GC_{m_q}C_{m_q}q + GC_{m,\delta e}C_{m,\delta e} + GC_{m,\delta SB}C_{m,\delta SB}\right.$$

$$\left. + GC_{m,\delta BF}C_{m,\delta BF}\right) + m_Y + X_{off}F_Z - Z_{off}F_X$$

$$M_Z = \bar{q}Sb\left[\left(GC_{n\beta}C_{n\beta} + GC_{n\beta,\delta SB}C_{n\beta,\delta SB}\right)\beta + \frac{b}{2V}\left(GC_{n_p}C_{n_p}p + GC_{n_r}C_{n_r}r\right) + GC_{n\delta r}C_{n\delta r}\delta_r\right.$$

$$\left. + GC_{n\delta a}C_{n\delta a}\delta_a\right] + m_Z + Y_{off}F_X - X_{off}F_Y$$

Translational Equations (Relative Geocentric)

$$\dot{V}_{X,RG} = l_1 a_x + m_1 a_y + n_1 a_z$$

$$\dot{V}_{Y,RG} = l_2 a_x + m_2 a_y + n_2 a_z$$

$$\dot{V}_{Z,RG} = l_3 a_x + m_3 a_y + n_3 a_z$$

Translational Equations (Inertial Body)

$$\dot{V}_X = -\sin\lambda\,\cos\theta_I\dot{V}_{X,RG} - \sin\theta_I\dot{V}_{Y,RG} - \cos\lambda\,\cos\theta_I\dot{V}_{Z,RG} + G_X$$

$$\dot{V}_Y = -\sin\lambda\,\sin\theta_I\dot{V}_{X,RG} + \cos\theta_I\dot{V}_{Y,RG} - \cos\lambda\,\sin\theta_I\dot{V}_{Z,RG} + G_Y$$

$$\dot{V}_Z = \cos\lambda\dot{V}_{X,RG} - \sin\lambda\dot{V}_{Z,RG} + G_Z$$

$$\dot{X} = V_X$$

$$\dot{Y} = V_Y$$

$$\dot{Z} = V_Z$$

Rotational Equations (Inertial Body)

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} I \end{bmatrix}^{-1} \left\{ \begin{bmatrix} M_X \\ M_Y \\ M_Z \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} I \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \right\}$$

where

$$\begin{bmatrix} I \end{bmatrix} = \begin{bmatrix} I_{XX} & -I_{XY} & -I_{XZ} \\ -I_{XY} & I_{YY} & -I_{YZ} \\ -I_{XZ} & -I_{YZ} & I_{ZZ} \end{bmatrix}$$

Rotational Equations (Euler Angle Rates)

$$P = p - l_1(\dot{A} + \omega) \cos \lambda - l_2\dot{\lambda} + l_3(\dot{\theta}_c + \omega) \sin \lambda$$

$$Q = q - m_1(\dot{A} + \omega) \cos \lambda - m_2\dot{\lambda} + m_3(\dot{\theta}_c + \omega) \sin \lambda$$

$$R = r - n_1(\dot{A} + \omega) \cos \lambda - n_2\dot{\lambda} + n_3(\dot{\theta}_c + \omega) \sin \lambda$$

$$\dot{\phi} = P + (Q \sin \phi + R \cos \phi)/\tan \theta$$

$$\dot{\theta} = Q \cos \phi - R \sin \phi$$

$$\dot{\psi} = (Q \sin \phi - R \cos \phi)/\cos \theta$$

Velocity Equations (Inertial Geocentric)

$$V_{X,IG} = -V_X \sin \lambda \cos \theta_I - V_Y \sin \lambda \sin \theta_I + V_Z \cos \lambda$$

$$V_{Y,IG} = -V_X \sin \theta_I + V_Y \cos \theta_I$$

$$V_{Z,IG} = -V_X \cos \lambda \cos \theta_I - V_Y \cos \lambda \sin \theta_I - V_Z \sin \lambda$$

Velocity Equations (Relative Geocentric)

$$V_{X,RG} = V_{X,IG}$$

$$V_{Y,RG} = V_{Y,IG} - REH\omega \cos \lambda$$

$$V_{Z,RG} = V_{Z,IG}$$

Velocity Equations (Atmospheric Relative)

$$V_{WX,RG} = V_{X,RG} - W_{X,RG}$$

$$V_{WY,RG} = V_{Y,RG} - W_{Y,RG}$$

$$V_{WZ,RG} = V_{Z,RG} - W_{Z,RG}$$

Velocity Equations (Body)

$$u_b = V_{WX,RG}l_1 + V_{WY,RG}l_2 + V_{WZ,RG}l_3$$

$$v_b = V_{WX,RG}m_1 + V_{WY,RG}m_2 + V_{WZ,RG}m_3$$

$$w_b = V_{WX,RG}n_1 + V_{WY,RG}n_2 + V_{WZ,RG}n_3$$

Direction Cosines

$$l_1 = \cos \psi \cos \theta$$

$$l_2 = \sin \psi \cos \theta$$

$$l_3 = -\sin \theta$$

$$m_1 = \cos \psi \sin \theta \sin \phi - \sin \psi \cos \phi$$

$$m_2 = \sin \psi \sin \theta \sin \phi + \cos \psi \cos \phi$$

$$m_3 = \cos \theta \sin \phi$$

$$n_1 = \cos \psi \sin \theta \cos \phi + \sin \psi \sin \phi$$

$$n_2 = \sin \psi \sin \theta \cos \phi - \cos \psi \sin \phi$$

$$n_3 = \cos \theta \cos \phi$$

Auxiliary Equations

$$V = \sqrt{u_b^2 + v_b^2 + w_b^2}$$

$$V_R = \sqrt{V_{X,RG}^2 + V_{Y,RG}^2 + V_{Z,RG}^2}$$

$$\alpha = \tan^{-1} (w_b/u_b)$$

$$\beta = \sin^{-1} (v_b/V)$$

$$\gamma = \tan^{-1}\left(-V_{Z,RG} \middle/ \sqrt{V_{X,RG}^2 + V_{Y,RG}^2}\right)$$

$$\psi_e = \tan^{-1}(V_{Y,RG}/V_{X,RG})$$

$$V_{INERT} = \sqrt{V_X^2 + V_Y^2 + V_Z^2}$$

$$\bar{q} = 0.5\rho V^2$$

$$N_{Re} = \rho V l_b / \mu_A$$

$$M = V/c_s$$

$$\dot{h} = -V_{Z,RG}$$

$$\theta_I = \tan^{-1}(Y/X)$$

$$\theta_c = \theta_I - \omega t$$

$$\dot{A} = V_{Y,RG}/REH \cos \lambda$$

Earth Model and Gravitational Equations

The Earth model is shown in the following sketch:
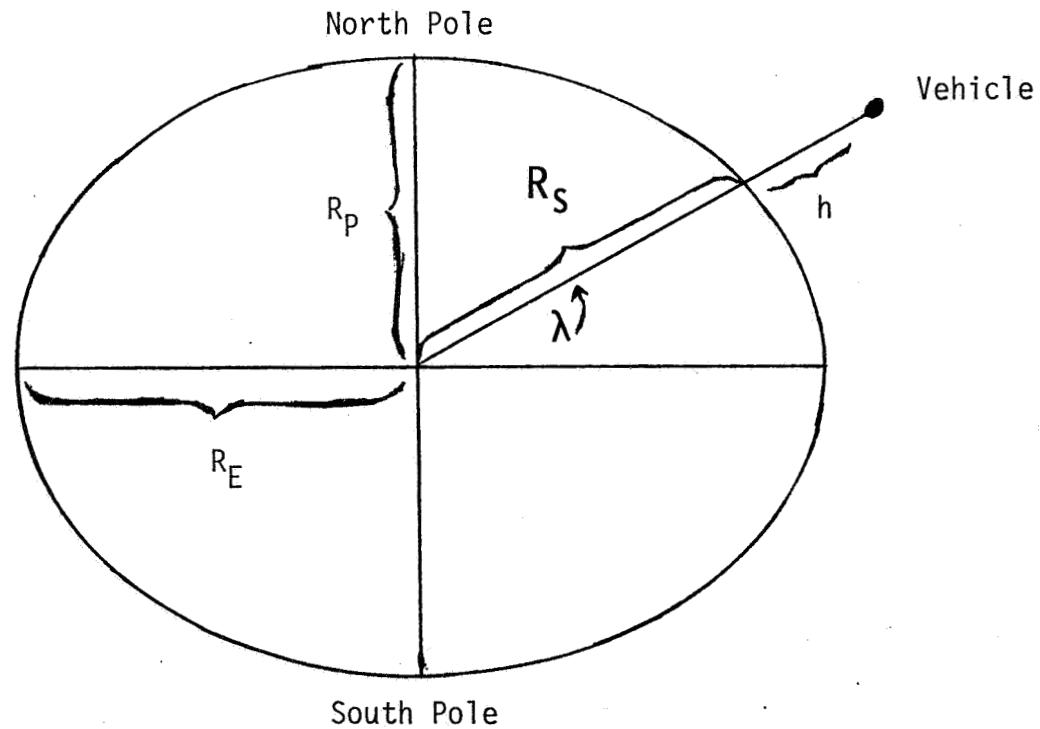
$$REH = \sqrt{X^2 + Y^2 + Z^2}$$

$$\lambda = \sin^{-1}(Z/REH)$$

$$R_S = R_E\left[1 + (k - 1)\sin^2\lambda\right]^{-1/2} \qquad \left(k = (R_E/R_P)^2\right)$$

$$h = REH - R_S$$

$$Z_R = Z/REH$$

$$R_R = R_E/REH$$

The gravitational equations are as follows:

$$G_X = -\mu(X/REH^3)\left[1 + JR_R^2(1 - 5Z_R^2)\right]$$

$$G_Y = -\mu(Y/REH^3)\left[1 + JR_R^2(1 - 5Z_R^2)\right]$$

$$G_Z = -\mu(Z/REH^3)\left[1 + JR_R^2(3 - 5Z_R^2)\right]$$

where

$$J = 1.5(J_2)$$

$$J_2 = 1.082630 \times 10^{-3}$$

$$\mu = 3.986009 \times 10^{14}$$

Viscous Equations

$$H_O = c_p T_{amb} + V^2/2$$

$$\dot{Q} = 8.8484 \times 10^6 \sqrt{\rho(V/3048)^{3.07}}$$

$$F = -T_w + C1\left\{\dot{Q}\left[1 - c_p(T_w/H_o)\right]\right\}^{0.25}$$

$$F_p = -1 - 0.25C1\left[c_p(\dot{Q}/H_o)\right]\bigg/\left\{\dot{Q}\left[1 - c_p(T_w/H_o)\right]\right\}^{0.75}$$

$$T_{w1} = (T_w - F/F_p) \quad \text{(Starting with } T_w = 556 \text{ K, iterate on } T_w \text{ until}$$
$$|T_{w1} - T_w| \leq 0.56 \text{ K)}$$

$$T_w = T_{w1}$$

$$C' = \sqrt{T_w/T_{amb}}(T_{amb} + 110.4)/(T_w + 110.4)$$

$$\bar{v} = M\sqrt{C'/N_{Re}}$$

$$\bar{v}_R = \bar{v}/\bar{v}_c$$

$$C_{AF} = GC_D C_D \cos \alpha - GC_L C_L \sin \alpha$$

$$DC_{AF} = \left[(\bar{v}_R)^{EA} - 1\right] C_{AF}$$

$$C_{mV} = 0.11583825 DC_{AF}$$

$$C_{LV} = -DC_{AF} \sin \alpha$$

$$C_{DV} = DC_{AF} \cos \alpha$$

C_{mV}, C_{LV}, and C_{DV} are set equal to zero when the viscous calculations are not used

ere

$$C1 = (1/\varepsilon\sigma)^{0.25} = 68.52 \ (\text{sec-m}^2\text{-K}^4/\text{J})^{0.25} \ (1272.99 \ (\text{sec-ft}^2\text{-}^{\circ}\text{R}^4/\text{Btu})^{0.25})$$

$$\varepsilon = 0.8$$

$$c_p = 1004.16 \ \text{J/kg-K} \ (0.24 \ \text{Btu/lbm-}^{\circ}\text{R})$$

$$EA = 0.636$$

$$\bar{v}_c = 0.007$$
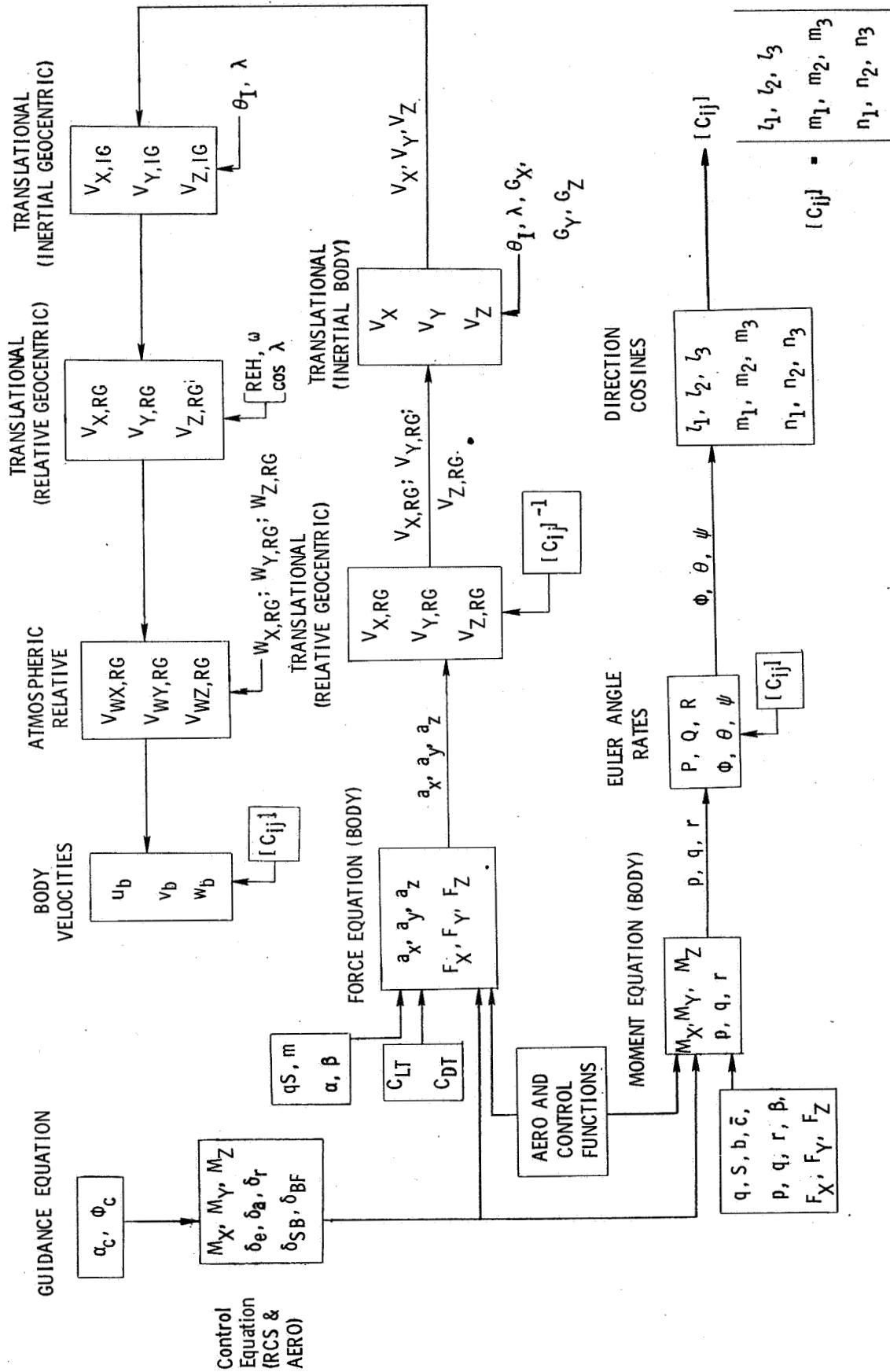
Figure C1.- Flow of variables for equations of motion.

INITIAL CONDITION TRANSFORMATION

$V_{X,RG}$ $V_{Y,RG}$ $V_{Z,RG}$

u_b v_b w_b

C_{ij}

θ Φ ψ

γ α β Φ ψ_e

ATMOSPHERIC

c_s, ρ T_{amb}, μ_A

h

GRAVITATIONAL

G_X, G_Y, G_Z

X, Y, Z R_E, R_P λ, μ

AUXILIARY EQUATIONS

X, Y, Z ρ, μ_A l_b, c_s

V, V_R, α, β

γ, ψ_e, V_{INERT}

q, R, h, REH

θ_I, λ, θ_c

u_b, v_b, w_b $V_{X,RG}$ $V_{Y,RG}$ $V_{Z,RG}$ $V_{X,IG}$ ω

VISCOUS

C_{LV} C_{DV} C_{mV}

H_o Q ρ V T_{amb} M N_{Re} α C_D C_L

AERODYNAMIC

C_L, C_D, C_m

C_{l_p}, C_{l_r}, C_{l_β}

C_{n_p}, C_{n_r}, C_{n_β}

C_{m_q}, C_{Y_β}

α M

CONTROL AND HINGE MOMENT

C_{L,δ_e}; $C_{L,\delta SB}$; $C_{L,\delta BF}$; C_{D,δ_e};

$C_{D,\delta SB}$; $C_{D,\delta BF}$; $C_{l_{\delta a}}$; $C_{l_{\delta r}}$; $C_{Y_{\delta a}}$;

$C_{Y_{\delta r}}$; $C_{m,\delta e}$; $C_{m,\delta SB}$; $C_{m,\delta BF}$;

$C_{n_{\delta a}}$; $C_{n_{\delta r}}$; C_{Y_β}, δSB; C_{n_β}, δSB;

$C_{l_B,\delta SB}$; $C_{H_{\delta e}}$; $C_{H_{\delta r}}$; C_{H_β}; $C_{H_{\delta SB}}$; $C_{H_{\delta BF}}$

α M δ_e δ_{SB} δ_{BF}

Figure C1.- Concluded.

CONVOLUTION INTEGRATION TECHNIQUE

Lawrence E. Barker and Lawrence F. Rowell
Langley Research Center

The integration technique described is for the solution of linear differential equations with constant coefficients. The primary advantages of this technique are (1) the stability of the solution, (2) the high degree of accuracy for large system roots, (3) the variety of difference equations possible from this method of implementation, and (4) the ease in programing solutions to transfer functions when using the available subroutines. The symbols which follow have been used in the description of the integration procedures:

A $n \times n$ matrix of system state coefficients

B $n \times r$ matrix of forcing function coefficients

$\dot{b}$ nth-order vector of forcing function coefficients

C   constant gain matrix

D,E   coefficients of states when coefficient of $\ddot{x}$ is one

G   constant gain matrix

h   integration interval size, sec

K   constant gain matrix

n   order of highest order derivative in system

P(h)   constant matrix e^{Ah}

$P_{11}, P_{12}, P_{21}, P_{22}$ elements of matrix P(h) for a second-order system

Q(h) 3×3 matrix of coefficients for three-term Taylor series

q_1, q_2, q_3 elements of $\bar{q}_i$ for a first-order system

$\left. \begin{array}{l} q_{11}, q_{12}, q_{13}, \\ q_{21}, q_{22}, q_{23} \end{array} \right\}$ elements of $\bar{q}_i$ for a second-order system

$\bar{q}_i(h)$ vector values of Q(h), where i = 1, 2, 3

r   number of forcing functions into system

APPENDIX D

s   Laplace operator

t   time, sec

t_k discrete sample time, $t = kh$ where $k = 0, 1, 2, \ldots$

$u(t)$ scalar forcing function input to a transfer function

$\bar{u}(t)$ rth-order vector representing forcing functions

$\bar{u}^*(t_k)$ vector of three terms resulting from Taylor series

W   single element of matrix A for a first-order system

$x(t)$ scalar solution state for a first-order system

$\bar{x}(t)$ n-dimensional vector representing system states of a dynamic system

$y(t)$ scalar output that results from $\bar{y}(t)$ for single output transfer function

$\bar{y}(t)$ nth-order vector of multiple outputs of a transfer function

Z   Z-transform operator

τ dummy integration variable

     A dot over a symbol indicates the derivative of the quantity with respect to time t. A double dot represents the second derivative with respect to time t.

     The behavior of a linear system can be described by a set of first-order differential equations of the form

$$\dot{\bar{x}}(t) = A\bar{x}(t) + B\bar{u}(t) \tag{D1}$$

where $\bar{x}(t)$ is the nth-order vector describing the system states, $\bar{u}(t)$ is an rth-order vector representing the forcing functions, A is an $n \times n$ matrix, B is an $n \times r$ matrix, and $\dot{\bar{x}}(t)$ is the time derivative of $\bar{x}(t)$. The well-known solution (ref. 4) to equation (D1) is

$$\bar{x}(t) = e^{At}\bar{x}(0) + \int_0^t e^{A(t-\tau)}B\bar{u}(\tau)\, d\tau \tag{D2}$$

where e^{At} is called the transition matrix and τ is a dummy integration variable. For digital (discrete) systems, the sample time is represented by t_K and equation (D2) becomes

$$\bar{x}(t_K + h) = P(h)\bar{x}(t_K) + \int_0^h P(h - \tau)B\bar{u}(t_K + \tau)\,d\tau \qquad \text{(D3)}$$

where h is the integration step size and $P(h) = e^{Ah}$. Equation (D3) provides an exact solution; however, in a digital simulation, the value is estimated on the basis of a knowledge of $\bar{u}(t)$ at discrete sample times. The solution of equation (D3) can be obtained by several approaches. The convolution integration technique is a numerical-analytical method derived by approximating the system forcing function $\bar{u}(t)$ with a Taylor series as it appears under the convolution integral in equation (D3). This technique is a special case of the local linearization described in reference 5.

Since systems defined in transfer function form will include a forcing function in only the highest order derivative, $u(t)$ can be treated as a scalar input $u(t)$, and B becomes

$$\bar{b} = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ 0 \\ 1 \end{bmatrix}_{n \times 1}$$

Therefore, equation (D1) becomes

$$\dot{\bar{x}}(t) = A\bar{x}(t) + \bar{b}u(t) \qquad \text{(D4)}$$

and equation (D3) becomes

$$\bar{x}(t_K + h) = P(h)\bar{x}(t_K) + \int_0^h P(h - \tau)\bar{b}u(t_K + \tau)\,d\tau \qquad \text{(D5)}$$

By retaining three terms of the Taylor series, the resulting difference equation is

$$\bar{x}(t_K + h) = P(h)\bar{x}(t_K) + Q(h)\bar{u}^*(t_K) \qquad \text{(D6)}$$

where

$$P(h) = e^{Ah}$$
$$Q(h) = \begin{bmatrix} \bar{q}_1(h), \bar{q}_2(h), \bar{q}_3(h) \end{bmatrix}$$

and

$$\bar{u}^*(t_K) = \begin{bmatrix} u(t_K) \\ \dot{u}(t_K) \\ \ddot{u}(t_K) \end{bmatrix}$$

The vectors $\bar{q}_i$, where $i = 1, 2,$ and 3, are defined by

$$\bar{q}_i(h) = \frac{1}{(i-1)!} \int_0^h P(h - \tau)\tau^{i-1}\bar{b}\, d\tau$$

The change in the state space vector $\bar{x}(t)$ is the linear sum of $P(h)\bar{x}(t_K)$, $\bar{q}_1(h)u(t_K)$, $\bar{q}_2(h)\dot{u}(t_K)$, and $\bar{q}_3(h)\ddot{u}(t_K)$. The only approximation in equation (D6) (see ref. 6) is the representation of the forcing function between sample points. Equation (D6) can be obtained also as a specific case from a more general algorithm derived by the method of local linearization applied to a nonlinear system of equations (ref. 5).

The constants $P(h)$ and $Q(h)$ are functions of the system roots and the integration step size, and their analytic expressions have been programed into subroutines for first- and second-order systems. The method adopted for solving third- or higher-order systems is to break them down by partial fractions into first- and second-order systems and then to solve the resulting systems by parallel programing. The constants $P(h)$ and $Q(h)$ ensure correct placement of the continuous system roots, and their calculations need be performed only once before the integration is begun. A representation of the state variable formulation for a single input, multiple output system is as follows:
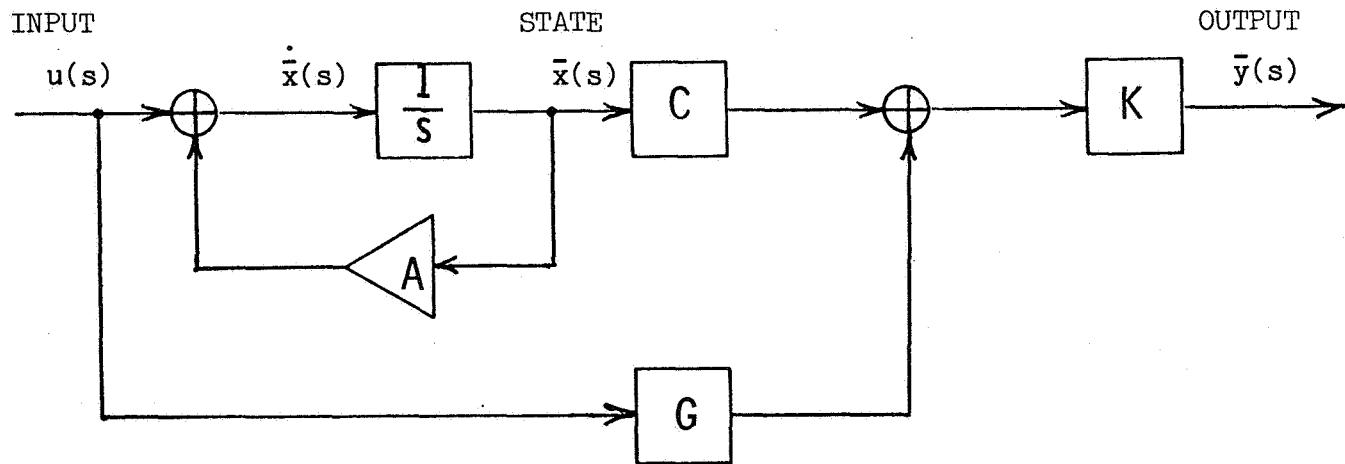
Given the system described by equation (D4), the output equation is

$$\bar{y}(t) = K\left[C\bar{x}(t) + Du(t)\right] \tag{D7}$$

where C, G, and K are gain matrices.

The programing concept described here differs from numerical integration
in that information on the forcing function is supplied to the integration
routine, and the resulting solution is used in an algebraic "output equation."
Numerical techniques require calculation of an algebraic derivative equation,
and the result is passed to the integration routine to be used with past infor-
mation in calculating the solution. When solving a transfer function, the
result of integrating by the convolution technique is to solve the corresponding
transfer function as if it had a numerator of 1. The output equation is used to
account for the "numerator dynamics" and thus complete the solution.

Table DI and table DII give a reference to the output equations for some
common first- and second-order transfer functions, respectively. The derivative
equation shown at the top of each table is just one of the possible output
equations.

Linear first-order systems are described by

$$\dot{x}(t) + Wx(t) = u(t)$$

where W is the negative of the system root, $u(t)$ is the forcing function, and
$x(t)$ is the solution state. The transfer function representation of the first-
order system and the convolution equations are as follows:
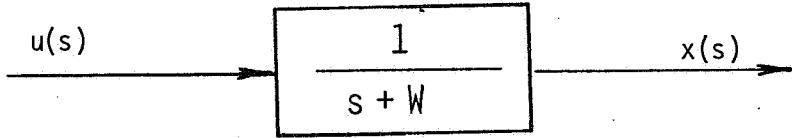
$$x_1(t_K + h) = P(h)x_1(t_K) + q_1 u(t_K) + q_2 \dot{u}(t_K) + q_3 \ddot{u}(t_K)$$

$$x(t_K + h) = u(t_K + h) - Wx_1(t_K + h)$$

where $P(h) = e^{-Wh}$

$$q_1 = (1 - P)/W$$

$$q_2 = (h - q_1)/W$$

$$q_3 = \left(\frac{1}{2} h^2 - q_2\right)\Big/ W$$

The derivative $\dot{x}(t)$ can be calculated if velocity is required as an output or
for limiting purposes.

Linear second-order systems are described by

$$\ddot{x}(t) + D\dot{x}(t) + Ex(t) = u(t)$$

where D and E are the coefficients of the states when the coefficient of $\ddot{x}(t)$ is one. The second-order convolution equations and the transfer function representation are as follows:
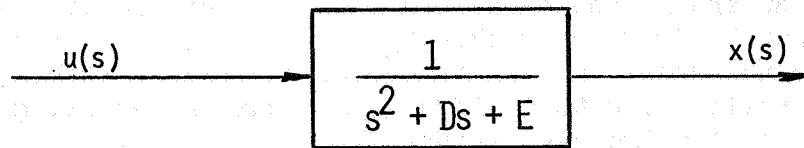
$$x_1(t_K + h) = p_{11}x_1(t_K) + p_{12}x_2(t_K) + q_{11}u(t_K) + q_{12}\dot{u}(t_K) + q_{13}\ddot{u}(t_K)$$

$$x_2(t_K + h) = p_{21}x_1(t_K) + p_{22}x_2(t_K) + q_{21}u(t_K) + q_{22}\dot{u}(t_K) + q_{23}\ddot{u}(t_K)$$

$$\ddot{x}(t_K + h) = u(t_K + h) - Dx_2(t_K + h) - Ex_1(t_K + h)$$

where p's and q's have equations that differ with the type of roots, that is, complex, real and equal, or real and unequal.

By varying the method of computing $u(t)$, $\dot{u}(t)$, and $\ddot{u}(t)$ (that is, there is no restriction to use the values at time t_K), many different algorithms can be developed for digitally representing a filter transfer function when the filter specification is in analog form. The choice should depend on the application, the method of computing, or the system to be simulated.

For example, to model the first-order system $W/(s + W)$, three of the many possible algorithms that can be developed are as follows:

Case (a)

$$x(t_K + h) = e^{-Wh}x(t_K) + (1 - e^{-Wh})u(t_K) \qquad (u = u(t_K); \quad \dot{u} = \ddot{u} = 0)$$

This difference equation corresponds to the Z-transform $(1 - e^{-Wh})/(Z - e^{-Wh})$.

Case (b)

$$x(t_K + h) = e^{-Wh}x(t_K) + (1 - e^{-Wh})u(t_K + h) \qquad (u = u(t_K + h); \quad \dot{u} = \ddot{u} = 0)$$

This equation corresponds to the Z-transform $(1 - e^{-Wh})Z/(Z - e^{-Wh})$.

Case (c)

$$x(t_K + h) = e^{-Wh}x(t_K) + (1 - e^{-Wh})\frac{u(t_K) + u(t_K + h)}{2}$$

$$\left(u = \frac{u(t_K) + u(t_K + h)}{2}; \quad \dot{u} = \dddot{u} = 0 \right)$$

This equation corresponds to the Z-transform $(1 - e^{-Wh})(Z + 1)/2(Z - e^{-Wh})$ where Z is the Z-transform operator.

For each of these cases, the roots of the continuous system are matched. The differences are in the reconstruction of the forcing function $u(t)$ which is treated separately from the stability of the process. In these examples, $\dot{u}$ and $\ddot{u}$ were not used; in many applications, $\dot{u}$ is available within the program or can be approximated with a difference equation resulting in many more possible algorithms and improved accuracy.

There are other methods for deriving algorithms for modeling the transfer function $W/(s + W)$. There are substitution methods for the Laplace operator s (Tustin, for example, which is the equivalent to case (c) above with a Padé approximation made for e^{-Wh}). Also, a numerical integration scheme such as rectangular or Euler could be used. These methods give an approximation on the placement of the continuous system roots, but for $hW \ll 1$, the results from all of these would be similar.

Equation (D6) has been programed for first- and second-order systems in subroutine form to make the use of the convolution technique quick and flexible. Thus, a programer can solve most linear transfer functions by, first, using a partial fraction decomposition to reduce his system to the first- and second-order blocks; next, using subroutines to calculate the constants; and, finally, using the integration subroutines and output equations to calculate the solution of the system.

TABLE DI.- CONVERSION OF TRANSFER FUNCTIONS TO OUTPUT EQUATIONS

FOR FIRST-ORDER SYSTEMS

$$\left[\text{Forcing function is } u(t_K)\right]$$

State equation:

$$x = x_1(t_K + h) = P(h)x(t_K) + q_1(h)u(t_K) + q_2(h)\dot{u}(t_K) + q_3(h)\ddot{u}(t_K)$$

First derivative:

$$\dot{x} = x_2(t_K + h) = u(t_K + h) - Wx_1(t_K + h)$$

Transfer function	Output equation
$\dfrac{1}{s + W}$	$y(t_K + h) = x_1(t_K + h)$
$\dfrac{K}{s + W}$	$y(t_K + h) = Kx_1(t_K + h)$
$\dfrac{Ks}{s + W}$	$y(t_K + h) = Kx_2(t_K + h)$ $= K\left[u(t_K + h) - Wx_1(t_K + h)\right]$
$\dfrac{K(As + B)}{s + W}$	$y(t_K + h) = KAx_2(t_K + h) + KBx_1(t_K + h)$ $= K\left[(B - AW)x_1(t_K + h) + Au(t_K + h)\right]$

TABLE DII.- CONVERSION OF TRANSFER FUNCTIONS TO OUTPUT EQUATIONS

FOR SECOND-ORDER SYSTEMS

State equations:

$$x = x_1(t_K + h) = p_{11}x_1(t_K) + p_{12}x_2(t_K) + q_{11}u(t_K) + q_{12}\dot{u}(t_K) + q_{13}\ddot{u}(t_K)$$

$$\dot{x} = x_2(t_K + h) = p_{21}x_1(t_K) + p_{22}x_2(t_K) + q_{21}u(t_K) + q_{22}\dot{u}(t_K) + q_{23}\ddot{u}(t_K)$$

Second derivative:

$$\ddot{x} = x_3(t_K + h) = u(t_K + h) - Dx_2(t_K + h) - Ex_1(t_K + h)$$

Transfer function	Output equation
$\dfrac{1}{s^2 + Ds + E}$	$y(t_K + h) = x_1(t_K + h)$
$\dfrac{K}{s^2 + Ds + E}$	$y(t_K + h) = Kx_1(t_K + h)$
$\dfrac{Ks}{s^2 + Ds + E}$	$y(t_K + h) = Kx_2(t_K + h)$
$\dfrac{Ks^2}{s^2 + Ds + E}$	$y(t_K + h) = Kx_3(t_K + h)$ $\qquad = K\left[u(t_K + h) - Dx_2(t_K + h) - Ex_1(t_K + h)\right]$
$\dfrac{K(s + A)}{s^2 + Ds + E}$	$y(t_K + h) = K\left[Ax_1(t_K + h) + x_2(t_K + h)\right]$
$\dfrac{K(As^2 + Bs + C)}{s^2 + Ds + E}$	$y(t_K + h) = K\left[Ax_3(t_K + h) + Bx_2(t_K + h) + Cx_1(t_K + h)\right]$ $\qquad = K\left[Au(t_K + h) + (B - AD)x_2(t_K + h)\right.$ $\qquad\qquad \left. + (C - AE)x_1(t_K + h)\right]$

APPENDIX E

ARFDS PROGRAM LISTING

The ARFDS program listing is given in this appendix.

```
          PROGRAM ARFDS (INPUT=201,OUTPUT=201,TAPE33=INPUT)
       C*****
       C*****      AUTOMATIC REENTRY FLIGHT DYNAMICS SIMULATOR --- THIS PROGRAM
       C*****      WILL STUDY THE CAPABILITY OF THE SPACE SHUTTLE CONTROL SYSTEM
       C*****      AND GUIDANCE SCHEME TO FLY THE VEHICLE TO A TARGET.
       C*****
       C*****      EITHER OBLATE OR SPERICAL EARTH MODELS MAY BE SELECTED.
       C*****      INPUT QUANTITIES WILL BE IN THE GEOCENTRIC REFERENCE FRAME.
       C*****      THE CONSTANTS AND VARIABLES USED IN THIS PROGRAM ARE IN
       C*****      ENGLISH UNITS.
       C*****
000003        COMMON/REALTIM/ADC(40),DAC(96),LDISI(108),LDISO(180),
              1NOPER,NHOLD,NRESET,NTERM,NPRINT,NREAD
000003        COMMON/INTCOMM/T,H,INT,NEQ,ISCHEME,DERINT(2,15)
000003        COMMON/INTINTR/INTERN(5,15)
000003        COMMON/CCONA1/H1,NEQ1,CON1(10,6)
000003        COMMON/TABLES/TABLE(100),INTEG(10)
000003        COMMON/DRIVER/ALPDG,MACH,DELSB,DELE,DELBF,DELEL,DELER
000003        COMMON/FUNCNAM/CL,CD,CM,CYB,CNB,CLB,CYDR,CLDR,CNDR,CLE,CDE,CME,
              1CYDA,CLDA,CNDA,CLSB,CDSB,CMSB,CYBSB,CNBSB,CLBSB,CHSB,CHR,CHB,CHQ,
              2CNR,CLR,CNP,CLP,CLBF,CDBF,CMBF,CHBF,CHER,CHEL
       C
000003        REAL MACH,MACHL,L,M,N,LOD1,LMN
       C
000003        DIMENSION IVARBUF(5),LOGIC(5)
000003        DIMENSION TABTAB1(9),TABTAB2(9),TABTAB3(9)
000003        DIMENSION TAB1A(5),TAB2A(6),TAB22A(4),TAB4A(5),TAB5A(5),TAB55A(4),
              1TAB6A(5),TAB8A(6)
000003        DIMENSION OLIMITR(2),OLIMITE(2)
000003        DIMENSION RLIMITD(2),RLIMITL(2),RLIMITR(2)
000003        DIMENSION REC1(3,3)
000003        DIMENSION XYZE   (3)
000003        DIMENSION CIGAR(3,3),TA(3),TAP(3),TARE(3),UTARE(3),UXYZE(3)
000003        DIMENSION REC(3,3),RG(3),RGP(3)
000003        DIMENSION ANSW(8)
       C
000003        LOGICAL LDISI,LDISO,LOGIC
000003        LOGICAL VARCHNG,INTABLS,AUTOTYP,TYPESW,ENABLE
       C
       C*****      THESE EQUIVALENCES SET UP EASY ACCESS TO THE VARIABLES THAT
       C*****      WILL BE CHANGED FROM THE CONTROL CONSOLE BETWEEN CASES.
000003        EQUIVALENCE              (TABLE(  1),CHORD   ),(TABLE(  2),SPAN    )
              1,(TABLE(  3),S     ),(TABLE(  4),RNREF   ),(TABLE(  5),WEIGHT  )
              2,(TABLE(  6),EX    ),(TABLE(  7),EY      ),(TABLE(  8),EZ      )
              3,(TABLE(  9),EXZ   ),(TABLE( 10),EXY     ),(TABLE( 11),EYZ     )
              4,(TABLE( 12),XOFF  ),(TABLE( 13),YOFF    ),(TABLE( 14),ZOFF    )
              5,(TABLE( 15),TIC   ),(TABLE( 16),CNGRATE),(TABLE( 17),OFTEN   )
              6,(TABLE( 18),PDGIC ),(TABLE( 19),QDGIC  ),(TABLE( 20),RDGIC   )
              7,(TABLE( 21),PHIDGIC),(TABLE( 22),VIC    ),(TABLE( 23),PSIEDIC)
              8,(TABLE( 24),ALPDGIC),(TABLE( 25),BETDGIC),(TABLE( 26),GAMDGIC)
              9,(TABLE( 27),GCLDGIC),(TABLE( 28),RLONDIC),(TABLE( 29),HAIC    )
              1,(TABLE( 30),RE      ),(TABLE( 31),C1     ),(TABLE( 32),C2      )
              2,(TABLE( 33),DELAIC ),(TABLE( 34),DELRIC ),(TABLE( 35),DELEIC )
000003        EQUIVALENCE
              1 (TABLE( 36),WXRG   ),(TABLE( 37),WYRG    ),(TABLE( 38),WZRG    )
              2,(TABLE( 39),GCLP   ),(TABLE( 40),GCLR    ),(TABLE( 41),GCLB    )
              3,(TABLE( 42),GCLDA  ),(TABLE( 43),GCLDR   ),(TABLE( 44),GCM     )
              4,(TABLE( 45),GCMQ   ),(TABLE( 46),GCME    ),(TABLE( 47),GCNB    )
              5,(TABLE( 48),GCNP   ),(TABLE( 49),GCNR    ),(TABLE( 50),GCNDR   )
              6,(TABLE( 51),GCNDA  ),(TABLE( 52),GCL     ),(TABLE( 53),GCD     )
              7,(TABLE( 54),GCLE   ),(TABLE( 55),GCDE    ),(TABLE( 56),GCYB    )
              8,(TABLE( 57),GCYDR  ),(TABLE( 58),GCYDA   ),(TABLE( 59),GCLBF   )
              9,(TABLE( 60),GCDBF  ),(TABLE( 61),GCMBF   ),(TABLE( 62),GCHBF   )
```

```
                  A,(TABLE( 63),GCLSB  ),(TABLE( 64),GCDSB  ),(TABLE( 65),GCMSB )
                  B,(TABLE( 66),GCYBSB ),(TABLE( 67),GCNBSB ),(TABLE( 68),GCLBSB )
                  C,(TABLE( 69),GCHSB  ),(TABLE( 70),GCHE   ),(TABLE( 71),GCHR   )
                  D,(TABLE( 72),GCHB   ),(TABLE( 73),GUIDDT ),(TABLE( 74),DELNDIC)
            C
000003            EQUIVALENCE(INTEG( 1),NOUT   ),(INTEG( 2),NRUN   )
                 1,(INTEG( 3),IDELT  ),(INTEG( 4),ISKEM  ),(INTEG( 5),ISLECIC)
                 2,(INTEG( 6),JSCAN  ),(INTEG( 7),NSTRT  )
            C
            C*****     THE DERINT ARRAY CONTAINS THE DERIVATIVE AND INTEGRAL NAMES
            C*****     OF ALL THE VARIABLES TO BE INTEGRATED BY THE NUMERICAL SCHEME
            C*****     IN SUBROUTINE IGRATE1.THE VARIALES IN ROW 1 ARE THE INTEGRALS
            C*****      OF THOSE IN ROW 2.
U00003            EQUIVALENCE(DERINT(1, 1),P       ),(DERINT(2, 1),PDOT    )
                 1,(DERINT(1, 2),Q       ),(DERINT(2, 2),QDOT    )
                 2,(DERINT(1, 3),R       ),(DERINT(2, 3),RDOT    )
                 3,(DERINT(1, 4),PHI     ),(DERINT(2, 4),PHIDOT  )
                 4,(DERINT(1, 5),THE     ),(DERINT(2, 5),THEDOT  )
                 5,(DERINT(1, 6),PSI     ),(DERINT(2, 6),PSIDOT  )
                 6,(DERINT(1, 7),VX      ),(DERINT(2, 7),VXDOT   )
                 7,(DERINT(1, 8),VY      ),(DERINT(2, 8),VYDOT   )
                 8,(DERINT(1, 9),VZ      ),(DERINT(2, 9),VZDOT   )
                 9,(DERINT(1,10),X       ),(DERINT(2,10),XDOT    )
                 A,(DERINT(1,11),Y       ),(DERINT(2,11),YDOT    )
                 B,(DERINT(1,12),Z       ),(DERINT(2,12),ZDOT    )
                 C,(DERINT(1,13),ERALI   ),(DERINT(2,13),ERRALPD)
                 D,(DERINT(1,14),EA2I    ),(DERINT(2,14),EA2    )
                 E,(DERINT(1,15),EY5I    ),(DERINT(2,15),EY5    )
            C
            C*****     THESE EQUIVALENCES ARE USED BY THE TYPEWRITER SOFTWARE WHEN
            C*****     TYPING OUT THE CHAGES TO VARIABLES MADE AT THE CONSOLE.
000003            EQUIVALENCE (LDISI(46),AUTOTYP),(LDISI(47),TYPESW)
000003            EQUIVALENCE (LDISI(48),INTABLS)
            C
            C*****     THIS IS AN ENTRY POINT TO SUBROUTINE DATAFUN WHERE THE AERO
            C*****     FUNCTION DATA IS READ AND PACKED(TWO DATA POINTS PER WORD).
000003            CALL DATAIC
            C
            C*****     THESE DATA STATEMENTS SET UP FUNCTIONS TO BE USED WITH AERO-
            C*****     DYNAMIC INTERFERENCE CALCULATIONS OF EMX,EMY,AND EMZ.
000004            DATA TABTAB1/0.,20.,20.,0.,5.,4*0./
000004            DATA TABTAB2/0.,5.,20.,0.,2.5,5.,5.,2*0./
000004            DATA TABTAB3/0.,5.,10.,-2.,3.,0.,5.,30.,20./
000004            DATA TAB1A/.746,.63,.533,.475,.436/
000004            DATA TAB2A/-.624,-.953,-1.069,-1.069,-1.069,-1.069/
000004            DATA TAB22A/-.701,-.934,-1.166,-1.069/
000004            DATA TAB4A/.74,.616,.541,.512,.493/
000004            DATA TAB5A/0.,.076,.114,.133,.152/
000004            DATA TAB55A/.076,.076,.076,.152/
000004            DATA TAB6A/.13,.192,.23,.244,.253/
000004            DATA TAB8A/-.141,-.115,-.111,-.111,-.111,-.111/
000004            TAB3=1.
000006            TAB7=1.02
            C*****     THESE ARE CONSTANTS THAT WILL BE USED IN THE GUIDANCE SCHEME.
000007            DATA GS/32.2/
000007            DATA CNMFS/1.6457883E-04/
000007            DATA ATK/3437.746771/
000007            DATA GSTART/.05/
000007            DATA RPT/227.8/
000007            DATA VSAT/25766.1973/
000007            DATA VQ/7000./
000007            DATA ALMN1/.7986355/
000007            DATA ALFM/25./
000007            DATA RPT1/12.5/
000007            DATA RAZ /-0.7679448709/
000007            DATA RTE /20909773.5/
000007            DATA DBAR /48000./
000007            DATA RTURN /15200./
000007            DATA ALMN2/0.9659258262/
000007            DATA EEF4/2000000./
000007            DATA DF/19.09/
```

```
000007          DATA TABLE/100*0./
000007          VSATS=VSAT*VSAT
000011          SINRAZ=SIN(RAZ)
000013          COSRAZ=COS(RAZ)
        C*****       THESE STATEMENTS SET UP THE HARDWARE CHANNELS TO BE USED
        C*****       AND  SPECIFY THE STATEMENT NUMBERS USED BY THE MODE CONTROLS.
000015          CALL NM218 (5LTYPER)
000016          CALL XDSPLAY (LDISI,LDISO,VARCHNG,ITYPE,IVARBUF,INTABLS)
000022          CALL CYCLE (90006S)
000024          CALL RTROUTE(MF,90034S)
000026          CALL INOUT(ADC,0,DAC,16,LDISI,48,LDISO,180)
000036          CALL DATABLX(TABLE,100,INTEG,10,LOGIC,5,ADC,0,DAC,16,LDISI,48,
        1LDISO,180)
000054          CALL LOSTIME(90017S)
000056          ASSIGN 90001 TO NOPER
000057          ASSIGN 90002 TO NHOLD
000060          ASSIGN 90003 TO NRESET
000061          ASSIGN 90004 TO NTERM
000062          ASSIGN 90014 TO NPRINT
000063          ASSIGN 90015 TO NREAD
        C*****       THESE STATEMENTS SPECIFY WHICH VARIABLES WILL BE PRINTED.THE
        C*****       FREQUENCY IS SPECIFIED BY NOUT.
000064          NOUT=1
000065          CALL READOUT(6,NOUT,T,HA,GCLATDG,RLONDG,V,RN)
000075          CALL READOUT(6,NOUT,MACH,ISLECT,ACCEL,QB,PDG,QDG)
000105          CALL READOUT(6,NOUT,RDG,THEDG,PSIDG,TRANGE,BETDG,GAMDG)
000115          CALL READOUT(6,NOUT,PDOT,QDOT,RDOT,PITERR,ERROLL,ERRYAW)
000125          CALL READOUT(6,NOUT,ALPDG,PHIDG,DELE,DELA,DELR,DELSB)
000135          CALL READOUT(6,NOUT,DELBF,ALPHAC,PHICM,DLEC,DELAC,DELRC)
000145          CALL READOUT(3,NOUT,NUDCTR,PSIEDG,VR)
000151          CALL READOUT(1,NOUT,FUELUSE)
000154          CALL READOUT(6,NOUT,TROLN,TROLP,TPITN,TPITP,TYAWN,TYAWP)
000164          CALL READOUT(6,NOUT,DELANUD,RPRI,DELEL,DELER,VBAR,VINERT)
000174          CALL READOUT(6,NOUT,HMER,HMEL,HMR,RUDR,ELERL,ELERR)
000204          CALL READOUT(3,NOUT,DRAG,HADOT,ERRPHI)
        C*****       THESE INITIALIZE LOGICAL DISCRETE OUTPUT AND INPUT CHANNELS
        C*****       TO FALSE.
000210          DO 90007 INDEX=1,180
000212    90007 LDISO(INDEX) = .FALSE.
000215          DO 90008 NDEX=1,48
000216    90008 LDISI(NDEX)=.FALSE.
        C*****            CONSTANTS SECTION ----
        C*****       THESE CONSTANTS AND CALCULATIONS,DOWN TO STATEMENT 90003,ARE
        C*****       ONES THAT NEED BE DONE ONLY ONCE WHEN THE PROGRAM IS READ IN.
        C*****       MANY VALUES INITIALIZED HERE CAN BE CHANGED FROM THE CONSOLE.
000221          LOGIC(1)=LOGIC(2)=LOGIC(3)=.F.
000225          JSCAN=50
000226          NRUN=0
000227          NEQ=15
000230          ISKEM=1
000231          IDELT=25
000232          NEQ1=6
000233          CON1(6,1)=.05
000234          CON1(6,2)=.05
000235          CON1(6,3)=1.
000236          CON1(6,4)=.05
000237          CON1(6,5)=.05
000240          CON1(6,6)=2.
000241          PI=3.1415926535
000243          HPI=.5*PI
000244          TPI=2.*PI
000245          DTR=PI/180.
000247          RTD=180./PI
000250          FTM=.3048
000251          PTS=1./32.174
000253          GE=32.217
000254          RLONT=DTR*(-120.5338)
000256          COSLONT=COS(RLONT)
000261          SINLONT=SIN(RLONT)
000263          GCLATT=DTR*34.55577617
000265          COSGCLT=COS(GCLATT)
000267          SINGCLT=SIN(GCLATT)
```

```
000271          REC(1,1)=-COSRAZ*SINGCLT*COSLONT -SINRAZ*SINLONT
000275          REC(1,2)=-COSRAZ*SINGCLT*SINLONT +SINRAZ*COSLONT
000300          REC(1,3)= COSRAZ*COSGCLT
000301          REC(2,1)= SINRAZ*SINGCLT*COSLONT -COSRAZ*SINLONT
000304          REC(2,2)= SINRAZ*SINGCLT*SINLONT +COSRAZ*COSLONT
000307          REC(2,3)=-SINRAZ*COSGCLT
000310          REC(3,1)=-COSGCLT*COSLONT
000312          REC(3,2)=-COSGCLT*SINLONT
000314          REC(3,3)=-SINGCLT
000315          DETERM=REC(1,1)*(REC(2,2)*REC(3,3)-REC(2,3)*REC(3,2))
                1-REC(2,1)*(REC(1,2)*REC(3,3)-REC(1,3)*REC(3,2))
                2+REC(3,1)*(REC(1,2)*REC(2,3)-REC(1,3)*REC(2,2))
000334          REC1(1,1)=REC(2,2)*REC(3,3)-REC(2,3)*REC(3,2)
000337          REC1(1,2)=REC(1,3)*REC(3,2)-REC(1,2)*REC(3,3)
000343          REC1(1,3)=REC(1,2)*REC(2,3)-REC(1,3)*REC(2,2)
000346          REC1(2,1)=REC(2,3)*REC(3,1)-REC(2,1)*REC(3,3)
000352          REC1(2,2)=REC(1,1)*REC(3,3)-REC(1,3)*REC(3,1)
000356          REC1(2,3)=REC(1,3)*REC(2,1)-REC(1,1)*REC(2,3)
000361          REC1(3,1)=REC(2,1)*REC(3,2)-REC(2,2)*REC(3,1)
000365          REC1(3,2)=REC(1,2)*REC(3,1)-REC(1,1)*REC(3,2)
000371          REC1(3,3)=REC(1,1)*REC(2,2)-REC(1,2)*REC(2,1)
000374          DO 40  I=1,3
000376          DO 41  J=1,3
000377          REC1(I,J)=REC1(I,J)/DETERM
000403       41 CONTINUE
000405       40 CONTINUE
000407          NSTRT=0
000410          GUIDDT=.04
000411          CHORD=39.5666667
000413          HCHORD=.5*CHORD
000414          SPAN=78.056666667
000416          HSPAN=.5*SPAN
000417          S=2690.
000421          RNREF=110.7
000422          WEIGHT=182986.
000424          EX=759000.
000425          EY=5765000.
000427          EZ=5912000.
000430          EXY=EYZ=0.
000432          EXZ=131000.
000433          XOFF=ZOFF=0.
000435          YOFF=.125
000436          DELBF=-11.7
000440          WXRG=WYRG=WZRG=0.
000443          TIC=0.
000444          PDGIC=QDGIC=RDGIC=0.
000447          PHIDGIC=-15.
000450          VIC=26046.
000452          PSIEDIC=336.72
000453          ALPDGIC=34.25
000455          BETDGIC=0.
000456          GAMDGIC=-.84
000457          GCLDGIC=47.84
000461          RLONDIC=17.2
000462          HAIC=400000.
000464          DELAIC=0.
000465          DELRIC=0.
000466          DELEIC=1.
000467          ISLECIC=0
000470          DELNDIC=0.
000471          CNGRATE=0.
000472          OFTEN=5.
000473          GCLP=GCLR=GCLB=GCLDA=GCLDR=1.
000500          GCM=GCMQ=GCLE=GCDE=GCME=GCL=GCD=1.
000507          GCNB=GCNP=GCNR=GCNDR=GCNDA=1.
000514          GCLSB=GCDSB=GCMSB=GCYBSB=GCNBSB=GCLBSB=GCHSB=GCHR=GCHB=GCHE=1.
000526          GCLBF=GCDBF=GCMBF=GCHBF=1.
000532          GCYB=GCYDR=GCYDA=1.
000535          GCLBF=GCDBF=GCMBF=1.
000540          GCLSB=GCDSB=GCMSB=GCYBSB=GCNBSB=GCLBSB=1.
000546          SLPAQ=7.5/20.
000550          TISPH=750./242.
```

```
000552          OLIMITR(1)=-22.8        $OLIMITR(2)=22.8
000555          OLIMITE(1)=-35.        $OLIMITE(2)=20.
000560          C1=C2=1.
000563          HAMAX=500000.*FTM
      C*****       AFTER CALL READY,THE PROGRAM WILL LOOP IN SYNCHRONIZATION
      C*****       WITH THE REAL-TIME CLOCK AS LONG AS THE MODE SELECTED IS
      C*****       RESET,HOLD,OR OPERATE.
000565          CALL READY
000566    90003 CONTINUE
      C*****           RESET SECTION ----
      C*****       THE RESET LOOP IS FROM STATEMENT 90003 TO 90050.WHILE THE
      C*****       RESET SWITCH IS DEPRESSED,ONLY THOSE EQUATIONS ARE EXECUTED.
      C*****       NO INTEGRATION IS PERFORMED.
000566          LDISO(111)=LDISO(112)=LDISO(113)=LDISO(114)=.F.
000573          INT=0
000574          ISCHEME=ISKEM
000575          ISLECT=ISLECIC
000577          ISTRT=NSTRT
000600          ISTART=0
000601          IFT=0
000602          IDFG2=IDFG3=0
000604          DELANUD=DELNDIC
000605          ISTP=0
000606          BA1=0.
000607          VBB=20500.
000610          YB=.3054326191
000612          TW=1000.
000613          H=1./IDELT
000616          RATELIM=H*20.
000620          RATLIMR=H*10.
000622          H1=H
000623          HI=1./H
000624          CALL PQ1
000625          DO 51 J=1,NEQ1
000627          CON1(1,J)=2B
000632          CON1(2,J)=0.
000635          CON1(3,J)=0.
000637          CON1(4,J)=0.
000641          CON1(5,J)=0.
000644       51 CONTINUE
000646          LATROL=LATPIT=LATYAW=LATDLA=0
000652          PITN=PITP=ROLN=ROLP=YAWN=YAWP=0.
000660          TROLN=TROLP=TPITN=TPITP=TYAWN=TYAWP=FUELUSE=0.
000667          OMASS=WEIGHT*PTS
000671          OMASSI=1./OMASS
000672          TERMA=EY *EZ -EYZ*EYZ
000675          TERMB=EXY*EZ +EXZ*EYZ
000700          TERMC=EXY*EYZ+EY *EXZ
000703          TERMD=EX *EZ -EXZ*EXZ
000706          TERME=EX *EYZ+EXZ*EXY
000711          TERMF=EX *EY -EXY*EXY
000714          DET=EX*TERMA -EXY*TERMB -EXZ*TERMC
000721          DETI=1./DET
000722          PITERR=ERROLL=ERRYAW=0.
000725          ERRPHI=0.
000726          DELRC=0.
000727          DELAC=0.
000730          DLECL=DLECR=0.
000732          DELANUD=0.
000733          ERRALPD=0.
000734          HMR=0.
000735          PASTD=PASTR=PASTL=0.
000740          RLIMITD(1)=RLIMITD(2)=0.
000742          ERALI=EA2I=EY5I=0.
000745          EA2=EY5=0.
000747          OMEGA=.729211515 E-4
      C*****       LOGIC 2 TRUE WILL GIVE A SPHERICAL EARTH WITH RADIUS RE.
000750          IF(LOGIC(2))53,54
000752       53 OJ=0.
000753          OMU=1.407662686E+16
000755          RE=20902910.
000756          RP=RE
```

```
000757          GO TO 55
000760       54 OJ=1.5*(1.082639 E-3)
000762          OMU=1.407645794E+16
000764          RE=20925662.73
000765          RP=20855024.2
000767       55 CONTINUE
000767          REORP=RE/RP
000771          REORPS=REORP*REORP
000772          RERPM1=REORPS-1.
000774          T=TIC
.000775         GCLAT=DTR*GCLDGIC
.000777         RLON=DTR*RLONDIC
001001          P=DTR*PDGIC
001003          Q=DTR*QDGIC
001005          R=DTR*RDGIC
001007          PHI=DTR*PHIDGIC
001011          VR=VIC
001012          PSIE=DTR*PSIEDIC
001014          HA=HAIC
001015          ALPHA=DTR*ALPDGIC
001017          BETA=DTR*BETDGIC
001021          GAMMA=DTR*GAMDGIC
001023          DELA=DELAIC
001024          DELR=DELRIC
001026          DELE=DELEIC
001027          DELEL=DELE+DELA
001031          DELER=DELE-DELA
001032          X1=DELEL
001034          X2=DELER
001035          X3=DELR
001037          ELERR=ELERL=RUDR=0.
001042          SINALP=SIN(ALPHA)
001044          COSALP=COS(ALPHA)
001046          SINBET=SIN(BETA)
001050          COSBET=COS(BETA)
001052          SINGAM=SIN(GAMMA)

001054          COSGAM=COS(GAMMA)
001056          SINPHI=SIN(PHI)
001060          COSPHI=COS(PHI)
001062          SINPSIE=SIN(PSIE)
001064          COSPSIE=COS(PSIE)
      C*****        CALCULATE COMPONENTS OF INERTIAL POSITION (X,Y,Z) FROM GCLAT,
      C*****        HA,AND RLON.
001066          SINGCL=SIN(GCLAT)
001070          COSGCL=COS(GCLAT)
001072          RSO=RE/SQRT(1. +RERPM1*SINGCL*SINGCL)
001101          REH=RSO+HA
001103          RLONI=RLON +OMEGA*T
001106          COSLONI=COS(RLONI)
001110          SINLONI=SIN(RLONI)
001112  ,       X=REH*COSGCL*COSLONI
001114          Y=REH*COSGCL*SINCONI
001116          Z=REH*SINGCL
      C*****        CALCULATE COMPONENTS OF INERTIAL VELOCITY (VX,VY,VZ) FROM
      C*****        GCLAT,RLON,GAMMA,PSIE,HA,AND VR.
001120          VXIG= VR*COSGAM*COSPSIE
001122          VYIG= VR*COSGAM*SINPSIE +REH*OMEGA*COSGCL
001127          VZIG=-VR*SINGAM
001131          VX=-VXIG*SINGCL*COSLONI -VYIG*SINLONI -VZIG*COSGCL*COSLONI
001140          VY=-VXIG*SINGCL*SINLONI +VYIG*COSLONI -VZIG*COSGCL*SINLONI
001146          VZ=VXIG*COSGCL  -VZIG*SINGCL
      C*****        CALCULATE THE INITIAL ANGLES (PSI,THE) FROM ALPHA,BETA,PHI,
      C*****        GAMMAA,AND PSIA (CONSTANT WINDS ACCOUNTED FOR).
001151          VYRG=VYIG -REH*OMEGA*COSGCL
001155          VWXRG=VXIG -WXRG
001157          VWYRG=VYRG -WYRG
001161          VWZRG=VZIG -WZRG
001163          VXVYR=SQRT(VWXRG*VWXRG +VWYRG*VWYRG)
001170          GAMMAA=ATAN2(-VWZRG,VXVYR)
001174          SINGAMA=SIN(GAMMAA)
001176          PSIA=ATAN2(VWYRG,VWXRG)
001201          SINPSIA=SIN(PSIA)
```

```
001203          COSPSIA=COS(PSIA)
001205          B11=SINPHI*SINBET +COSPHI*SINALP*COSBET
001211          B3 =COSALP*COSALP*COSBET*COSBET +B11*B11
001215          B13=COSALP*COSBET*SINGAMA +B11*SQRT(B3-SINGAMA*SINGAMA)
001226          THE=ASIN(B13/B3)
001232          SINTHE=SIN(THE)
001234          COSTHE=COS(THE)

001236          B17=COSPHI*SINBET -SINPHI*COSBET*SINALP
001242          B18=COSTHE*COSALP*COSBET +SINTHE*B11
001246          ZT=B18*SINPSIA -B17*COSPSIA
001252          ZB=B18*COSPSIA +B17*SINPSIA
001255          PSI=ATAN2(ZT,ZB)
001260    90002 CONTINUE
          C*****          HOLD SECTION ----
          C*****     WHILE THE HOLD SWITCH IS DEPRESSED,THE EQUATIONS BETWEEN
          C*****     90002 AND 90050 ARE EXECUTED.AGAIN,NO INTEGRATION IS PER-
          C*****     FORMED.
001260    90006 CONTINUE
          C*****          OPERATE SECTION ----
          C*****     WHILE THE OPERATE SWITCH IS DEPRESSED,THE EQUATIONS BETWEEN
          C*****     90006 AND 90014 ARE EXECUTED.THE CALLS TO IGRATE1,CONVO1,AND
          C*****     MODLAG PERFORM THE INTEGRATIONS.
001260          PDG=P*RTD
001262          QDG=Q*RTD
001264          RDG=R*RTD
001265          IF(ABS(PHI)-PI)61,61,60
001271       60 PHI=PHI -SIGN(TPI,PHI)
001275       61 IF(ABS(THE)-HPI)63,63,62
001301       62 THE=-THE +SIGN(PI,THE)
001305       63 IF(ABS(PSI)-PI)65,65,64
001311       64 PSI=PSI -SIGN(TPI,PSI)
001315       65 CONTINUE
001315          PHIDG=PHI*RTD
001317          THEDG=THE*RTD
001321          PSIDG=PSI*RTD
001322          SINPHI=SIN(PHI)
001325          COSPHI=COS(PHI)
001327          SINTHE=SIN(THE)
001331          COSTHE=COS(THE)
001333          TANTHE=TAN(THE)
001335          SINPSI=SIN(PSI)
001337          COSPSI=COS(PSI)
001341          ZS=Z*Z
001342          REH=SQRT(X*X +Y*Y +ZS)
001350          REHI=1./REH
001351          REHIS=REHI*REHI
001352          REOREH=RE*REHI
001354          REOREHS=REOREH*REOREH
001355          ZOREHS=ZS *REHIS
001357          PZR=1. +OJ*REOREHS*(1.-5.*ZOREHS)
001364          OMPR=-OMU*PZR*REHIS*REHI
001370          GXI=OMPR*X
001372          GYI=OMPR*Y
001374          GZI=OMPR*Z/PZR*(1. +OJ*REOREHS*(3.-5.*ZOREHS))
001404          GCLAT=ASIN(Z*REHI)
001410          GCLATDG=GCLAT*RTD
001412          SINGCL=SIN(GCLAT)
001414          COSGCL=COS(GCLAT)
001416          RLONI=ATAN2(Y,X)
001421          SINLONI=SIN(RLONI)
001423          COSLONI=COS(RLONI)
001425          RLON=RLONI -OMEGA*T
001430          RLONDG=RTD*RLON
001431          SINLON=SIN(RLON)
001434          COSLON=COS(RLON)
001436          RSO=RE/SQRT(1. +RERPM1*SINGCL*SINGCL)
001445          HA=REH-RSO
001447          VXIG=-VX*SINGCL*COSLONI -VY*SINGCL*SINLONI +VZ*COSGCL
001455          VYIG=-VX*SINLONI +VY*COSLONI.
001461          VZIG=-VX*COSGCL*COSLONI -VY*COSGCL*SINLONI -VZ*SINGCL
001466          VXRG=VXIG
001470          VYRG=VYIG -REH*OMEGA*COSGCL
001474          VZRG=VZIG
001475          EL1=COSPSI*COSTHE
001477          EL2=SINPSI*COSTHE
001501          EL3=-SINTHE
```

```
001502          EM1=COSPSI*SINTHE*SINPHI  -SINPSI*COSPHI
001505          EM2=SINPSI*SINTHE*SINPHI  +COSPSI*COSPHI
001510          EM3=COSTHE*SINPHI
001511          EN1=COSPSI*SINTHE*COSPHI  +SINPSI*SINPHI
001515          EN2=SINPSI*SINTHE*COSPHI  -COSPSI*SINPHI
001517          EN3=COSTHE*COSPHI
001521          VWXRG=VXRG-WXRG
001523          VWYRG=VYRG-WYRG
001525          VWZRG=VZRG-WZRG
001527          UB=VWXRG*EL1 +VWYRG*EL2 +VWZRG*EL3
001535          VB=VWXRG*EM1 +VWYRG*EM2 +VWZRG*EM3
001543          WB=VWXRG*EN1 +VWYRG*EN2 +VWZRG*EN3
001551          V=SQRT(UB*UB +VB*VB +WB*WB)
001557          VR=SQRT(VXRG*VXRG +VYRG*VYRG +VZRG*VZRG)
001565          ALPHA=ATAN2(WB,UB)
001570          ALPDG=ALPHA*RTD
001572          SINALP=SIN(ALPHA)
001574          COSALP=COS(ALPHA)
001576          TANALP=TAN(ALPHA)
001600          BETA=ASIN(VB/V)
001604          BETDG=BETA*RTD
001606          VXVY=SQRT(VXRG*VXRG+VYRG*VYRG)
001613          GAMMA=ATAN2(-VZRG,VXVY)
001617          GAMDG=GAMMA*RTD
001621          COSGAM=COS(GAMMA)
001623          SINGAM=SIN(GAMMA)
001625          PSIE=ATAN2(VYRG,VXIG)
001630          PSIEDG=PSIE*RTD
001632          SINPSIE=SIN(PSIE)
001634          COSPSIE=COS(PSIE)
001636          VI=1./V
001640          VS=V*V
001641          VSOVS=VS/VSATS
001643          VXS=VX*VX
001644          VYS=VY*VY
001646          VZS=VZ*VZ
001647          VINERT=SQRT(VXS +VYS +VZS)
        C*****        UPDATE THE ATMOSPHERIC VARIABLES.
001655          CALL ATMOS3(HA*FTM,HAMAX,ANSW)
001661          TAMB=518.67*ANSW(2)
001663          RHO=ANSW(3)*.00237689
001665          CS=ANSW(4)*1116.45
001667          RMU=ANSW(8)*.373717E-6
001671          QB=.5*RHO*VS
001674          QS=QB*S
001676          FQBAR=28.8/(QB+10.)
001701          RQQB=2./(QB +10.)
001703          QSO=QS*OMASSI
001705          RN=RHO*V*RNREF/RMU
001711          MACH=V/CS
001713          HADOT=-VZRG
001714          XDOT=VX
001716          YDOT=VY
001717          ZDOT=VZ
        C*****        GO TO SUBROUTINE DATAFUN AND CALCUATE AERO FUNCTIONS THAT
        C*****        DEPEND ON MACH AND ALPHA.
001721          IF(LDISI(37))800,801
001722      800 IF(INT-1)801,801,802
001725      801 CALL DATAFUN(1)
001727      802 CONTINUE
001727          CCL=GCL*CL
001731          CCD=GCD*CD
        C*****        IF LOGIC 1 IS TRUE,THE VISCOUS INTERACTION WILL BE BY-PASSED
001733          IF(LOGIC(1))GO TO 706
001735          IF(MACH.LT.4.)GO TO 706
001737          HO=.24*TAMB +VS/50063.
001743          SHR=17700.*SQRT(RHO)*(V*.0001)**3.07
001755      701 SHRTWHO=SHR*(1.-.24*TW/HO)
001762          F=-TW +1272.992266*(SHRTWHO**.25)
001767          FP=-1. -76.37953596*SHR/(HO*(SHRTWHO**.75))
001777          TW1=TW -F/FP
002002          IF(ABS(TW1-TW)-1.)704,704,702
002006      702 TW=TW1
002010          GO TO 701
002010      704 TW=TW1
```

```
002012          CPR=SQRT(TW/TAMB)*(TAMB+198.72)/(TW+198.72)
002024          VBAR=MACH*SQRT(CPR/RN)
002031          IF(VBAR-.007)706,707,707
002033      706 CLV=CDV=CMV=0.
002036          GO TO 709
002037      707 VBAROI=VBAR/.007
002041          CAF=CCD*COSALP -CCL*SINALP
002045          DCAF=(VBAROI**.636 -1.)*CAF
002053          CLV=-DCAF*SINALP
002055          CDV= DCAF*COSALP
002057          CMV=DCAF*.11583825
002061      709 CCL1=CCL +CLV
002063          CCD1=CCD +CDV
002065      710 CONTINUE
         C*****      GUIDANCE SCHEME --- THE CODE FROM HERE TO CONTROL SYSTEM TEN
         C*****      IS USED TO GENERATE COMMANDED ALPHA AND PHI.
002065          IF(LDISI(33))313,309
002067      313 IF(INT-1)314,314,510
002072      314 IF(AMOD(T +.0025,GUIDDT) -.005)309,309,510
002100      309 XLFAC=QSO*SQRT(CCL1*CCL1 +CCD1*CCD1 +(GCYB*CYB*BETDG)**2)
002111          XLOD=CCL1/CCD1
002115          IF(XLOD-.5)300,301,301
002115      300 XLOD=.5
002117      301 XLODS=XLOD*XLOD
002121          DRAG=XLFAC/SQRT(1. +XLODS)
002126          IF(V-15000.)303,307,302
002130      302 IF(V-22000.)307,307,304
002133      303 HS=19777.7 +.48148*V
002136          GO TO 305
002137      304 HS=62750. -1.625*V
002142      305 IF(HS-20500.)306,308,308
002145      306 HS=20500.
002147          GO TO 308
002147      307 HS=27000.
002151      308 CONTINUE
         C*****         TARGETING
002151          XYZE(1)=REH*COSLON*COSGCL
002154          XYZE(2)=REH*SINLON*COSGCL
002156          XYZE(3)=REH*SINGCL
002160          CALL MULT31(REC,XYZE,RG)
002162          RG(3)=RG(3)+RTE
002164          RGP(1)=RG(1) +DBAR
002166          RGP(2)=RG(2) -SIGN(RTURN,RG(2))
002172          RGP(3)=RG(3)
002174          DT=SQRT(AMAX1(0.,(RGP(1)*RGP(1) +RGP(2)*RGP(2) -RTURN*RTURN)))
002206          DTH=ATAN2(RTURN,DT)
002211          DVHEAD=ATAN2(-RGP(2),-RGP(1))-SIGN(DTH,RG(2))
002223          U=ABS(DVHEAD)
002225          IF(RGP(1))312,312,310
002226      310 IF(ABS(RG(2))-RTURN)311,312,312
002232      311 U=TPI -U
002234      312 CONTINUE
002234          ARC=RTURN*U
002236          RGP(1)=-DT*COS(DVHEAD)
002242          RGP(2)=-DT*SIN(DVHEAD)
002246          RG(3)=RG(3)-RTE
002250          DO 315 J=1,3
002252      315 TARE(J)=RG(J) -RGP(J)
002256          CALL UNVEC(RG,UXYZE)
002260          CALL UNVEC(TARE,UTARE)
002262          CTH=DOT(UXYZE,UTARE)
002265          TRANGE=ATK*ACOS(CTH)+ARC*CNMFS
002272          DO 316 J=1,3
002274      316 TA(J)=-RGP(J)
002277          CALL MULT31(REC1,TA,TA)
002302          CIGAR(1,1)=-SINGCL*COSLON
002304          CIGAR(1,2)=-SINGCL*SINLON
002306          CIGAR(1,3)= COSGCL
002310          CIGAR(2,1)=-SINLON
002311          CIGAR(2,2)= COSLON
002312          CIGAR(2,3)=0.
```

95

```
002313          CIGAR(3,1)=-COSGCL*COSLON
002314          CIGAR(3,2)=-COSGCL*SINLON
002315          CIGAR(3,3)=-SINGCL
002316          CALL MULT31(CIGAR,TA,TAP)
002321          DO 322 J=1,2
002323          IF(ABS(TAP(J))-1.E-10)321,321,322
002327     321  TAP(J)=SIGN(1.E-10,TAP(J))
002333     322  CONTINUE
002335          PSIET=ATAN2(TAP(2),TAP(1))
002340          DELAZ=PSIE -PSIET
002342          IF(PSIE*PSIET)325,325,328
002343     325  IF(ABS(PSIE)-HPI)328,328,326
002347     326  DELAZ=DELAZ -TPI
002351          IF(PSIE)327,328,328
002353     327  DELAZ=DELAZ +2.*TPI
002356     328  CONTINUE
002356          IF(ISTART-1)330,331,330
002360     330  RK2ROL=-SIGN(1.,DELAZ)
002364     331  ISTART=1
002365          IF(YB -.0890117918)345,335,335
002370     335  YB=-.1308996939 +.0001090830783*V
002373          IF(YB -.3054326191)336,339,337
002376     336  IF(YB -.0872664626)338,339,339
002401     337  YB=.3054326191
002403          GO TO 339
002403     338  YB=.0872664626
002405     339  CONTINUE
002405          LMN=ALMN2*XLOD
002407          IF(ABS(DELAZ) -(YB-.03))341,341,340
002414     340  LMN=ALMN1*XLOD
002416     341  IF(V-3500.)342,345,345
002421     342  LMN=.93969*XLOD
C*****          CONTROLLER GAIN SELECTION
002423     345  C17=.00678 -.000264*DRAG
002426          IF(DRAG-40.)351,351,350
002431     350  C16=.065
002433          GO TO 360
002433     351  IF(DRAG-20.)353,353,352
002436     352  C16=.105 -.001*DRAG
002441          GO TO 360
002442     353  IF(DRAG-7.5)355,355,354
002445     354  C16=.1794 -.00472*DRAG
002450          GO TO 360
002451     355  C16=.35 -.0274667*DRAG
002454          C17=.014 -.00122667*DRAG
002457     360  IF(C17-.003)361,362,362
002462     361  C17=.003
002464     362  CONTINUE
002464          IF(V-13600.)901,901,900
002467     900  C4=C5=0.
002471          GO TO 910
002472     901  IF(V-13100.)903,903,902
002475     902  C4=-1768.
002477          C5=.13
002500          GO TO 910
002501     903  IF(V-6845.)905,905,904
002504     904  C4=-65.
002506          C5=0.
002507          GO TO 910
002507     905  IF(V-6375.)907,907,906
002512     906  C4=-546.25
002514          C5=.07
002515          GO TO 910
002516     907  IF(V-5800.)909,909,908
002521     908  C4=-100.
002523          C5=0.
002524          GO TO 910
002524     909  C4=83.1
002526          C5=-.03261
002527     910  CONTINUE
002527          T1=(VINERT*VINERT/VSATS -1.)*GS
002533          T2=CNMFS*(VS -VQ*VQ)/(2.*(TRANGE-RPT))
002541          IF(ISTRT)365,366,365
```

96

```
002542        365 GO TO(370,380,390,400)ISLECT
002552        366 IF(XLFAC-(GSTART*GS))367,370,370
              C*****        CONSTANT ATTITUDE PHASE
002556        367 LOD1=ABS(LMN)
002560            LDISO(109)=.T.
002562            LDISO(62)=LDISO(63)=LDISO(64)=LDISO(65)=.F.
002567            LDISO(46 )=LDISO(47 )=LDISO(48 )=.F.
002573            GO TO 410
              C*****        CONSTANT HEAT RATE PHASE
002573        370 ISTRT=1
002574            IF(V-20500.)380,382,382
002577        371 DREFP=C21 +C22*V +C23*VS
002605            IF(DREFP-T2)372,372,390
002607        372 RDTREF=-HS*(2.*C21*VI +C22) +C4 +C5*V
002617            ALDREF=-T1/DREFP
002621            ISLECT=1
002622            LDISO(46 )=LDISO(62 )=.T.
002625            LDISO(61)=LDISO(63)=LDISO(64)=LDISO(65)=.F.
002631            LDISO(109)=LDISO(47 )=LDISO(48 )=.F.
002635            GO TO 405
              C*****        EQUILIBRIUM GLIDE RANGING PHASE
002635        380 ISLECT=2
002636            LDISO(47 )=LDISO(63 )=.T.
002641            LDISO(61)=LDISO(62)=LDISO(64)=LDISO(65)=.F.
002645            LDISO(109)=LDISO(46 )=LDISO(48 )=.F.
002651            IF(V-9000.)400,381,381
002653        381 IF(DREFP-(T2-2.))382,382,450
002660        382 ISTP=1
002661        383 BA=BA1*DTR
002663            IF(IFT)385,384,385
002664        384 BA=60.*DTR
002666            BA1=60.
002667        385 VBB=V
002671            IF(V-20500.)421,421,420
002673        420 VBB=20500.
002675        421 COSBA=COS(BA)
002700            TEMP=1.4*VSATS*COSBA/GS
002703            IF((ALFM*TEMP )-VSATS)422,423,423
002706        422 VCG=SQRT(VSATS-(ALFM*TEMP ))
002714            IF(VCG-VQ)423,425,425
002716        423 VCG=VQ
002720        425 IF(V-20500.)435,426,426
002723        426 VBBS=VBB*VBB
002725            D23=-GS*(VBBS/VSATS -1.)/(COSBA*1.4)
002734            AK=-3.5*D23/VBB
002736            C23=-AK/11000.
002740            C22=-52000.*C23
002741            C21=D23 -C22*VBB -C23*VBBS
002746            TT11=-.5/C23*ALOG((C21 +C22*VBB +C23*VBBS)/(C21 +C22*V +C23*VS))
002762            SQ=4.*C23*C21 -C22*C22
002765            IF(SQ)430,429,429
002767        429 SQQ=SQRT(SQ)
002772            TT22=C22/(C23*SQQ)*(ATAN((2.*C23*VBB+C22)/SQQ) -ATAN((2.*C23*V +
                 1C22)/SQQ))
003014            GO TO 435
003014        430 SQQ=SQRT(-SQ)
003020            E1=(2.*C23*VBB +C22 -SQQ)/(2.*C23*VBB +C22 +SQQ)
003030            E2=(2.*C23*V +C22 -SQQ)/(2.*C23*V +C22 +SQQ)
003040            TT22=(C22/(C23*2.*SQQ))*ALOG(E1/E2)
003050        435 VBBS=VBB*VBB
003052            RFF=(TT11+TT22)*CNMFS
003055            IF(VCG-V)436,436,450
003057        436 IF(V-20500.)437,438,438
003062        437 RFF=0.
003063        438 REQ=.5*TEMP *CNMFS*ALOG((VCG*VCG -VSATS)/(VBBS -VSATS))
003075            RCG=CNMFS*(VCG*VCG -VQ*VQ)/(2.*ALFM)
003102            GO TO (440,445,450)ISTP
003111        440 R11=REQ +RCG +RFF
003114            DBB=-5.
003116            IF(BA1-35.)442,442,441
003120        441 DBB=5.
003122        442 BA2=BA1-DBB
003124            BA=BA2*DTR
003126            ISTP=2
003127            GO TO 421
003127        445 R12=REQ +RCG +RFF
```

```
003132          BAD=BA1+((TRANGE-R11-RPT)/((R11-R12)/(BA1-BA2)))
003141          IF(BAD)446,449,447
003143      446 BAD=0.
003144          GO TO 449
003145      447 IF(BAD-85.)449,449,448
003150      448 BAD=85.
003152      449 CONTINUE
003152          BA1=BAD
003154          IFT=1
003155          ISTP=3
003156          GO TO 383
003156      450 TDREF=(VSQVS      -1.)*GS
003161          COSBADD=1.4*COS(BAD*DTR)
003167          DREFP=-TDREF/COSBADD
003171          RDTREF=-2.*HS*DREFP*VI -2.*HS*GS*V/(COSBADD*VSATS) +C4 +C5*V
003204          VOLD=V
003205          RDTOLD=RDTREF
003207          ALDREF=-T1/DREFP +2.*HS*GS/(COSBADD*VSATS) +2.*HS*DREFP/VS
003221          IF(V-20500.)457,457,371
003223      457 IF(DREFP-T2)405,390,390
       C*****          CONSTANT DRAG PHASE
003226      390 ISLECT=3
003227          LDISO(48 )=LDISO(64 )=.T.
003232          LDISO(61)=LDISO(62)=LDISO(63)=LDISO(65)=.F.
003236          LDISO(109)=LDISO(46 )=LDISO(47 )=.F.
003242          IF(V-9000.)400,391,391
003244      391 DREFP=T2
003246          RDTREF=-2.*HS*DREFP*VI +C4 +C5*V
003254          IF(IDFG2)392,392,393
003255      392 V1OLD=VOLD -500.
003257          RDC=.002*(RDTREF -RDTOLD)
003262          IDFG2=1
003263      393 IF(V-V1OLD)395,395,394
003266      394 RDTREF=RDTREF -RDC*(V -V1OLD)
003272      395 VOLD2=V
003274          RDTOL2=RDTREF
003275          ALDREF=-T1/T2 +2.*HS*DREFP/VS
003303          GO TO 405
       C*****          TRANSITION PHASE
003303      400 ISLECT=4
003304          LDISO(65)=.T.
003305          LDISO(61)=LDISO(62)=LDISO(63)=LDISO(64)=.F.
003312          LDISO(109)=LDISO(46)=LDISO(47)=LDISO(48)=.F.
003317          ITR=1
003320          EEF=GS*HA +.5*VS
003323      460 IF(DREFP-(DF+.01))396,398,398
003327      396 IF(DREFP-(DF-.01))398,398,397
003333      397 DREFP=DF +SIGN(.01,DREFP-DF)
003340      398 IF(DREFP -.01)399,401,401
003343      399 DREFP=.01
003345      401 C11=(DREFP -DF)/(EEF -EEF4)
003351          RER1=-CNMFS/C11*ALOG(DF/DREFP)
003357          GO TO(465,470)ITR
003365      465 R11=RER1
003367          ITR=2
003370          DREFP=DREFP+1.
003372          GO TO 460
003372      470 R12=RER1
003374          DREFP=DREFP-((TRANGE-R11-RPT1)/(R11-R12))-1.
003403          IF(DREFP-.01)466,467,467
003405      466 DREFP=.01
003407      467 DLIM=2.2*GS/SQRT(1. +XLODS)
003416          IF(DREFP-DLIM)472,472,471
003420      471 DREFP=DLIM
003422          C11=0.
003423      472 IF(V-7600.)473,475,475
003426      473 C16=2.*C16
003430          C17=0.9*C17
003431      475 CAGI=1./(VS +2.*GS*HS)
003436          VC=VS*V
003440          RDTREF=(-HS*(2.*DREFP*V -C11*VC) +C4*VS +C5*VC)*CAGI
003452          IF(IDFG3)476,476,477
003453      476 V2OLD=VOLD2 -500.
003455          RDC=(RDTREF -RDTOL2)*.002
```

98

```
003460          IDFG3=1
003461      477 IF(V-V20LD)479,479,478
003464      478 RDTREF=RDTREF -RDC*(V-V20LD)
003470      479 ALDREF=GS*(1.-VSOVS)/DREFP +2.*V*RDTREF*CAGI+2.*RDTREF*RDTREF*GS*
                1CAGI/DREFP  +2.*DREFP*HS*CAGI+2.*GS*RDTREF*HS*VI*CAGI+RDTREF*VI
                2+GS*RDTREF*RDTREF/(DREFP*VS)
     C*****       CONTROLLER
003515      405 HDSER=V*REH*REH*SINGCL*COSGCL
003521          RDTREF=RDTREF -1.5389E-17*HDSER*COSGAM*COSPSIE
003525          LOD1=ALDREF +C16*(DRAG-DREFP) -C17*(HADOT-RDTREF)
003536          IF(ABS(LOD1)-XLOD)410,410,407
003541      407 LOD1=SIGN(XLOD,LOD1)
     C*****       LATERAL LOGIC
003544      410 IF(ABS(LOD1)-LMN)413,411,411
003550      411 IF(DELAZ*RK2ROL)412,412,413
003553      412 LOD1=SIGN(LMN,LOD1)
003556          GO TO 480
003557      413 IF(ABS(DELAZ) -YB)480,415,415
003563      415 IF(DELAZ*RK2ROL)480,480,416
003566      416 RK2ROL=-RK2ROL
     C*****       ROLL COMMAND AND ANGLE OF ATTACK COMMAND
003567      480 ARG=LOD1/XLOD
003571          IF(ABS(ARG)-1.)482,481,481
003574      481 ARG=SIGN(1.,LOD1)
003577      482 ROLLC=RK2ROL*ACOS(ARG)
003603          IF(V-13600.)484,484,483
003605      483 ALPCMD=34.25
003607          GO TO 492
003607      484 IF(V-9000.)486,486,485
003612      485 ALPCMD=16.836 +.00128*V
003615          GO TO 492
003616      486 IF(V-8000.)487,490,490
003621      487 IF(ABS(ROLLC)-1.22173)489,489,488
003625      488 ROLLC=SIGN(1.22173,ROLLC)
003630      489 IF(V-5000.)491,491,490
003633      490 ALPCMD=4.3516 +.0040563*V -1.543E-7*VS
003641          GO TO 492
003641      491 ALPCMD=5.385 +.003077*V
003644      492 ALPHAC=ALPCMD
003646          PHICM=RTD*ROLLC
     C*****        CONTROL SYSTEM TEN --- AERODYNAMIC AND RCS
     C*****       RCS AERODYNAMIC INTERFERENCE FUNCTIONS
003650      510 QBL=QB
003652          MACHL=MACH
003653          CALL FLIMIT(QBL,0.,20.)
003656          CALL FLIMIT(MACHL,2.,30.)
003661          IF(QB-20.)511,511,512
003664      511 CALL FUNC1(TAB2A ,TABTAB2,QBL)
003667          TAB2=TABTAB2(1)
003671          CALL FUNC1(TAB5A ,TABTAB1,QBL)
003673          TAB5=TABTAB1(1)
003675          GO TO 513
003675      512 CALL FUNC1(TAB22A,TABTAB3,MACHL)
003700          TAB2=TABTAB3(1)
003702          CALL FUNC1(TAB55A,TABTAB3,MACHL)
003704          TAB5=TABTAB3(1)
003706      513 CONTINUE
003706          CALL FUNC1(TAB1A,TABTAB1,QBL)
003711          TAB1=TABTAB1(1)
003713          CALL FUNC1(TAB4A,TABTAB1,QBL)
003715          TAB4=TABTAB1(1)
003717          CALL FUNC1(TAB6A,TABTAB1,QBL)
003721          TAB6=TABTAB1(1)
003723          CALL FUNC1(TAB8A,TABTAB2,QBL)
003725          TAB8=TABTAB2(1)
     C*****       DELSB FUNCTION
003727          IF(VR-10000.)671,670,670
003731      670 DELSB=0.
003732          GO TO 681
003733      671 IF(VR-8000.)673,673,672
003736      672 DELSB=85. +.0425*(8000.-VR)
003742          GO TO 681
003743      673 IF(VR-4500.)675,674,674
003746      674 DELSB=85.
```

```
003750          GO TO 681
003750      675 IF(VR-4000.)677,677,676
003753      676 DELSB=55. +0.06*(VR-4000.)
003757          GO TO 681
003760      677 DELSB=55.
003762      681 CONTINUE
       C*****       DELE CONTROL --- ACTIVE WHEN QB GE 2
003762          DELE=.5*(DELER +DELEL)
003765          DELA=.5*(DELEL -DELER)
       C*****       GO TO SUBROUTINE DATAFUN AND CALCULATE AERO FUNCTIONS THAT
       C*****       DEPEND ON CONTROL DEFLECTIONS.
003770          IF(LDISI(37))805,806
003771      805 IF(INT-1)806,806,807
003774      806 CALL DATAFUN(2)
003776      807 CONTINUE
003776          ERRPHI=PHICM-PHIDG
004000          ABPHI=ABS(ERRPHI)
004002          FGV=RTD*GE*SINPHI*COSTHE*VI
004006          RPRI=RDG-FGV
004010          BATADT=PDG*SINALP -RPRI*COSALP
004014          ERRALP=ALPHAC -ALPDG
004016          IF(ABS(ERRALP) -2.)529,529,528
004021      528 ERRALP=SIGN(2.,ERRALP)
004024      529 CONTINUE
004024          IF(QB -2.)526,527,527
004027      526 ERALI=0.
004030          DELET=DLEC=DELEIC
004033          GO TO 520
       C*****       GKAL FUNCTION
004033      527 IF(VR-12000.)521,522,522
004036      521 IF(VR-10000.)523,523,524
004041      522 GKAL=2.
004043          GO TO 525
004043      523 GKAL=6.
004045          GO TO 525
004045      524 GKAL=6. +.002*(10000.-VR)
004051      525 CONTINUE
004051          ERRALPD=GKAL*ERRALP
004053          IF(MACH -7.)516,516,515
004056      515 SBTERM=0.
004057          GO TO 517
004060      516 SBTERM=.25*(85. -DELSB)
004063      517 IF(INT)531,530,531
004064      530 DELET=DELEIC +SBTERM
004066      531 DLEC=-ERRALPD -.1*ERALI +6.*QDG -SBTERM +DELET
004076      520 CONTINUE
       C*****       RCS SECTION ---CALCULATED ONCE PER INTEGRATION STEP.
004076          ROLNL=ROLN
004100          ROLPL=ROLP
004101          YAWNL=YAWN
004103          YAWPL=YAWP
004104          IF(INT-1)540,540,580
004107      540 IF(QB -10.)541,541,601
       C*****       RCS ---ROLL LOGIC
004112      541 ER1=(BETDG -.05*CON1(2,1))*SINALP
004116          ER2=RPRI/TANALP-PDG
004121          ERROLL=ER2-ER1
004123          ABB=ABS(ERROLL)
004124          IF(ABB-.05)601,601,602
004127      601 LDISO(81)=LDISO(82)=LDISO(103)=LDISO(104)=.F.
004134          ROLN=ROLP=0.
004136          LATROL=0
004137          GO TO 613
004137      602 IF(ABB-.2)603,609,609
004142      603 IF(LATROL)607,601,608
004144      607 IF(ERROLL)605,605,601
004146      608 IF(ERROLL)601,601,606
004150      609 IF(ABB-1.)604,604,610
004153      604 IF(ERROLL)   605,605,606
004155      605 LATROL=-1
004156          LDISO(103)=.T.
004157          LDISO(104)=.F.
004160          LDISO(81)=LDISO(82)=.F.
004162          ROLN=2.
004164          ROLP=0.
004165          GO TO 613
```

```
004165       606 LATROL=1
004166           LDISO(81)=LDISO(82)=.F.
004171           LDISO(104)=.T.
004171           LDISO(103)=.F.
004172           ROLP=2.
004174           ROLN=0.
004175           GO TO 613
004175       610 IF(ERROLL)611,611,612
004177       611 LATROL=-1
004200           LDISO(81)=LDISO(103)=.T.
004203           LDISO(82)=LDISO(104)=.F.
004205           ROLN=4.
004206           ROLP=0.
004207           GO TO 613
004207       612 LATROL=1
004210           LDISO(82)=LDISO(104)=.T.
004213           LDISO(81)=LDISO(103)=.F.
004215           ROLP=4.
004216           ROLN=0.
        C*****        RCS --- PITCH LOGIC
004217       613 IF(QB -20.)614,614,615
004222       614 EP1=QDG -.05*CON1(2,2)
004225           PITERR=ERRALP -EP1
004227           ABB=ABS(PITERR)
004231           IF(ABB-.2)615,615,616
004233       615 LDISO(105)=.F.
004234           LDISO(106)=.F.
004235           PITN=PITP=0.
004237           LATPIT=0
004240           GO TO 622
004240       616 IF(ABB-.25)617,619,619
004243       617 IF(LATPIT)618,615,6175
004245       618 IF(PITERR)620,620,615
004247      6175 IF(PITERR)615,615,621
004251       619 IF(PITERR)620,620,621
004253       620 LATPIT=-1
004254           LDISO(105)=.T.
004255           LDISO(106)=.F.
004256           PITN=2.
004257           PITP=0.
004260           GO TO 622
004260       621 LATPIT=1
004261           LDISO(105)=.F.
004262           LDISO(106)=.T.
004263           PITP=2.
004264           PITN=0.
        C*****        RCS ---YAW LOGIC
004265       622 IF(ALPDG -18.)623,623,624
004270       623 IF(MACH-5.)625,625,624
004273       625 IF(ALPDG-10.)626,542,542
004276       626 ERRYAW=ABYAW=0.
004300           GO TO 629
        C*****        YAW LOGIC ---ALPHA LE 18 AND GE 10,MACH LE 5(EYB EQNS).
004301       542 EY4=3.4*BATADT -4.*CON1(2,6)
004305           ERRYAW=EY4
004306           ABYAW=ABS(ERRYAW)
004310           GO TO 820
        C*****        YAW LOGIC ---ALPHA GT 18 OR MACH GT 5(EYA EQNS).
004310       624 IF(ABPHI -3.)640,640,641
004313       640 ERPHLM=0.
004314           GO TO 642
004315       641 ERPHLM=ABPHI -3.
004317           IF(ERPHLM -10.)586,586,585
004322       585 ERPHLM=10.
004324       586 ERPHLM=SIGN(ERPHLM,ERRPHI)
004327       642 RLL=RPRI -CON1(2,3)
004331           EY2=2.*RPRI +FQBAR*RLL
004335           IF(QB -20.)544,544,543
004340       543 BLL=0.
004341           GO TO 545
004342       544 BLL=COSALP*(BETDG -.05*CON1(2,4))
004346       545 ERRYAW=ERPHLM*SINALP -EY2 +BLL
004352           ABYAW=ABS(ERRYAW)
004354           IF(ABYAW-.05)629,629,630
004356       629 LATYAW=0
```

```
004357          LDISO(107)=LDISO(108)=LDISO(83)=LDISO(84)=.F.
004364          LDISO(85)=LDISO(86)=LDISO(87)=LDISO(88)=.F.
004371          YAWN=YAWP=0.
004373          GO TO 639
004373      630 IF(ABYAW-.20)631,635,635
004376      635 IF(ABYAW -.5)632,632,810
004401      631 IF(LATYAW)628,629,6295
004403      628 IF(ERRYAW)633,633,629
004405     6295 IF(ERRYAW)629,629,634
004407      632 IF(ERRYAW)633,633,634
004411      633 LATYAW=-1
004412          LDISO(107)=.T.
004413          LDISO(108)=LDISO(83)=LDISO(84)=.F.
004417          LDISO(85)=LDISO(86)=LDISO(87)=LDISO(88)=.F.
004423          YAWN=1.
004425          YAWP=0.
004426          GO TO 639
004426      634 LATYAW=1
004427          LDISO(108)=.T.
004430          LDISO(107)=LDISO(83)=LDISO(84)=.F.
004434          LDISO(85)=LDISO(86)=LDISO(87)=LDISO(88)=.F.
004440          YAWP=1.
004442          YAWN=0.
004443          GO TO 639
004443      810 IF(ABYAW -1.)636,636,811
004446      811 IF(QB-20.)636,812,812
004451      812 IF(ABYAW -1.5)813,813,816
004454      813 IF(ERRYAW)814,814,815
004456      814 LATYAW=-1
004457          LDISO(107)=LDISO(85)=.T.
004462          LDISO(86)=LDISO(87)=LDISO(88)=.F.
004465          LDISO(108)=LDISO(83)=LDISO(84)=.F.
004471          YAWN=3.
004472          YAWP=0.
004473          GO TO 639
004473      815 LATYAW=1
004474          LDISO(108)=LDISO(86)=.T.
004477          LDISO(107)=LDISO(83)=LDISO(84)=.F.
004502          LDISO(85)=LDISO(87)=LDISO(88)=.F.
004506          YAWP=3.
004507          YAWN=0.
004510          GO TO 639
004510      816 IF(ERRYAW)817,817,818
004512      817 LATYAW=-1
004513          LDISO(107)=LDISO(87)=.T.
004516          LDISO(108)=LDISO(83)=LDISO(84)=.F.
004521          LDISO(85)=LDISO(86)=LDISO(88)=.F.
004525          YAWN=4.
004526          YAWP=0.
004527          GO TO 639
004527      818 LATYAW=1
004530          LDISO(108)=LDISO(88)=.T.
004533          LDISO(107)=LDISO(83)=LDISO(84)=.F.
004536          LDISO(85)=LDISO(86)=LDISO(87)=.F.
004542          YAWP=4.
004543          YAWN=0.
004544          GO TO 639
004544      636 IF(ERRYAW)637,637,638
004546      637 LATYAW=-1
004547          LDISO(83)=LDISO(107)=.T.
004552          LDISO(84)=LDISO(108)=.F.
004554          LDISO(85)=LDISO(86)=LDISO(87)=LDISO(88)=.F.
004560          YAWN=2.
004562          YAWP=0.
004563          GO TO 639
004563      638 LATYAW=1
004564          LDISO(83)=LDISO(107)=.F.
004567          LDISO(84)=LDISO(108)=.T.
004571          LDISO(85)=LDISO(86)=LDISO(87)=LDISO(88)=.F.
004575          YAWP=2.
004577          YAWN=0.
004600          GO TO 639
004600      820 IF(ABYAW -.5)629,629,821
004603      821 IF(ABYAW -1.)631,822,822
004606      822 IF(ABYAW-1.5)632,632,823
```

```
004611       823 IF(ABYAW -2.)636,636,824
004614       824 IF(ABYAW-2.5)813,813,816
004617       639 CONTINUE
          C*****     UP-DOWN COUNTER
004617           NUDCTR=0
004620           IF(YAWN-YAWNL-.1)831,831,830.
004624       830 NUDCTR=YAWN-YAWNL+.1
004630           GO TO 833
004631       831 IF(YAWP-YAWPL-.1)833,833,832
004635       832 NUDCTR=YAWPL-YAWP-.1
004641       833 IF(ROLN-ROLNL-.1)835,835,834
004645       834 NUDCTR=NUDCTR+ROLNL-ROLN-.1
004653           GO TO 860
004653       835 IF(ROLP-ROLPL-.1) 860,860,836
004657       836 NUDCTR=NUDCTR+ROLP-ROLPL+.1
004665       860 CONTINUE
004665           IF(VR -10000.)861,861,862
004670       861 GNUD=.1
004672           GO TO 863
004672       862 GNUD=ROQB
004674       863 DELANUD=DELANUD +GNUD*NUDCTR
004700           IF(ABS(DELANUD) -5.)865,865,864
004703       864 DELANUD=SIGN(5.,DELANUD)
004706       865 CONTINUE
004706           EMX=8250.*((ROLP -ROLN)*TAB1 +(YAWP -YAWN)*TAB2)
004715           EMY=28267.5*(PITP*TAB3 -PITN*TAB4 +(YAWP +YAWN)*TAB5 +(ROLP +ROLN)
                1*TAB6)
004731           EMZ=28675.5*((YAWP -YAWN)*TAB7 +(ROLP -ROLN)*TAB8)
004737           TX=TY=TZ=0.
004742       575 TROLN=TROLN +ROLN*H
004745           TROLP=TROLP +ROLP*H
004750           TPITN=TPITN +PITN*H
004752           TPITP=TPITP +PITP*H
004755           TYAWN=TYAWN +YAWN*H
004760           TYAWP=TYAWP +YAWP*H
004762           FUELUSE=TISPH*(TROLN +TROLP +TPITN +TPITP +TYAWN +TYAWP)
004771       580 CONTINUE
004771           IF(QB -2.)665,550,550
004774       550 IF(ALPDG -18.)551,551,657
004777       551 IF(MACH-5.)658,658,657
          C*****     DELA CONTROL --- ALPDG LE 18 AND MACH LE 5
005002       658 IF(ABPHI-20.)6585,6585,6586
005005      6585 ERRPHI2=ERRPHI
005007           GO TO 6587
005007      6586 ERRPHI2=SIGN(20.,ERRPHI)
005012      6587 CONTINUE
005012           EA2=.25*ERRPHI2 -PDG*COSALP -RPRI*SINALP
005020           DELAC2=EA2 +.1*EA2I
005023           DELAC1=0.
          C*****     DELR CONTROL---ALPHA LT 18 AND MACH LT 5
005024           IF(VR -2000.)554,554,555
005026       554 GKR=2.
005030           GO TO 556
005030       555 IF(VR -2500.)519,518,518
005033       518 GKR=4.
005035           GO TO 556
005035       519 GKR=2. +.004*(VR -2000.)
005041       556 EY5=GKR*(3.4*BATADT -4.*CON1(2,6))
005046           DELRC=-EY5 -.05*EY5I
005051           DIF=DELRC -DELR
005053           IF(ABS(DIF) -RATLIMR)891,891,890
005056       890 DELRC=DELR +SIGN(RATLIMR,DIF)
005062       891 CONTINUE
005062           ABB=ABS(DELRC)
005064           IF(ABB-22.8)6576,6576,6575
005066      6575 DELRC=SIGN(22.8,DELRC)
005071      6576 CONTINUE
005071           HMR=QB*7331.*(CHR*DELR +CHB*BETDG)
005077           IF(ABS(HMR) -9.9E5)883,882,882
005102       882 DRMX=0.
005103           GO TO 881
005104       883 DRMX=21.*SQRT(1. -HMR/9.9E5)
005113           IF(DRMX -10.)881,881,880
005115       880 DRMX=10.
005117       881 RLIMITD(1)=-DRMX
```

```
005121          RLIMITD(2)= DRMX
005122          GO TO 659
       C*****        DELA CONTROL --- ALPDG GT 18 OR MACH GT 5
005122      657 IF(QB -20.)561,561,560
005125      560 EDELA1=0.
005126          GO TO 562
005127      561 EDELA1=BETDG -.05*CON1(2,5)
005132      562 CONTINUE
005132          EDELA2=RPRI/TANALP -PDG -EDELA1*SINALP
005137          ABEDA2=ABS(EDELA2)
005141          IF(ABEDA2-.05)643,643,644
005143      643 ERMP=0.
005144          LATDLA=0
005145          GO TO 648
005145      644 IF(ABEDA2-.2)645,646,646
005150      645 IF(LATDLA)6455,643,6456
005152     6455 IF(EDELA2)647,647,643
005154     6456 IF(EDELA2)643,643,6475
005156      646 IF(EDELA2)647,647,6475
005160      647 LATDLA=-1
005161          GO TO 6485
005162     6475 LATDLA=1
005163     6485 ERMP=EDELA2
005165      648 CONTINUE
005165          IF(QB -10.)566,566,565
005170      565 IF(QB -30.)568,567,567
005173      566 GKDA=10.
005175          GO TO 569
005175      567 GKDA=2.5
005177          GO TO 569
005177      568 GKDA=10. -(QB-10.)*SLPAQ
005203      569 CONTINUE
005203      687 DELAC1=GKDA*ERMP +DELANUD
005206          DELAC2=0.
005207      659 DELAC=DELAC1 + DELAC2
005211          CALL LIMIT(DLEC,-35.,20.)
005214          IF(DELE -5.)870,871,871
005217      870 IF(DELE +20.)872,872,873
005222      871 ABDA=20. -DELE
005224          GO TO 874
005225      872 ABDA=35. +DELE
005227          GO TO 874
005230      873 ABDA=15.
005232      874 CONTINUE
005232      664 IF(ABS(DELAC) -ABDA)665,665,666
005236      666 DELAC=SIGN(ABDA,DELAC)
005241      665 CONTINUE
005241          DLECL=DLEC +DELAC
005243          DIF=DLECL -DELEL
005245          IF(ABS(DIF) -RATELIM)886,886,885
005250      885 DLECL=DELEL+SIGN(RATELIM,DIF)
005254      886 CONTINUE
005254          DLECR=DLEC -DELAC
005256          DIF=DLECR -DELER
005260          IF(ABS(DIF) -RATELIM)888,888,887
005263      887 DLECR=DELER+SIGN(RATELIM,DIF)
005267      888 CONTINUE
005267          CALL LIMIT(DLECL,-35.,20.)
005272          CALL LIMIT(DLECR,-35.,20.)
005275          HMEC=19047.*QB*GCHE
005300          HMER=HMEC*CHER
005302          HMEL=HMEC*CHEL
005304          IF(ABS(HMER) -1.17E6)893,892,892
005307      892 DEMXR=0.
005310          GO TO 876
005311      893 DEMXR=32.*SQRT(1. -HMER/1.17E6)
005320          IF(DEMXR -20.)876,876,875
005322      875 DEMXR=20.
005324      876 RLIMITR(1)=-DEMXR
005326          RLIMITR(2)= DEMXR
005327          IF(ABS(HMEL) -1.17E6)896,895,895
005332      895 DEMXL=0.
005333          GO TO 878
```

```
005334          896 DEMXL=32.*SQRT(1. -HMEL/1.17E6)
005343              IF(DEMXL -20.)878,878,877
005345          877 DEMXL=20.
005347          878 RLIMITL(1)=-DEMXL
005351              RLIMITL(2)= DEMXL
005352              THECDOT=VYRG*REHI/COSGCL +OMEGA
005355              TEMP1=THECDOT*COSGCL
005356              TEMP2=-VXRG*REHI
005360              TEMP3=-THECDOT*SINGCL
005362              BIGP=P-EL1*TEMP1-EL2*TEMP2-EL3*TEMP3
005371              BIGQ=Q-EM1*TEMP1-EM2*TEMP2-EM3*TEMP3
005400              BIGR=R-EN1*TEMP1-EN2*TEMP2-EN3*TEMP3
005407              PHIDOT=BIGP+TANTHE*(BIGQ*SINPHI+BIGR*COSPHI)
005414              THEDOT=BIGQ*COSPHI-BIGR*SINPHI
005420              PSIDOT=(BIGQ*SINPHI+BIGR*COSPHI)/COSTHE
005424              CCD=CCD1 +GCDE *CDE  +GCDSB*CDSB +GCDBF*CDBF
005433              CCL=CCL1 +GCLE *CLE  +GCLSB*CLSB +GCLBF*CLBF
005442              AX=QSO*(CCL*SINALP -CCD*COSALP)  +TX*OMASSI
005450              AY=QSO*((GCYB*CYB+GCYBSB*CYBSB)*BETDG +GCYDR*CYDR*DELR +GCYDA*CYDA
                   1*DELA) +TY*OMASSI
005465              AZ=-QSO*(CCL*COSALP +CCD*SINALP)+TZ*OMASSI
005474              ACCEL=SQRT(AX*AX +AY*AY +AZ*AZ)*PTS
005504              FX=OMASS*AX -TX
005507              FY=OMASS*AY -TY
005512              FZ=OMASS*AZ -TZ
005515              VXDRG=EL1*AX+EM1*AY+EN1*AZ
005522              VYDRG=EL2*AX+EM2*AY+EN2*AZ
005530              VZDRG=EL3*AX+EM3*AY+EN3*AZ
005536              VXDOT=-VXDRG*SINGCL*COSLONI -VYDRG*SINLONI -VZDRG*COSGCL*COSLONI
                   1+GXI
005546              VYDOT=-VXDRG*SINGCL*SINLONI +VYDRG*COSLONI -VZDRG*COSGCL*SINLONI
                   1+GYI
005556              VZDOT= VXDRG*COSGCL -VZDRG*SINGCL +GZI
005562              BTV=HSPAN*VI
005564              CTV=HCHORD*VI
005566              QSSPAN=QS*SPAN
005570              QSCHORD=QS*CHORD
005572              L=QSSPAN*(BTV*(GCLP*CLP*P+GCLR*CLR*R) +BETDG*(GCLB*CLB+GCLBSB*
                   1CLBSB) +GCLDA*CLDA*DELA +GCLDR*CLDR*DELR) +EMX -(YOFF*FZ-ZOFF*FY)
005622              M=QSCHORD*(GCM*CM+CMV +CTV*GCMQ*CMQ*Q +GCME*CME +GCMSB*CMSB+GCMBF*
                   1CMBF) +EMY -(ZOFF*FX-XOFF*FZ)
005646              N=QSSPAN*((GCNB*CNB +GCNBSB*CNBSB)*BETDG +BTV*(GCNP*CNP*P +GCNR*
                   1CNR*R) +GCNDR*CNDR*DELR +GCNDA*CNDA*DELA) +EMZ -(XOFF*FY -YOFF*FX)
005676              ROT1=EX*P -EXY*Q -EXZ*R
005704              ROT2=-EXY*P +EY*Q -EYZ*R
005711              ROT3=-EXZ*P -EYZ*Q +EZ*R
005717              ROTL=L -Q*ROT3 +R*ROT2
005723              ROTM=M -R*ROT1 +P*ROT3
005730              ROTN=N -P*ROT2 +Q*ROT1
005734              PDOT=DETI*(TERMA*ROTL +TERMB*ROTM +TERMC*ROTN)
005743              QDOT=DETI*(TERMB*ROTL +TERMD*ROTM +TERME*ROTN)
005752              RDOT=DETI*(TERMC*ROTL +TERME*ROTM +TERMF*ROTN)
005761              IF(INT.GT.1)GO TO 90005
     C*****         STORE PAST VALUES FOR CONVOLUTION INTEGRATION.
005764              CON1(4,1)=CON1(4,4)=CON1(4,5)=BATADT
005770              CON1(4,2)=QDOT*RTD
005772              CON1(4,6)=(AY-CON1(3,6))*HI
005775              CON1(4,3)=(RPRI -CON1(3,3))*HI
005777              CON1(3,1)=CON1(3,4)=CON1(3,5)=BETDG
006004              CON1(3,2)=QDG
006005              CON1(3,3)=RPRI
006006              CON1(3,6)=AY
006007              U1=PASTL
006011              U2=PASTR
006012              U3=PASTD
006014              PASTL=DLECL
006015              PASTR=DLECR
006016              PASTD=DELRC
     C*****         RECORDER OUTPUT AND TABLE UPDATES
006017              DAC( 1)=HA*2.E-6
006021              DAC( 2)=V*2.E-5
006023              DAC( 3)=ACCEL*.2
006025              DAC( 4)=(ALPDG +10.)*.02
006030              DAC( 5)=ERRALP*.5
006032              DAC( 6)=(PHIDG +100.)*.005
```

```
006035          DAC( 7)=ERRPHI*.05
006037          DAC( 8)=TRANGE*.004
006041          DAC( 9)=(DELA+25.)*.02
006044          DAC(10)=DELE*.025
006046          DAC(11)=DELR*.05
006047          DAC(12)=PDG*.125
006051          DAC(13)=QDG
006053          DAC(14)=RDG*.125
006054          DAC(15)=BETDG*.25
006056          DAC(16)=(GAMDG+20.)*.04
006061          TABLE( 80)=T
006063          TABLE( 81)=QB
006064          TABLE( 82)=HA
006066          TABLE( 83)=VR
006067          TABLE(84)=GAMMAA
006071          TABLE(85)=PSIA
       C*****        PUT A NEW DISPLAY OUT TO THE CONTROL CONSOLE DIGITAL DECIMAL
       C*****        DISPLAY AND TYPE OUT ANY CHANGES MADE TO VARIABLES.
006072          IF(LDISI(22))CALL SCANNER(JSCAN)
006075          CALL DSPLAY
006076          IF(LDISI(17))GO TO 90050
006100          IF(VARCHNG) CALL TYPEVAR
006102          IF(ENABLE.AND.TYPESW) CALL TYPEVAR
006105          ENABLE=.NOT.TYPESW
006107          IF(AUTOTYP.AND.LDISI(14)) CALL TYPEVAR
006112    90050 CALL RTMODE
006113    90001 CONTINUE
006113          IF(T-CNGRATE)90061,90061,90060
006116    90060 IF(AMOD(T +.0025,OFTEN) -.005)90061,90061,90005
006124    90061 CALL RECORD
006125    90005 CONTINUE
006125          IF(LDISI(35))CALL RTPAUSE(LDISI(40),LDISO(113),LDISI(44))
       C*****        INTEGRATE NON-LINEAR DIFFERENTIAL EQUATIONS.
006131          CALL IGRATE1
006132          IF(.NOT.LDISI(34))GO TO 90065
006134          IF(INT.GT.1)INT=1
006137    90065 CONTINUE
006137          IF(INT.GT.1)GO TO 90006
006143          IF(LDISO(111))752,750
006144      750 IF(HA-70000.)751,751,752
006147      751 LDISO(111)=.T.
006150          CALL RECORD
006151      752 IF(LDISO(112))755,753
006153      753 IF(V-1500.)754,754,755
006156      754 LDISO(112)=.T.
006157          CALL RECORD
006160      755 IF(LDISO(113))760,756
006162      756 IF(HA)757,757,760
006164      757 LDISO(113)=.T.
006165      760 CONTINUE
       C*****        INTEGRATE LINEAR DIFFERENTIAL EQUATIONS(CONTROL SYSTEM TRANS-
       C*****        FER FUNCTIONS)WITH CONVO1 AND MODLAG.
006165          CALL CONVO1
006166          CALL MODLAG(PASTL,DELEL,10.,RLIMITL,OLIMITE,U1,X1)
006175          CALL MODLAG(PASTR,DELER,10.,RLIMITR,OLIMITE,U2,X2)
006204          CALL MODLAG(PASTD,DELR ,20.,RLIMITD,OLIMITR,U3,X3)
006213          ELERL=(DELEL -X1)*HI
006216          ELERR=(DELER -X2)*HI
006221          RUDR =(DELR -X3)*HI
006224          X1=DELEL
006225          X2=DELER
006226          X3=DELR
006230          CALL RECYCLE
006231    90014 CONTINUE
       C*****        THIS IS THE CODE EXECUTED WHEN THE PRINT SWITCH IS DEPRESSED.
006231          NRUN=NRUN +1
006233          WRITE(MF, 1)NRUN
006240          WRITE(MF, 4)LOGIC(1),LOGIC(2),LDISI(33),LDISI(34),LDISI(37)
006256          WRITE(MF, 2)(INTEG(II),II=3,8)
006270          WRITE(MF,10)(TABLE(II),II=1,100)
006302          WRITE(MF, 3)
```

106

```
006306      90030 CALL PLAYBAK(90032S)
006310            WRITE(MF,11)T,HA,RN,GCLATDG,RLONDG,V,MACH,ISLECT
006334            WRITE(MF,10)DELANUD,HADOT,DRAG,HMER,HMEL,HMR,DELEL,DELER
006360            WRITE(MF,13)NUDCTR,GAMDG,RPRI,ELERR,ELERL,RUDR,VBAR,VINERT
006404            WRITE(MF,10)ACCEL,QB,PDG,QDG,RDG,THEDG,PSIDG,TRANGE
006430            WRITE(MF,10)BETDG,ERRPHI,PDOT,QDOT,RDOT,PITERR,ERROLL,ERRYAW
006454            WRITE(MF,10)ALPDG,PHIDG,DELE,DELA,DELR,DELSB,DELBF
006476            WRITE(MF,10)ALPHAC,PHICM,DLEC,DELAC,DELRC,PSIEDG,VR
006520            WRITE(MF,12)TROLN,TROLP,TPITN,TPITP,TYAWN,TYAWP,FUELUSE
006542            GO TO 90030
006543      90032 CONTINUE
006543            CALL APRINT
006544      90015 CONTINUE
            C*****      THIS IS THE CODE EXECUTED WHEN THE READ SWITCH IS DEPRESSED.
006544            CALL RELEASE
006545            CALL READOUT(6,NOUT,T,HA,GCLATDG,RLONDG,V,RN)
006555            CALL READOUT(6,NOUT,MACH,ISLECT,ACCEL,QB,PDG,QDG)
006565            CALL READOUT(6,NOUT,RDG,THEDG,PSIDG,TRANGE,BETDG,GAMDG)
006575            CALL READOUT(6,NOUT,PDOT,QDOT,RDOT,PITERR,ERROLL,ERRYAW)
006605            CALL READOUT(6,NOUT,ALPDG,PHIDG,DELE,DELA,DELR,DELSB)
006615         .  CALL READOUT(6,NOUT,DELBF,ALPHAC,PHICM,DLEC,DELAC,DELRC)
006625            CALL READOUT(3,NOUT,NUDCTR,PSIEDG,VR)
006631            CALL READOUT(1,NOUT,FUELUSE)
006634            CALL READOUT(6,NOUT,TROLN,TROLP,TPITN,TPITP,TYAWN,TYAWP)
006644            CALL READOUT(6,NOUT,DELANUD,RPRI,DELEL,DELER,VBAR,VINERT)
006654            CALL READOUT(6,NOUT,HMER,HMEL,HMR,RUDR,ELERL,ELERR)
006664            CALL READOUT(3,NOUT,DRAG,HADOT,ERRPHI)
006670            CALL AREAD
006671      90004 CONTINUE
            C*****      THIS IS THE CODE EXECUTED WHEN THE TERMINATE SWITCH IS DE-
            C*****      PRESSED.
006671            CALL ATERM
006672      90017 CALL SYNCH
006673            CALL RTMODE
006674          1 FORMAT(/2X,*RUN NUMBER *,I3//2X,*THESE ARE THE VALUES IN TABLES FO
                  1R THIS RUN.*)
006674          4 FORMAT(20(2X,L3))
006674          2 FORMAT(20(2X,I3))
006674          3 FORMAT(//2X,1HT,15X,2HHA,14X,2HRN,14X,7HGCLATDG,9X,6HRLONDG,10X,
                  21HV,15X,4HMACH,12X,6HISLECT/
                  12X,7HDELANUD,9X,5HHADOT,11X,4HDRAG,12X,4HHMER,12X,4HHMEL,12X,3HHMR
                  2,13X,5HDELEL,11X,5HDELER/
                  12X,6HNUDCTR,10X,5HGAMDG,11X,4HRPRI,12X,5HELERR,11X,5HELERL,11X,
                  24HRUDR,12X,4HVBAR,12X,6HVINERT/
                  12X,5HACCEL,11X,2HQB,14X,3HPDG,13X,3HQDG,13X,3HRDG,13X,5HTHEDG,11X,
                  25HPSIDG,11X,6HTRANGE/
                  12X,5HBETDG,11X,6HERRPHI,10X,4HPDOT,12X,4HQDOT,12X,4HRDOT,12X,
                  26HPITERR,10X,6HERROLL,10X,6HERRYAW/
                  12X,5HALPDG,11X,5HPHIDG,11X,4HDELE,12X,4HDELA,12X,4HDELR,12X,
                  25HDELSB,11X,5HDELBF/
                  12X,6HALPHAC,10X,5HPHICM,11X,4HDLEC,12X,5HDELAC,11X,5HDELRC,11X,
                  26HPSIEDG,10X,2HVR/
                  12X,5HTROLN,11X,5HTROLP,11X,5HTPITN,11X,5HTPITP,11X,5HTYAWN,11X,
                  25HTYAWP,11X,7HFUELUSE/)
006674         10 FORMAT(8(2X,E14.7))
006674         11 FORMAT(7(2X,E14.7),2X,I14)
006674         12 FORMAT(7(2X,E14.7)/)
006674         13 FORMAT(2X,I14,7(2X,E14.7))
006674      90034 FORMAT(1X,*JOB,43,28800,75000,1024.  C1202 15319T  100857  L.F.ROW
                  1ELL    ROOM 2124    PROGRAM ARFDS*)
006674            END
```

CROSS REFERENCE MAP-ARFDS

PROGRAM LENGTH INCLUDING I/O BUFFERS
011321

STATEMENT FUNCTION REFERENCES

LOCATION	GEN TAG	SYM TAG	REFERENCES

STATEMENT NUMBER REFERENCES

LOCATION	GEN TAG	SYM TAG	REFERENCES					
007235	C00337	1	006232					
007251	C00353	2	006256					
007254	C00356	3	006302					
007246	C00350	4	006240					
007422	C00524	10	006270	006334	006404	006430	006454	006476
007425	C00527	11	006310					
007431	C00533	12	006520					
007434	C00536	13	006360					
000752	L00356	53	000750					
000760	L00363	54	000751					
000767	L00367	55	000757					
001271	L00535	60	NONE					
001275	L00536	61	001267	001270				
001301	L00537	62	NONE					
001305	L00540	63	001277	001300				
001311	L00541	64	NONE					
001315	L00542	65	001307	001310				
002115	L01022	300	NONE					
002117	L01023	301	002114					
002130	L01027	302	NONE					
002133	L01030	303	002127					
002137	L01032	304	002132					
002142	L01033	305	002136					
002145	L01034	306	NONE					
002147	L01036	307	002127	002131	002132			
002151	L01037	308	002143	002144	002146			
002100	L01016	309	002066	002077				
002226	L01061	310	NONE					
002232	L01062	311	NONE					
002234	L01063	312	002225	002230	002231			
002067	L01014	313	002065					
002072	L01015	314	002070					
002252	L01074	315	002255					
002274	L01111	316	002276					
002327	L01134	321	002325					
002333	L01135	322	002326					
002343	L01143	325	002342					
002347	L01144	326	NONE					
002353	L01146	327	NONE					
002356	L01147	328	002342	002345	002346	002351	002352	
002360	L01150	330	NONE					
002364	L01151	331	002357					
002370	L01153	335	002366					
002376	L01155	336	NONE					
002401	L01156	337	002375					
002403	L01160	338	002400					
002405	L01161	339	002374	002377	002400	002402		
002414	L01163	340	NONE					
002416	L01164	341	002412	002413				
002421	L01165	342	NONE					
002423	L01166	345	002367	002417	002420			
002431	L01170	350	NONE					
002433	L01172	351	002427	002430				
002436	L01173	352	NONE					
002442	L01175	353	002434	002435				
002445	L01176	354	NONE					
002451	L01200	355	002443	002444				
002457	L01202	360	002432	002441	002450			
002462	L01203	361	NONE					
002464	L01204	362	002460	002461				
002542	L01234	365	NONE					
002552	L01237	366	002541					
002556	L01240	367	NONE					

002573	L01245	370	002546	002554	002555	
002577	L01247	371	003222			
002607	L01251	372	002606			
002635	L01260	380	002547	002576		
002653	L01265	381	002652			
002660	L01266	382	002575	002576	002656	
002661	L01267	383	003155			
002664	L01271	384	NONE			
002667	L01273	385	002663			
003226	L01404	390	002550	002606	003224	
003244	L01411	391	003243			
003255	L01414	392	003254			
003263	L01417	393	003254			
003266	L01420	394	NONE			
003272	L01421	395	003264	003265		
003327	L01434	396	NONE			
003333	L01435	397	NONE			
003340	L01436	398	003325	003326	003331	003332
003343	L01437	399	NONE			
003303	L01425	400	002551	002652	003243	
003345	L01440	401	003341	003342		
003515	L01477	405	002634	003225	003302	
003541	L01503	407	NONE			
003544	L01504	410	002572	003537	003540	
003550	L01505	411	003546			
003553	L01506	412	003551			
003557	L01510	413	003547	003552		
003563	L01511	415	003561			
003566	L01512	416	NONE			
002673	L01275	420	NONE			
002675	L01276	421	002672	003126		
002706	L01302	422	NONE			
002716	L01305	423	002704	002705		
002720	L01306	425	002715			
002723	L01307	426	002721			
002767	L01321	429	002765			
003014	L01327	430	002766			
003050	L01335	435	002722	003013		
003057	L01340	436	003056			
003062	L01341	437	NONE			
003063	L01342	438	003060	003061		
003111	L01347	440	003106			
003120	L01352	441	NONE			
003122	L01353	442	003117			
003127	L01357	445	003107			
003143	L01362	446	NONE			
003145	L01364	447	003142			
003150	L01365	448	NONE			
003152	L01366	449	003141	003144	003146	003147
003156	L01372	450	002657	003056	003110	
003223	L01403	457	003222			
003323	L01433	460	003371			
003365	L01445	465	003363			
003405	L01454	466	NONE			
003407	L01455	467	003403	003404		
003372	L01451	470	003364			
003420	L01460	471	NONE			
003423	L01462	472	003417			
003426	L01463	473	NONE			
003431	L01465	475	003424	003425		
003453	L01471	476	003452			
003461	L01474	477	003452			
003464	L01475	478	NONE			
003470	L01476	479	003462	003463		
003567	L01513	480	003556	003562	003564	003565
003574	L01515	481	003573			
003577	L01516	482	003573			
003605	L01521	483	NONE			
003607	L01523	484	003604			
003612	L01524	485	NONE			
003616	L01526	486	003610	003611		
003621	L01527	487	NONE			
003625	L01530	488	NONE			
003630	L01531	489	003623	003624		
003633	L01532	490	003617	003620		

003641	L01534	491	003631	003632				
003644	L01535	492	003606	003615	003640			
003650	L01537	510	002071	002077				
003664	L01546	511	003662					
003675	L01555	512	003663					
003706	L01563	513	003674					
004056	L01647	515	NONE					
004060	L01651	516	004054	004055				
004063	L01652	517	004057					
005033	L02210	518	005031					
005035	L02212	519	005032					
004076	L01655	520	004032					
004036	L01637	521	NONE					
004041	L01640	522	004034	004035				
004043	L01642	523	004037	004040				
004045	L01644	524	004040					
004051	L01645	525	004042	004044				
004027	L01633	526	NONE					
004033	L01636	527	004025	004026				
004021	L01631	528	NONE					
004024	L01632	529	004017	004020				
004064	L01653	530	NONE					
004066	L01654	531	004063					
004107	L01662	540	004105					
004112	L01663	541	004110					
004301	L01773	542	004274	004275				
004340	L02011	543	NONE					
004342	L02013	544	004336	004337				
004346	L02014	545	004341					
004774	L02173	550	004772					
004777	L02174	551	004775					
005026	L02205	554	005025					
005030	L02207	555	005025					
005041	L02213	556	005027	005034				
005125	L02237	560	NONE					
005127	L02241	561	005123	005124				
005132	L02242	562	005126					
005170	L02262	565	NONE					
005173	L02263	566	005166	005167				
005175	L02265	567	005171	005172				
005177	L02267	568	005172					
005203	L02270	569	005174	005176				
004742	L02163	575	NONE					
004771	L02172	580	004106					
004322	L02004	585	NONE					
004324	L02005	586	004320	004321				
004127	L01670	601	004111	004125	004142	004145	004146	004147
004137	L01674	602	004126					
004142	L01675	603	NONE					
004153	L01701	604	004151					
004155	L01702	605	004144	004145	004153			
004165	L01711	606	004147	004154				
004144	L01676	607	NONE					
004146	L01677	608	004143					
004150	L01700	609	004140	004141				
004175	L01720	610	004152					
004177	L01721	611	004175					
004207	L01727	612	004176					
004217	L01734	613	004136	004164	004174	004206		
004222	L01735	614	004220					
004233	L01741	615	004221	004232	004243	004246	004247	004250
004240	L01746	616	004232					
004243	L01747	617	NONE					
004245	L01750	618	NONE					
004251	L01752	619	004241	004242				
004253	L01753	620	004245	004246	004251			
004260	L01761	621	004250	004252				
004265	L01766	622	004237	004257				
004270	L01767	623	004266					
004310	L01777	624	004267	004272				
004273	L01770	625	004271					
004276	L01771	626	NONE					
004403	L02027	628	NONE					
004356	L02017	629	004300	004355	004401	004404	004405	004406
			004601	004602				

004373	L02024	630	004355					
004401	L02026	631	004375	004605				
004407	L02031	632	004377	004400	004607	004610		
004411	L02032	633	004403	004404	004407			
004426	L02041	634	004406	004410				
004376	L02025	635	004374					
004544	L02111	636	004444	004445	004450	004612	004613	
004546	L02112	637	004544					
004563	L02121	638	004545					
004617	L02135	639	004372	004425	004442	004472	004507	004526
			004543	004562	004577			
004313	L02000	640	004311					
004315	L02002	641	004312					
004327	L02006	642	004314					
005143	L02245	643	005142	005150	005153	005154	005155	
005145	L02250	644	005142					
005150	L02251	645	NONE					
005156	L02254	646	005146	005147				
005160	L02255	647	005152	005153	005156			
005165	L02261	648	005144					
005122	L02236	657	004776	005001				
005002	L02175	658	005000					
005207	L02272	659	005121					
005232	L02304	664	005224	005227				
005241	L02306	665	004773	005234	005235			
005236	L02305	666	NONE					
003731	L01600	670	003730					
003733	L01602	671	003730					
003736	L01603	672	NONE					
003743	L01605	673	003734	003735				
003746	L01606	674	003744					
003750	L01610	675	003745					
003753	L01611	676	NONE					
003760	L01613	677	003751	003752				
003762	L01614	681	003732	003742	003747	003757		
005203	L02270	687	005174	005176				
001755	L00761	701	002007					
002006	L00770	702	NONE					
002010	L00772	704	002004	002005				
002033	L01000	706	001734	001736				
002037	L01002	707	002032					
002061	L01011	709	002036					
002065	L01013	710	NONE					
006144	L02522	750	NONE					
006147	L02523	751	006145					
006151	L02526	752	006143	006146				
006153	L02527	753	NONE					
006156	L02530	754	006154					
006160	L02533	755	006151	006152	006155			
006162	L02534	756	NONE					
006164	L02535	757	006162					
006165	L02536	760	006160	006161	006163			
001722	L00745	800	001721					
001725	L00746	801	001721	001723				
001727	L00750	802	001724					
003771	L01617	805	003770					
003774	L01620	806	003770	003772				
003776	L01622	807	003773					
004443	L02050	810	004400					
004446	L02051	811	NONE					
004451	L02052	812	004447					
004454	L02053	813	004452	004615	004616			
004456	L02054	814	004454					
004473	L02063	815	004455					
004510	L02072	816	004453	004616				
004512	L02073	817	004510					
004527	L02102	818	004511					
004600	L02130	820	004307					
004603	L02131	821	NONE					
004606	L02132	822	004604					
004611	L02133	823	NONE					
004614	L02134	824	NONE					
004624	L02137	830	NONE					
004631	L02141	831	004622	004623				

004635	L02142	832	NONE				
004641	L02143	833	004630	004633	004634		
004645	L02144	834	NONE				
004653	L02146	835	004643	004644			
004657	L02147	836	NONE				
004665	L02150	860	004652	004655	004656		
004670	L02151	861	004666				
004672	L02153	862	004667				
004674	L02154	863	004671				
004703	L02156	864	NONE				
004706	L02157	865	004701	004702			
005217	L02276	870	NONE				
005222	L02277	871	005215	005216			
005225	L02301	872	005220	005221			
005230	L02303	873	005221				
005232	L02304	874	005224	005227			
005322	L02333	875	NONE				
005324	L02334	876	005310	005321			
005345	L02344	877	NONE				
005347	L02345	878	005333	005344			
005115	L02232	880	NONE				
005117	L02233	881	005103	005114			
005102	L02225	882	005100				
005104	L02227	883	005101				
005250	L02311	885	NONE				
005254	L02312	886	005247				
005263	L02315	887	NONE				
005267	L02316	888	005262				
005056	L02217	890	NONE				
005062	L02220	891	005054	005055			
005307	L02326	892	005306				
005311	L02330	893	005306				
005332	L02337	895	005331				
005334	L02341	896	005331				
002467	L01205	900	NONE				
002472	L01207	901	002465	002466			
002475	L01210	902	NONE				
002501	L01213	903	002473	002474			
002504	L01214	904	NONE				
002507	L01217	905	002502	002503			
002512	L01220	906	NONE				
002516	L01223	907	002510	002511			
002521	L01224	908	NONE				
002524	L01227	909	002517	002520			
002527	L01231	910	002471	002500	002506	002515	002523
004247	L01751	6175	004244				
004405	L02030	6295	004402				
005152	L02252	6455	NONE				
005154	L02253	6456	005151				
005162	L02257	6475	005155	005157			
005163	L02260	6485	005161				
005066	L02222	6575	NONE				
005071	L02223	6576	005065				
005005	L02176	6585	005003				
005007	L02200	6586	005004				
005012	L02201	6587	005006				
006113	L02504	90001	000056				
001260	L00531	90002	000022	000057	006141		
000566	L00265	90003	000060				
006671	L02670	90004	000061				
006125	L02507	90005	005762	006123			
001260	L00531	90006	000022	000057	006141		
000212	L00074	90007	000214				
000216	L00101	90008	NONE				
006231	L02555	90014	000062				
006544	L02635	90015	000063				
006672	L02671	90017	000054				
006306	L02601	90030	006542				
006543	L02634	90032	006306				
007440	C00542	90034	000024				
006112	L02503	90050	006077	006110			
006116	L02505	90060	NONE				
006124	L02506	90061	006114	006115	006123		
006137	L02516	90065	006133	006135			

```
BLOCK NAMES AND LENGTHS
REALTIM- 000656     INTCOMM- 000043     INTINTR- 000113     CCONA1 - 000076
TABLES - 000156     DRIVER - 000007     FUNCNAM- 000043
```

VARIABLE REFERENCES

LOCATION	GEN TAG	SYM TAG	REFERENCES					
010561	V01007	ABB	004123	004137	004150	004230	004240	005063
010613	V01043	ABDA	005223	005226	005231	005233	005236	
010610	V01040	ABEDA2	005140	005145				
010544	V00772	ABPHI	004001	004310	004315	005002		
010563	V01011	ABYAW	004276	004307	004353	004373	004376	004443
			004451	004600	004603	004606	004611	004614
010073	V00132	ACCEL	000077	005503	006022	006407	006557	
000000C01	A00001	ADC	000026	000042				
010472	V00720	AK	002735					
010463	V00711	ALDREF	002620	003220	003302	003514	003530	
010047	V00076	ALFM	002702	002706	003075			
010046	V00075	ALMN1	002414					
010055	V00104	ALMN2	002405					
010534	V00762	ALPCMD	003606	003614	003640	003643	003644	
000000C06	V00152	ALPDG	000126	001571	004014	004265	004273	004774
			006024	006457	006606			
000027C05	V00304	ALPDGIC	000454	001015				
010257	V00467	ALPHA	001016	001041	001043	001567	001571	001573
			001575					
010113	V00161	ALPHAC	000136	003645	004013	006501	006616	
010023	A00047	ANSW	001657					
010443	V00671	ARC	002235	002267				
010532	V00760	ARG	003570	003576	003577			
010041	V00070	ATK	002266					
000265C01	V00061	AUTOTYP	006106					
010626	V01071	AX	005447	005474	005504	005522	005530	
010627	V01072	AY	005464	005474	005507	005516	005523	005531
			005771	006005				
010630	V01076	AZ	005473	005475	005512	005517	005525	005533
010464	V00712	BA	002662	002665	002675	003125	004070	005452
010510	V00736	BAD	003140	003143	003145	003151	003152	003161
010546	V00774	BATADT	004013	004301	005041	005763		
010205	V00400	BA1	000606	002661	002666	003115	003122	003132
			003153					
010506	V00734	BA2	003123	003132				
010260	V00470	BETA	001020	001045	001047	001603		
010103	V00142	BETDG	000111	001605	002101	004112	004342	005073
			005127	005453	005606	005661	005777	006054
			006433	006571				
000030C05	V00305	BETDGIC	000455	001017				
010623	V01055	BIGP	005370	005411				
010624	V01056	BIGQ	005377	005406	005414			
010625	V01057	BIGR	005406	005415				
010570	V01016	BLL	004340	004345	004350			
010637	V01110	BTV	005563	005606	005662			
010323	V00541	B11	001210	001212	001223	001244		
010325	V00543	B13	001225					
010330	V00547	B17	001241	001247				
010331	V00550	B18	001245					
010324	V00542	B3	001214	001226				
010427	V00654	CAF	002044	002051	005564			
010526	V00754	CAGI	003435	003450	003477	003507		
010413	V00637	CCD	001732	002041	002063	005432	005442	005466
010432	V00657	CCD1	002064	002103	002111	005423		
010412	V00635	CCL	001730	002042	002061	005441	005465	
010431	V00656	CCL1	002062	002102	002110	005432		
000001C07	V00640	CD	001731					
000036C07	V01065	CDBF	005430					
000012C07	V01063	CDE	005424					
000020C07	V01064	CDSB	005426					
010424	V00651	CDV	000316	001604	001617	002022	002034	002056
			002063	005276	005445			
000027C07	V01034	CHB	005072					
000042C07	V01046	CHEL	005302					
000041C07	V01045	CHER	005300					
000000C05	V00252	CHORD	000412	005570				
000026C07	V01033	CHR	005071					
007754	A00036	CIGAR	002303	002316				

000000C07	V00636	CL	001727					
000005C07	V01116	CLB	005575					
000035C07	V01070	CLBF	005437					
000024C07	V01117	CLBSB	005577					
000015C07	V01120	CLDA	005611					
000007C07	V01121	CLDR	005613					
000011C07	V01066	CLE	005433					
000034C07	V01114	CLP	005601					
000032C07	V01115	CLR	005603					
000017C07	V01067	CLSB	005435					
010423	V00650	CLV	002035	002054	002061			
000002C07	V01122	CM	005626					
000037C07	V01126	CMBF	005640					
000013C07	V01124	CME	005634					
000030C07	V01123	CMQ	005631					
000021C07	V01125	CMSB	005636					
010425	V00652	CMV	002033	002060	005627			
000004C07	V01127	CNB	005657					
000023C07	V01130	CNBSB	005660					
000016C07	V01134	CNDA	005667					
000010C07	V01133	CNDR	005664					
000017C05	V00317	CNGRATE	000471	006113				
010040	V00067	CNMFS	002270	002537	003053	003072	003100	003354
000033C07	V01131	CNP	005651					
000031C07	V01132	CNR	005653					
000002C04	A00007	CON1	NONE					
010266	V00476	COSALP	001045	001211	001221	001242	001575	002041
			002055	004011	004344	005014	005443	005465
010465	V00713	COSBA	002677	002724				
010512	V00740	COSBADD	003166	003206				
010270	V00500	COSBET	001051	001207	001222	001237		
010272	V00502	COSGAM	001055	001120	001622	003522		
010300	V00510	COSGCL	001071	001112	001124	001135	001143	001153
			001415	001452	001471	002152	002306	003517
			005352	005542	005552			
010165	V00243	COSGCLT	000266	000277	000306			
010353	V00573	COSLON	001435	002151	002302			
010304	V00514	COSLONI	001107	001112	001131	001141	001424	001447
			001456	005536	005550			
010162	V00240	COSLONT	000260	000271				
010274	V00504	COSPHI	001061	001205	001235	001326	001503	001511
			005410	005420				
010336	V00556	COSPSI	001340	001475				
010322	V00540	COSPSIA	001204	001250				
010276	V00506	COSPSIE	001065	001120	001635	003523		
010062	V00111	COSRAZ	000014	000270				
010327	V00546	COSTHE	001235	001242	001332	001475	001507	001517
			004003	005422				
010422	V00647	CPR	002023					
010404	V00624	CS	001666	001711				
010444	V00672	CTH	002264					
010640	V01111	CTV	005565	005630				
000003C07	V00661	CYB	002100	005450				
000022C07	V01073	CYBSB	005452					
000014C07	V01075	CYDA	005457					
000006C07	V01074	CYDR	005455					
000036C05	V00365	C1	000562					
010523	V00751	C11	003350	003355	003422	003441		
010451	V00677	C16	002432	002440	002447	002453	003426	003531
010450	V00676	C17	002425	002456	002457	002463	003430	003532
000037C05	V00366	C2	000561					
010457	V00705	C21	002577	002607	002745	002762		
010460	V00706	C22	002577	002611	002740	002763	002773	003002
			003012	003021	003031	003045		
010461	V00707	C23	002601	002737	002760	002772	003001	003010
			003020	003030	003043			
010452	V00700	C4	002470	002476	002505	002513	002522	002525
			002613	003200	003247	003443		
010453	V00701	C5	002467	002477	002506	002514	002523	002526
			002614	003201	003250	003445		
000050C01	A00002	DAC	000027	000044	006020			
010053	V00102	DBAR	002164					
010505	V00733	DBB	003115	003121	003122			
010430	V00655	DCAF	000007	002052				
010111	V00155	DELA	000127	001023	001027	003767	005460	005611
			005670	006040	006465	006607		

010116	V00164	DELAC	000141	000727	005210	005232	005236	005241
			005254	006507	006621			
010602	V01030	DELAC1	005023	005205	005207			
010601	V01027	DELAC2	005022	005206	005207			
000040C05	V00312	DELAIC	000464	001022				
010132	V00200	DELANUD	000165	000604	000732	004674	004703	005204
			006337	006645				
010446	V00674	DELAZ	002341	002347	002353	002360	002410	003550
			003557	003563				
000004C06	V00160	DELBF	000136	000437	006473	006616		
000003C06	V00154	DELE	000127	001026	003764	005214	005217	005222
			005225	006043	006463	006607		
000042C05	V00314	DELEIC	000466	001025	004030	004064		
000005C06	V00202	DELEL	000166	001030	001032	003762	005243	005252
			006166	006213	006223	006353	006646	
000006C06	V00203	DELER	000166	001031	001033	003762	005256	005265
			006175	006216	006355	006646		
010550	V00776	DELET	004032	004065	004073			
000111C05	V00316	DELNDIC	000470	000603				
010112	V00156	DELR	000131	001025	001035	005051	005060	005071
			005455	005614	005665	006045	006204	006220
			006226	006467	006611			
010117	V00165	DELRC	000142	000726	005050	005061	005062	005066
			006015	006511	006622			
000041C05	V00313	DELRIC	000465	001024				
000002C06	V00157	DELSB	000132	003731	003741	003747	003756	003761
			004060	006471	006612			
010616	V01050	DEMXL	005332	005342	005346	005347		
010615	V01047	DEMXR	005307	005317	005323	005324		
000005C02	A00005	DERINT	NONE					
010236	V00433	DET	000720					
010167	V00245	DETERM	000333	000401				
010237	V00434	DETI	000721	005741	005750	005757		
010057	V00106	DF	003323	003327	003333	003347		
010604	V01032	DIF	005052	005056	005244	005250	005257	005263
010115	V00163	DLEC	000137	004031	004075	005211	005241	005254
			006505	006617				
010240	V00435	DLECL	000731	005242	005253	005267	006013	
010241	V00436	DLECR	000730	005255	005266	005272	006014	
010525	V00753	DLIM	003415	003420				
010144	V00214	DRAG	000205	002125	002424	002433	002437	002442
			002446	002452	003527	006343	006665	
010456	V00704	DREFP	002604	002617	002655	003170	003173	003211
			003223	003245	003277	003324	003330	003333
			003340	003344	003346	003367	003400	003406
			003415	003421	003440	003470	003527	
010605	V01035	DRMX	005102	005112	005116	005117		
010437	V00665	DT	002205	002206	002240	002244	003022	
010440	V00666	DTH	002210	002211				
010154	V00232	DTR	000246	000254	000262	000775	001002	002661
			002664	003124	003161			
010441	V00667	DVHEAD	002222	002236	002242			
010471	V00717	D23	002733	002741				
000040C02	V00446	EA2	000746	005017				
000037C02	V00444	EA2I	000743	005020				
010606	V01036	EDELA1	005125	005131	005134			
010607	V01037	EDELA2	005136	005152	005154	005156	005163	
010522	V00750	EEF	003322	003345				
010056	V00105	EEF4	003345					
010142	V00212	ELERL	000200	001040	006215	006373	006660	
010143	V00213	ELERR	000201	001041	006220	006371	006661	
010356	V00576	EL1	001476	001527	005362	005514		
010357	V00577	EL2	001500	001530	005363	005521		
010360	V00600	EL3	001501	001532	005364	005527		
010572	V01020	EMX	004714	005617				
010573	V01021	EMY	004730	005642				
010574	V01022	EMZ	004736	005672				
010361	V00601	EM1	001504	001535	005371	005515		
010362	V00602	EM2	001507	001536	005372	005522		
010363	V00603	EM3	001510	001540	005373	005530		
010034	V00063	ENABLE	006102	006106				
010364	V00604	EN1	001514	001543	005400	005516		
010365	V00605	EN2	001516	001544	005401	005524		
010366	V00606	EN3	001520	001546	001707	002701	003517	004004
			005402	005532				

010562	V01010	EP1	004224					
000035C02	V00443	ERALI	000744	004027	004067			
010611	V01041	ERMP	005143	005164	005203			
010565	V01013	ERPHLM	004313	004316	004323	004324	004346	
010547	V00775	ERRALP	004015	004021	004051	004225	006027	
000036C02	V00437	ERRALPD	000733	004052	004066			
010106	V00150	ERROLL	000121	000723	004122	004144	004146	004153
			004175	006447	006601			
010146	V00216	ERRPHI	000206	000725	003777	004324	005005	005007
			006034	006435	006666			
010600	V01026	ERRPHI2	005006	005011	005012			
010107	V00151	.ERRYAW	000122	000722	004277	004305	004351	004403
			004405	004407	004454	004510	004544	006451
			006602					
010557	V01005	ER1	004115	004121				
010560	V01006	ER2	004120					
000005C05	V00261	EX	000424	000702	005675			
000011C05	V00264	EXY	000431	000675	000706	005676		
000010C05	V00266	EXZ	000432	000676	000703	005700	005711	
000006C05	V00262	EY	000426	000672	000700	000711	005704	
000012C05	V00265	EYZ	000430	000673	005705			
010567	V01015	EY2	004334	004347				
010564	V01012	EY4	004304					
000042C02	V00447	EY5	000745	005045				
000041C02	V00445	EY5I	000742	005046				
000007C05	V00263	EZ	000427	000672	005713			
010477	V00725	E1	003027	003037				
010500	V00726	E2	003037					
010417	V00644	F	001766	001777				
010545	V00773	FGV	004005					
010420	V00645	FP	000353	001410	001776	004727	005770	
010407	V00627	FQBAR	001700	004332				
010156	V00234	FTM	000250	000563	001655			
010123	V00171	FUELUSE	000152	000660	004770	006537	006632	
010631	V01077	FX	005506	005622	005647			
010632	V01100	FY	005511	005573	005646			
010633	V01101	FZ	005514	005572	005624			
010104	V00143	GAMDG	000112	001620	006056	006365	006572	
000031C05	V00306	GAMDGIC	000456	001021				
010261	V00471	GAMMA	001022	001051	001053	001616	001620	001622
010316	V00534	GAMMAA	001173	006067				
000064C05	V00334	GCD	000500	001731				
000073C05	V00355	GCDBF	000530	000536	005426			
000066C05	V00331	GCDE	000503	005424				
000077C05	V00343	GCDSB	000524	000544	005425			
000107C05	V00352	GCHB	000515					
000075C05	V00357	GCHBF	000526					
000105C05	V00353	GCHE	000514	005276				
000106C05	V00351	GCHR	000516					
000104C05	V00350	GCHSB	000517					
000063C05	V00333	GCL	000501	001727				
010254	V00460	GCLAT	000776	001065	001067	001407	001411	001413
010066	V00125	GCLATDG	000067	001411	006321	006547		
010164	V00242	GCLATT	000264	000266				
000050C05	V00323	GCLB	000475	005575				
000072C05	V00354	GCLBF	000531	000537	005435			
000103C05	V00347	GCLBSB	000520	000540	005576			
000051C05	V00324	GCLDA	000474	005610				
000032C05	V00307	GCLDGIC	000460	000775				
000052C05	V00325	GCLDR	000473	005613				
000065C05	V00330	GCLE	000504	005433				
000046C05	V00321	GCLP	000477	005601				
000047C05	V00322	GCLR	000476	005603				
000076C05	V00342	GCLSB	000525	000545	005434			
000053C05	V00326	GCM	000506	005626				
000074C05	V00356	GCMBF	000527	000535	005637			
000055C05	V00332	GCME	000502	005633				
000054C05	V00327	GCMQ	000505	005630				
000100C05	V00344	GCMSB	000523	000543	005635			
000056C05	V00335	GCNB	000513	005656				
000102C05	V00346	GCNBSB	000521	000541	005657			
000062C05	V00341	GCNDA	000507	005666				
000061C05	V00340	GCNDR	000510	005663				
000057C05	V00336	GCNP	000512	005651				
000060C05	V00337	GCNR	000511	005653				
000067C05	V00360	GCYB	000534	002100	005450			

000101C05	V00345	GCYBSB	000522	000542	005451		
000071C05	V00362	GCYDA	000532	005456			
000070C05	V00361	GCYDR	000533	005454			
010160	V00236	GE	000253	004002			
010551	V00777	GKAL	004042	004044	004050	004051	
010612	V01042	GKDA	005174	005176	005202	005203	
010603	V01031	GKR	005027	005034	005040	005044	
010571	V01017	GNUD	004671	004673	004674		
010037	V00066	GS	002531	002552	002701	002731	003157	003176
			003213	003317	003414	003432	003473	003501
010042	V00071	GSTART	002552				
000110C05	V00251	GUIDDT	000410	002073			
010347	V00567	GXI	001371	005543			
010350	V00570	GYI	001373	005553			
010351	V00571	GZI	001403	005557			
000001C02	V00404	H	000615	004743	004753		
010065	V00124	HA	000066	001014	001101	001446	001654	003320
			006016	006064	006144	006162	006315	006546
010145	V00215	HADOT	000205	001713	003525	006341	006665
000034C05	V00311	HAIC	000463	001013			
010176	V00367	HAMAX	000564	001657			
010172	V00253	HCHORD	000413	005564			
010531	V00757	HDSER	003520				
010213	V00410	HI	000623	005773	006214	006222	
010614	V01044	HMEC	005277				
010137	V00207	HMEL	000175	005303	005326	005334	006347	006655
010136	V00206	HMER	000175	005301	005303	005311	006345	006655
010140	V00210	HMR	000176	000734	005076	005104	006351	006656
010414	V00641	HO	001742	001757	001772		
010152	V00230	HPI	000243	001276	002344		
010436	V00664	HS	002135	002141	002142	002146	002150	002612
			003173	003212	003246	003277	003355	003433
			003443	003503			
010173	V00255	HSPAN	000416	005562			
000000C04	V00407	H1	000622				
010170	V00246	I	000375	000377	000404		
000146C05	V00225	IDELT	000231	000613			
010202	V00375	IDFG2	000603	003253	003262		
010203	V00376	IDFG3	000602	003451	003460		
010201	V00374	IFT	000601	002663	003154		
010654	V01146	II	NONE				
010147	V00217	INDEX	000211				
000002C02	V00370	INT	000573	001722	002067	003771	004063	004104
			005760	006133	006137		
000267C01	V00060	INTABLS	000020				
000144C05	A00011	INTEG	000037				
000000C03	A00006	INTERN	NONE				
000004C02	V00371	ISCHEME	000574				
000147C05	V00224	ISKEM	000230	000573			
000150C05	V00315	ISLECIC	000467	000575			
010072	V00131	ISLECT	000076	000576	002542	002621	002635	003226
			003303	006331	006556		
010200	V00373	ISTART	000600	002356	002364		
010204	V00377	ISTP	000605	002660	003101	003126	003155	
010177	V00372	ISTRT	000577	002541	002573		
010521	V00747	ITR	003317	003357	003367		
010063	V00112	ITYPE	000017				
007611	A00012	IVARBUF	000020				
010171	V00247	J	000376	000626	002251	002273	002322	002327
			002333				
000151C05	V00221	JSCAN	000225	006073			
007604	V00014	L	005621	005716			
010217	V00414	LATDLA	000646	005144	005150	005160	005162	
010215	V00412	LATPIT	000650	004237	004243	004253	004260	
010214	V00411	LATROL	000651	004136	004142	004155	004165	004177
			004207				
010216	V00413	LATYAW	000647	004356	004401	004411	004426	004456
			004473	004512	004527	004546	004563	
000210C01	A00003	LDISI	000016	000030	000046		
000364C01	A00004	LDISO	000016	000032	000050		
007610	V00020	LMN	002406	002415	002422	002556	003545	003553
007607	V00017	LOD1	002557	003535	003541	003544	003553	003567
			003574				
007616	A00013	LOGIC	000040	000223	001733	006243	
007605	V00015	M	005645	005722			

000001C06	V00012	MACH	000076	001712	001734	002027	003651	004053
			004270	004777	006327	006556		
007603	V00013	MACHL	003652	003656	003676	003702		
010064	V00113	MF	000024	006233	006240	006256	006270	006302
			006310	006334	006360	006404	006430	006454
			006476	006520				
007606	V00016	N	005675	005727				
010150	V00220	NDEX	000215	000217				
000003C02	V00223	NEQ	000227					
000001C04	V00226	NEQ1	000232	000643				
000651C01	V00115	NHOLD	000057					
000650C01	V00114	NOPER	000056					
000144C05	V00122	NOUT	000064	000065	000075	000105	000115	000125
			000135	000145	000151	000154	000164	000174
			000204	006545	006555	006565	006575	006605
			006615	006625	006631	006634	006644	006654
			006664					
000654C01	V00120	NPRINT	000062					
000655C01	V00121	NREAD	000063					
000652C01	V00116	NRESET	000060					
000145C05	V00222	NRUN	000226	006231	006235			
000152C05	V00250	NSTRT	000407	000576				
000653C01	V00117	NTERM	000061					
010120	V00166	NUDCTR	000146	004617	004627	004640	004645	004657
			004675	006363	006626			
000020C05	V00320	OFTEN	000472	006117				
010246	V00451	OJ	000752	000761	001361	001376		
007730	A00030	OLIMITE	000556	006170	006177			
007726	A00027	OLIMITR	000553	006206				
010226	V00423	OMASS	000670	005503				
010227	V00424	OMASSI	000671	001703	005446	005463	005472	
010245	V00450	OMEGA	000747	001103	001123	001152	001425	001470
			005353					
010346	V00566	OMPR	001367	001401				
010247	V00452	OMU	000754	000763	001364			
000005C02	V00462	P	001002	001260	005361	005602	005652	005676
			005711	005724				
010242	V00440	PASTD	000737	006012	006204			
010244	V00442	PASTL	000735	006007	006166			
010243	V00441	PASTR	000736	006010	006175			
010075	V00134	PDG	000101	001261	004010	004117	005013	005133
			006047	006413	006561			
000021C05	V00276	PDGIC	000446	001001				
000006C02	V00144	PDOT	000116	005742	006437	006576		
000013C02	V00465	PHI	001010	001055	001057	001265	001271	001315
			001322	001324				
010114	V00162	PHICM	000137	003647	003776	006503	006617	
010110	V00153	PHIDG	000126	001316	003776	006031	006461	006606
000024C05	V00301	PHIDGIC	000447	001007				
000014C02	V01060	PHIDOT	005413					
010151	V00227	PI	000242	001266	001301	001306		
010105	V00147	PITERR	000117	000724	004226	004245	004247	004251
			006445	006577				
010220	V00415	PITN	000657	004236	004256	004264	004721	004750
010221	V00416	PITP	000656	004235	004257	004263	004720	004752
000017C02	V00553	PSI	001257	001305	001311	001320	001334	001336
010320	V00536	PSIA	001200	001202	006070			
010101	V00140	PSIDG	000107	001321	006423	006567		
000020C02	V01062	PSIDOT	005423					
010256	V00466	PSIE	001013	001061	001063	001627	001631	001633
			002337	002343	002351			
010121	V00167	PSIEDG	000146	001631	006513	006626		
000026C05	V00303	PSIEDIC	000452	001012				
010445	V00673	PSIET	002337					
010157	V00235	PTS	000252	000667	005502			
010345	V00565	PZR	001363	001402				
000007C02	V00463	Q	001004	001262	005370	005632	005677	005705
010074	V00133	QB	000077	001673	003650	003661	004024	004107
			004217	004335	004446	004771	005074	005122
			005165	005170	005177	005275	006062	006411
			006557					
010535	V00763	QBL	003651	003653	003665	003671	003707	003713
			003717	003723				
010076	V00135	QDG	000102	001263	004071	004222	006003	006051
			006415	006562				

000022C05	V00277	QDGIC	000445	001003				
000010C02	V00145	QDOT	000116	005751	005767	006441	006576	
010406	V00626	QS	001675	001703	005565			
010642	V01113	QSCHORD	005571	005642				
010411	V00631	QSO	001704	002107	005444	005462	005470	
010641	V01112	QSSPAN	005567	005616	005672			
000011C02	V00464	R	001006	001263	005377	005604	005654	005701
			005706	005714				
010211	V00405	RATELIM	000617	005246	005250	005261	005263	
010212	V00406	RATLIMR	000621	005053	005056			
010051	V00100	RAZ	000010	000012				
010503	V00731	RCG	003101	003111	003127			
010516	V00744	RDC	003261	003270	003457	003466		
010077	V00136	RDG	000106	001264	004006	006052	006417	006566
000023C05	V00300	RDGIC	000444	001005				
000012C02	V00146	RDOT	000117	005760	006443	006577		
010514	V00742	RDTOLD	003206	003257				
010520	V00746	RDTOL2	003274	003455				
010462	V00710	RDTREF	002616	003203	003205	003253	003257	003267
			003273	003451	003455	003465	003476	003505
			003521					
000035C05	V00453	RE	000755	000764	000767	001077	001352	001443
010004	A00044	REC	000274	000326	000352	000356	000366	002157
007740	A00034	REC1	000336	002277				
010302	V00512	REH	001102	001111	001151	001347	001444	001470
			002151	003515				
010340	V00560	REHI	001350	001366	001404	005352		
010341	V00561	REHIS	001351	001355	001365			
010342	V00562	REOREH	001353					
010343	V00563	REOREHS	001354	001361	001377			
010251	V00455	REORP	000770					
010252	V00456	REORPS	000771					
010502	V00730	REQ	003074	003111	003127			
010253	V00457	RERPM1	000773	001072	001436			
010524	V00752	RER1	003356	003365	003372			
010501	V00727	RFF	003054	003062	003112	003130		
010015	A00045	RG	002160	002164	002256			
010020	A00046	RGP	002165	002173	002215	002224	002241	
010403	V00623	RHO	001630	001664	001671	001705	001745	006021
010447	V00675	RK2ROL	002363	003550	003563	003566	003601	
007732	A00031	RLIMITD	000740	005120	006205			
007734	A00032	RLIMITL	005350	006167				
007736	A00033	RLIMITR	005325	006176				
010566	V01014	RLL	004330	004333				
010255	V00461	RLON	001000	001102	001427	001431	001433	
010067	V00126	RLONDG	000067	001430	006323	006547		
000033C05	V00310	RLONDIC	000461	000777				
010303	V00513	RLONI	001105	001107	001420	001422	001424	
010161	V00237	RLONT	000255	000256	000260			
010405	V00625	RMU	001670	001707				
010071	V00130	RN	000072	001710	002024	006317	006552	
000003C05	V00257	RNREF	000421	001706				
010533	V00761	ROLLC	003602	003621	003625	003646		
010222	V00417	ROLN	000655	004076	004135	004163	004174	004205
			004216	004641	004646	004710	004715	004731
			004742					
010553	V01001	ROLNL	004077	004641	004646			
010223	V00420	ROLP	000654	004077	004134	004164	004173	004206
			004215	004653	004660	004707	004714	004730
			004745					
010554	V01002	ROLPL	004100	004653	004660			
010410	V00630	ROQB	001702	004672				
010646	V01140	ROTL	005722	005734	005743	005752		
010647	V01141	ROTM	005727	005736	005744	005753		
010650	V01142	ROTN	005733	005737	005746	005755		
010643	V01135	ROT1	005703	005723	005731			
010644	V01136	ROT2	005710	005720	005730			
010645	V01137	ROT3	005716	005725				
010250	V00454	RP	000756	000766	000767			
010133	V00201	RPRI	000165	004007	004116	004327	005014	005132
			005774	006367	006645			
010043	V00072	RPT	002533	003136				
010050	V00077	RPT1	003376					
010301	V00511	RSO	001100	001444				

010155	V00233	RTD	000247	001260	001315	001410	001427	001570
			001604	001617	001630	003645	004001	005770
010052	V00101	RTE	002162	002246				
010054	V00103	RTURN	002166	002175	002205	002227	002234	
010141	V00211	RUDR	000176	001037	006223	006375	006656	
010504	V00732	R11	003113	003133	003366	003373		
010507	V00735	R12	003131	003134	003373			
000002C05	V00256	S	000420	001674				
010552	V01000	SBTERM	004056	004062	004064	004071		
010415	V00642	SHR	001754	001760	001774			
010416	V00643	SHRTWHO	001761	001767				
010265	V00475	SINALP	001043	001206	001240	001573	002043	002053
			004010	004114	004346	005015	005134	005442
			005467					
010267	V00477	SINBET	001047	001205	001236			
010271	V00501	SINGAM	001053	001127	001624			
010317	V00535	SINGAMA	001175	001215	001222			
010277	V00507	SINGCL	001067	001072	001116	001131	001140	001147
			001413	001436	001447	001463	002156	002302
			003516	005360	005536	005546	005557	
010166	V00244	SINGCLT	000270	000301	000313			
010352	V00572	SINLON	001433	002154	002304			
010305	V00515	SINLONI	001111	001114	001133	001422	001451	001462
			005540	005546				
010163	V00241	SINLONT	000262	000272	000302	000312		
010273	V00503	SINPHI	001057	001204	001236	001324	001502	001512
			004002	005407				
010335	V00555	SINPSI	001336	001477	001512			
010321	V00537	SINPSIA	001202	001246				
010275	V00505	SINPSIE	001063	001122	001633			
010061	V00110	SINRAZ	000012	000272				
010326	V00545	SINTHE	001233	001243	001330	001500		
010174	V00363	SLPAQ	000547	005201				
000001C05	V00254	SPAN	000415	005566				
010474	V00722	SQ	002175	002764	002767	003014		
010475	V00723	SQQ	002771	002774	003003	003011	003017	003022
			003032	003044				
000000C02	V00123	T	000066	000774	001103	001425	002072	006061
			006113	006116	006313	006546		
007765	A00037	TA	002277	002300	002316			
000000C05	A00010	TABLE	000036					
007623	A00014	TABTAB1	003671	003673	003706	003711	003713	003715
			003717	003721				
007634	A00015	TABTAB2	003664	003667	003723	003725		
007645	A00016	TABTAB3	003675	003700	003702	003704		
010540	V00766	TAB1	003712	004711				
007656	A00017	TAB1A	003706					
010536	V00764	TAB2	003670	003701	004712			
007663	A00020	TAB2A	003664					
007671	A00021	TAB22A	003675					
010035	V00064	TAB3	000005	004720				
010541	V00767	TAB4	003716	004722				
007675	A00022	TAB4A	003712					
010537	V00765	TAB5	003674	003705	004723			
007702	A00023	TAB5A	003670					
007707	A00024	TAB55A	003701					
010542	V00770	TAB6	003722	004725				
007713	A00025	TAB6A	003716					
010036	V00065	TAB7	000006	004733				
010543	V00771	TAB8	003726	004734				
007720	A00026	TAB8A	003722					
010402	V00622	TAMB	001662	001737	002013			
010372	V00612	TANALP	001012	001570	001577	004116	005132	
010334	V00554	TANTHE	001334	005412				
007770	A00040	TAP	002317	002335				
007773	A00041	TARE	002260					
010511	V00737	TDREF	003160	003166				
010466	V00714	TEMP	002702	002706	003072			
010620	V01052	TEMP1	005355	005362	005371	005400		
010621	V01053	TEMP2	005357	005364	005373	005402		
010622	V01054	TEMP3	005361	005366	005375	005404		
010230	V00425	TERMA	000674	000714	005734			
010231	V00426	TERMB	000677	000715	005735			
010232	V00427	TERMC	000702	000716	005736	005751		
010233	V00430	TERMD	000705	005743				

010234	V00431	TERME	000710	005745	005752		
010235	V00432	TERMF	000713	005754			
000015C02	V00544	THE	001231	001233	001275	001301	001317	001326
			001330	001332			
010617	V01051	THECDOT	005354				
010100	V00137	THEDG	000106	001320	006421	006566	
000016C02	V01061	THEDOT	005417				
000016C05	V00275	TIC	000443	000773			
010175	V00364	TISPH	000551	004767			
010153	V00231	TPI	000244	001271	001311	002232	002347	002354
010126	V00174	TPITN	000156	000664	004747	004763	006527	006636
010127	V00175	TPITP	000156	000663	004752	004764	006531	006636
010102	V00141	TRANGE	000107	002271	002533	003135	003375	006036
			006425	006567			
010124	V00172	TROLN	000155	000666	004742	004762	006523	006635
010125	V00173	TROLP	000155	000665	004745	004762	006525	006635
010473	V00721	TT11	002761	003051			
010476	V00724	TT22	003013	003047	003052		
010210	V00403	TW	000612	001756	001764	001776	002007	002011
010421	V00646	TW1	002001	002006	002010		
010575	V01023	TX	004741	005445	005504		
010576	V01024	TY	004740	005462	005507		
010130	V00176	TYAWN	000160	000662	004755	004764	006533	006640
010131	V00177	TYAWP	000161	000661	004757	004765	006535	006641
000266C01	V00062	TYPESW	006102	006105			
010577	V01025	TZ	004737	005471	005512		
010454	V00702	T1	002532	002617	003210	003275	
010455	V00703	T2	002540	002605	002653	003223	003244	003275
010442	V00670	U	002224	002232	002234		
010367	V00607	UB	001534	001550	001565		
007776	A00042	UTARE	002260	002262			
010001	A00043	UXYZE	002256	002262			
010651	V01143	U1	006010	006170			
010652	V01144	U2	006011	006177			
010653	V01145	U3	006013	006206			
010070	V00127	V	000071	001556	001600	001636	001705	001743
			002125	002130	002134	002140	002371	002416
			002464	002472	002501	002507	002516	002574
			002600	002614	002650	002667	002720	002746
			002772	003030	003055	003057	003177	003220
			003241	003251	003263	003266	003272	003423
			003436	003461	003464	003475	003515	003602
			003607	003613	003616	003630	003634	003642
			006020	006153	006325	006551	
010033	V00057	VARCHNG	000017	006077			
010370	V00610	VB	001542	001551	001577		
010134	V00204	VBAR	000170	002030	002037	006377	006650
010426	V00653	VBAROI	002040	002045			
010206	V00401	VBB	000607	002670	002674	002723	002734	002742
			002751	003001	003020	003050	
010470	V00716	VBBS	002724	002726	002743	002752	003051	003063
010527	V00755	VC	003437	003441	003446		
010467	V00715	VCG	002713	002717	003054	003064	003076	
010374	V00614	VI	001637	002610	003174	003247	003506	004004
			005562				
000025C05	V00302	VIC	000451	001010			
010135	V00205	VINERT	000171	001654	002527	006401	006651	
010513	V00741	VOLD	003204	003255			
010517	V00745	VOLD2	001503	002140	002424	002746	003273	003453
			003740	003755	004223	004743	004750	005037
010045	V00074	VQ	002536	002714	002716	003076	
010122	V00170	VR	000147	001011	001117	001126	001564	003726
			003733	003736	003743	003750	003753	004033
			004036	004045	004665	005023	005030	005035
			006065	006515	006627		
010375	V00615	VS	001640	001672	001740	002535	002602	002747
			003216	003300	003321	003431	003444	003470
			003636				
010044	V00073	VSAT	000007				
010060	V00107	VSATS	000010	001641	002527	002700	002707	002727
			003063	003171	003207		
010376	V00616	VSOVS	001642	003156	003472		
010312	V00530	VWXRG	001156	001163	001176	001522	001526	001534
			001542				
010313	V00531	VWYRG	001160	001163	001175	001524	001527	001535
			001543				

010314	V00532	VWZRG	001162	001167	001526	001531	001537	001545
000021C02	V00524	VX	001137	001446	001455	001642	001714	
000022C02	V01105	VXDOT	005545					
010634	V01102	VXDRG	005521	005535	005545	005555		
010306	V00521	VXIG	001121	001130	001137	001146	001155	001454
			001466	001625				
010354	V00574	VXRG	001467	001520	001556	001605	005356	
010377	V00617	VXS	001643	001647				
010373	V00613	VXVY	001612	001614				
010315	V00533	VXVYR	001167	001171				
000023C02	V00525	VY	001145	001450	001644	001715		
000024C02	V01106	VYDOT	005555					
010635	V01103	VYDRG	005527	005537	005547			
010307	V00522	VYIG	001126	001132	001141	001151	001460	001467
010311	V00527	VYRG	001154	001157	001473	001522	001557	001606
			001624	005351				
010400	V00620	VYS	001645	001647				
000025C02	V00526	VZ	001150	001452	001463	001645	001717	
000026C02	V01107	VZDOT	005561					
010636	V01104	VZDRG	005535	005541	005551	005556		
010310	V00523	VZIG	001130	001134	001142	001161	001465	001473
			003011					
010355	V00575	VZRG	001474	001524	001560	001612	001712	
010401	V00621	VZS	001352	001646				
010515	V00743	V1OLD	003256	003263	003266			
010530	V00756	V2OLD	003454	003461	003464			
010371	V00611	WB	001550	001552	001564			
000004C05	V00260	WEIGHT	000423	000666				
000043C05	V00272	WXRG	000442	001155	001521			
000044C05	V00273	WYRG	000441	001157	001523			
000045C05	V00274	WZRG	000440	001161	001525			
000027C02	V00516	X	001113	001342	001370	001416		
000030C02	V00632	XDOT	001262	001715				
010433	V00660	XLFAC	002110	002124	002553			
010434	V00662	XLOD	002112	002116	002117	002405	002414	002421
			003032	003536	003541	003567		
010435	V00663	XLODS	002120	003407				
000013C05	V00267	XOFF	000434	-005623	005645			
007751	A00035	XYZE	002153	002160				
010262	V00472	X1	001033	006172	006213	006224		
010263	V00473	X2	001034	006201	006216	006225		
010264	V00474	X3	001036	006210	006221	006227		
000031C02	V00517	Y	001115	001342	001372	001415		
010224	V00421	YAWN	000653	004101	004372	004424	004442	004471
			004507	004525	004543	004561	004577	004620
			004624	004706	004717	004755		
010555	V01003	YAWNL	004102	004620	004624			
010225	V00422	YAWP	000652	004102	004371	004425	004441	004472
			004506	004526	004542	004562	004576	004631
			004635	004706	004716	004732	004760	
010556	V01004	YAWPL	004103	004631	004635			
010207	V00402	YB	000611	002365	002372	002376	002402	002404
			002407	003560				
000032C02	V00633	YDOT	001716					
000014C05	V00271	YOFF	000435	005571	005646			
000033C02	V00520	Z	001117	001340	001401			
010333	V00552	ZB	001254	001255				
000034C02	V00634	ZDOT	001720	005075				
000015C05	V00270	ZOFF	000433	005572	005622			
010344	V00564	ZOREHS	001356	001374	002101			
010337	V00557	ZS	001341	001343	001354			
010332	V00551	ZT	001251	001254				

START OF CONSTANTS
006676

START OF TEMPORARIES
007453

START OF INDIRECTS
007575

UNUSED COMPILER SPACE
005600

```
              SUBROUTINE LIMIT(VARI,VMIN,VMAX)
000006        IF(VARI-VMIN)2,4,1
000010      1 IF(VARI-VMAX)4,4,3
000013      2 VARI=VMIN
000014        GO TO 4
000015      3 VARI=VMAX
000016      4 RETURN
000017        END


              SUBROUTINE PQ1
000002        COMMON/CCONA1/H,NEQ1,CON1(10,1)
        C *** CON1( 1,N)=NTERM11
        C *** CON1( 2,N)=X
        C *** CON1( 3,N)=U
        C *** CON1( 4,N)=UD
        C *** CON1( 5,N)=UDD
        C *** CON1( 6,N)=W
        C *** CON1( 7,N)=P
        C *** CON1( 8,N)=Q1
        C *** CON1( 9,N)=Q2
        C *** CON1(10,N)=Q3
000002        HH5=H*H*.5
000004        DO 10 N=1,NEQ1
000006        Z=1./CON1(6,N)
000012        CON1( 7,N)=EXP(-CON1(6,N)*H)
000022        CON1( 8,N)=(1.-CON1(7,N))*Z
000025        CON1( 9,N)=( H-CON1(8,N))*Z
000031        CON1(10,N)=(HH5-CON1(9,N))*Z
000035     10 CONTINUE
000040        RETURN
000041        END
```

ADDRESS	LENGTH	BINARY CONTROL CARDS.
0	24	IDENT CONVOl
24		END

BLOCKS	TYPE	ADDRESS	LENGTH
PROGRAM*	LOCAL	0	24
CCONAl	COMMON	0	14

ENTRY POINTS.

CONVOl   -   1

EXTERNAL SYMBOLS.

GOTOER.

```
                          IDENT   CONVOl
                          ENTRY   CONVOl
                          EXT     GOTOER.
                          USE     /CCONAl/
 0                    H    BSS     1
 1                    NEQ  BSS     1
 2                    NTERM BSS    1
 3                    X    BSS     1
 4                    U    BSS     1
 5                    UD   BSS     1
 6                    UDD  BSS     1
 7                    W    BSS     1
10                    P    BSS     1
11                    Q1   BSS     1
12                    Q2   BSS     1
13                    Q3   BSS     1
                          USE     *
 0  03171626173455000000  AVFD VFD     42/0HCONVOl,18/0
 1  00000000000000000000  CONVOl DATA  0
 2  5140000001 C              SA4     NEQ          .X4=NEQ
        5150777767 C          SA5     NTERM-10     .X5=NTERM(0)
 3  66600                     SB6     B0           .I=0
        63240                 SB2     X4           .B2=NEQ
        6170000012            SB7     10           .B7=10
 4  6140000004                SB4     4            .B4=4
        6110000001            SB1     1            .B1=1
 5  54557                LOOP SA5     A5+B7        .A5=ADD(NTERM(I+1))
        43600                 MX6     0            .X6=0
        63550                 SB5     X5+B0        .B5=NTERM(I+1)
 6  0605000010 +              LE      B5,B0,ERROR .IF(NTERM*.LE.0)GO TO ERROR
        0654000010 +          GE      B5,B4,ERROR .IF(NTERM*.GE.4)GO TO ERROR
 7  0250000010 +              JP      B5+ERROR    .GO TO ERROR+NTERM*
10  0100000000 X         ERROR RJ     GOTOER.     .CALL GOTOER.
        0700000000 +      -    LT      B0,B0,AVFD
11  0400000017 +         +    EQ      B0,B0,C1     .GO TO C1
12  0400000015 +         +    EQ      B0,B0,C2     .GO TO C2
13  5015000011 +         +    SA1     A5+9         .A1=ADD(Q3(I+1))
        5025000004            SA2     A5+4         .A2=ADD(UDD(I+1))
14  40612                     FX6     X1*X2        .X6=Q3*UDD
15  5015000010           C2   SA1     A5+8         .X1=Q2
```

```
            5025000003          SA2     A5+3        .X2=UD
16  40712                       FX7     X1*X2       .X7=Q2*UD
        30667                   FX6     X6+X7       .X6=X6+Q2*UD
            24606               NX6     B0,X6       .
17  5015000007        C1        SA1     A5+7        .X1=Q1
        5025000002              SA2     A5+2        .X2=U
20  55311                       SA3     A1-B1       .X3=P
        55421                   SA4     A2-B1       .X4=X
            40712               FX7     X1*X2       .X7=Q1*U
                40534           FX5     X3*X4       .X5=P*X
21  30667                       FX6     X6+X7       .X6=X6+Q1*U
        24606                   NX6     B0,X6       .
            30665               FX6     X6+X5       .X6=X6+P*X
                24606           NX6     B0,X6       .
22  66661                       SB6     B6+B1       .I=I+1
        54640                   SA6     A4          .X=EXPRESION
            0762000005 +        LT      B6,B2,LOOP  .IF(I.LT.NEQ)GO TO LOOP
23  0400000001 +                EQ      CONVOL      .RETURN
24                              END
```

```
            37500   STORAGE USED          59 STATEMENTS      19 SYMBOLS
                    6600 ASSEMBLY       0.196 SECOND         31 REFERENCES
```

```
              SUBROUTINE MODLAG(U,Y,GAIN,RLIMIT,DLIMIT,UPAST,XK)
000012        COMMON/INTCOMM/T,H,INT,NEQ,ISCHEME,DERINT(2,28)
000012        DIMENSION RLIMIT(2),DLIMIT(2)
       C**        COMPUTE ERROR SIGNAL
000012        ERROR=GAIN*(U-Y)
       C***    COMPUTE INPUT DERIVIATIVE
000014        UD=(U-UPAST)/H
       C
       C**        COMPUTE XDOT BY APPLYING RATE LIMIT TO ERROR
       C***       ALSO COMPUTE PARTIAL OF H WITH RESPECT TO ERROR AND
       C***             PARTIAL OF H WITH RESPECT TO TIME
000016        IF(ERROR-RLIMIT(1))1,1,3
000021      1 XDOT=RLIMIT(1)
000023        PHWRE=PHWRT=0.
000025        GO TO 10
000025      3 IF(ERROR-RLIMIT(2))4,2,2
000030      2 XDOT=RLIMIT(2)
000032        PHWRE=PHWRT=0.
000034        GO TO 10
000034      4 XDOT=ERROR
000036        PHWRE=1.  $  PHWRT=GAIN*UD
000041     10 CONTINUE
       C**        COMPUTE PARTIAL G WITH RESPECT TO X
000041        IF(XK-DLIMIT(1))5,5,6
000044      5 PGWRX=0.
000045        GO TO 11
000046      6 IF(XK-DLIMIT(2))7,8,8
000051      8 PGWRX=0.
000052        GO TO 11
000053      7 PGWRX=1.
000055     11 CONTINUE
       C**        COMPUTE A ***
000055        A = -GAIN*PHWRE*PGWRX
       C**        COMPUTE P AND Q
000060        IF(A)22,20,22
000061     20 P = H  $  Q=H*H/2.
000065        GO TO 23
000065     22 P=(EXP(A*H)-1.)/A
000074        Q=(P-H)/A
000076     23 CONTINUE
       C**         INTEGRATE
000076        XKP1=XK + P*XDOT + Q*PHWRT
000104        Y = XKP1
       C***       APPLY DEFLECTION LIMIT TO DETERMINE FINAL OUTPUT
000110        IF(Y-DLIMIT(1))30,30,31
000112     30 Y = DLIMIT(1)
000113        GO TO 32
000114     31 IF(Y-DLIMIT(2))33,34,34
000117     34 Y = DLIMIT(2)
000121        GO TO 32
000121     33 Y = XKP1
000122     32 CONTINUE
000122        RETURN
000123        END


              SUBROUTINE UNVEC(A,B)
000005        DIMENSION A(3),B(3)
000005        AMAG=SQRT(A(1)**2+A(2)**2+A(3)**2)
000014        IF(AMAG.LT..00001) AMAG=0.00001
000020        DO 1 I=1,3
000022        B(I)=A(I)/AMAG
000025      1 CONTINUE
000027        RETURN
000030        END


              FUNCTION DOT(A,B)
000005        DIMENSION A(3),B(3)
000005        DOT=0.
000006        DO 1 I=1,3
000007        DOT=DOT+A(I)*B(I)
000013      1 CONTINUE
000015        RETURN
000016        END
```

126

```
         SUBROUTINE MULT31(A,B,C)
000006   DIMENSION A(3,3),B(3),C(3)
000006   TEM1=B(1)
000007   TEM2=B(2)
000010   TEM3=B(3)
000012   DO 1 I=1,3
000013   C(I)=A(I,1)*TEM1+A(I,2)*TEM2+A(I,3)*TEM3
000030  1 CONTINUE
000032   RETURN
000033   END


         SUBROUTINE ATMOS3(ZZ,ZMAX,ANSW)
000006   DIMENSION HB(10),ZB(14),TMB(24),ALP(24),GB(14),ANSW(8)
000006   DATA (DUM=28.9644),(DUM1=.101325E6),(DUM2=288.16),(DUM3=9.80665)
000006   DATA
        1   ((HB(I),I=1,10)=-5000.0,0.0,11000.0,20000.0,32000.0,47000.0,
          152000.0,61000.0,79000.0,88743.0)
000006   DATA
        1   ((ZB(I),I=1,14)=90000.0,100000.0,110000.0,120000.0,150000.0,16
          10000.0,170000.0,190000.0,230000.0,300000.0,400000.0,500000.0,60000
          20.0,700000.0)
000006   DATA
        1   ((TMB(I),I=1,24)=320.65,288.15,216.65,216.65,228.65,270.65,
          1270.65,252.65,180.65,180.65,180.65,210.65,260.65,360.65,
          2960.65,1110.65,1210.65,1350.65,1550.65,1830.65,2160.65,
          32420.65,2590.65,2700.65)
000006   DATA
        1   ((ALP(I),I=1,24)=12.087778,11.526088,10.027120,8.6079235,
          16.7662077,4.7086738,4.0775458,2.9019653,.37006732E-1,-1.8055744,
          2-1.8055744,-3.5040610,-4.9124564,-5.9828218,-7.5886378,-7.9035491,
          3-8.1833670,-8.6884559,-9.5726883,-10.879634,-12.421644,-13.724116,
          4-14.879663,-15.942630)
000006   DATA
        1   ((GB(I),I=1,14)=9.535,9.505,9.476,9.447,9.360,9.331,9.302,
          19.246,9.134,8.942,8.679,8.428,8.187,7.956)
000006   Z=ZZ
000006   IF(Z.GT.ZMAX) GO TO 499
000012  3 G =((((((-5.5905936E-33)*Z+2.972462E-26)*Z-1.5167771E-19)*Z+7.25394
          155E-13)*Z-3.0854195E-6)*Z+9.80665
000023   IF(Z-90000.0) 4,4,20
000025  4 H=((((((-9.5013649E-35*Z)+6.0621354E-28)*Z-3.8667054E-21)*Z+
          12.4656553E-14)*Z-1.5731262E-7)*Z+1.0)*Z
000036   IF(Z) 6,5,10
000040  5 N=2
000041   GO TO 9
000042  6 IF(Z+5000.0) 499,7,7
000045  7 N=1
000046   GO TO 13
000047  9 AL=0.0
000050   GO TO 14
000051  10 DO 11 N=3,10
000053   IF(H-HB(N)) 12,9,11
000056  11 CONTINUE
```

```
000060          N=9
000061          GO TO 9
000062       12 N=N-1
000064       13 AL=(TMB(N+1)-TMB(N))/(HB(N+1)-HB(N))
000071       14 TM=TMB(N)+AL*(H-HB(N))
000075          IF(AL) 15,16,15
000076       15 ALPP=ALP(N)-3.41631947E-2*ALOG((AL*(H-HB(N))+TMB(N))/TMB(N))/AL
000113          GO TO 30
000115       16 ALPP=ALP(N)-3.41631947E-2*(H-HB(N))/TMB(N)
000123          GO TO 30
000123       20 IF(Z-700000.0) 22,21,21
000126       21 M=14
000127          N=24
000130          GO TO 26
000131       22 DO 23 M=2,14
000133          IF(Z-ZB(M)) 24,25,23
000136       23 CONTINUE
000140       24 M=M-1
000142       25 N=M+10
000144       26 AL =(TMB(N+1)-TMB(N))/(ZB(M+1)-ZB(M))
000151          GG=(G+GB(M))/2.0
000154          TM=TMB(N)+AL *(Z-ZB(M))
000161          IF(AL ) 27,28,27
000162       27 ALPP=ALP(N)-3.48367635E-3*GG*ALOG((Z-ZB(M)+TMB(N)/AL)*AL/TMB(N))/A
             1L
000200          GO TO 30
000203       28 ALPP=ALP(N)-3.48367635E-3*GG*(Z-ZB(M))/TMB(N)
000212       30 P=EXP(ALPP)
000214          RHO=2.84381743E-3*P/TM
000217          IF(Z-90000.0) 31,31,32
000224       31 WM=28.9644
000225          CS      =.589102444E-1*TM**.5
000231          T=TM*WM/DUM
000234          XMU=8.14785279E-2*T**1.5/(T+110.4)
000244          GO TO 500
000244       32 CS      =.791785926
000245          XMU=6.7974E-1
000247          Z=Z*1.E-3
000251          IF(Z-169.5) 33,34,34
000253       33 WM= (((((((.10923927E-13*Z-.11006185E-10)*Z+.49584323E-8)*Z-.13071
             1742E-5)*Z+.22007775E-3)*Z-.24083631E-1)*Z+.16586041E1)*Z-.65156805
             2E2)*Z+1139.0647
000271          GO TO 35
000271       34 WM= ((((((-.21764045E-18*Z+.18921597E-14)*Z-.43903233E-11)*Z+.4509
             16184E-8)*Z-.23285059E-5)*Z+.63311551E-3)*Z-.115575533)*Z+36.084062
000305       35 T=TM*WM/DUM
000307          Z=Z*1.E3
000311          GO TO 500
000312      499 T=0.0
000313          TM=0.0
000313          P=0.0
000314          RHO=0.0
000315          G=0.0
000316          WM=0.0
000317          CS      =.791785926
000321          XMU=1.0
000323      500 ANSW(1)=P/DUM1
000325          ANSW(2)=T/DUM2
000327          ANSW(3)=RHO
000331          ANSW(4)=CS
000332          ANSW(5)=G/DUM3
000334          ANSW(6)=TM/DUM2
000336          ANSW(7)=WM
000340          ANSW(8)=XMU
000341          RETURN
000342          END
```

REFERENCES

1. Malkin, M. S.: Space Shuttle/The New Baseline. Astronaut. & Aeronaut., vol. 12, no. 1, Jan. 1974, pp. 62-68.

2. Houck, Jacob A.; Gibson, Lucille H.; and Steinmetz, George G.: A Real-Time Digital Computer Program for the Simulation of a Single-Rotor Helicopter. NASA TM X-2872, 1974.

3. Powell, Richard W.; Stone, Howard W.; and Rowell, Lawrence F.: Effects of Modifications to the Space Shuttle Entry Guidance and Control Systems. NASA TN D-8273, 1976.

4. Gupta, Someshwar C.; and Hasdorff, Lawrence: Fundamentals of Automatic Control. John Wiley & Sons, Inc., c.1970.

5. Barker, Lawrence E., Jr.; Bowles, Roland L.; and Williams, Louise H.: Development and Application of a Local Linearization Algorithm for the Integration of Quaternion Rate Equations in Real-Time Flight Simulation Problems. NASA TN D-7347, 1973.

6. Giese, Clarence: State Variable Difference Methods for Digital Simulation. Simulation, vol. 8, no. 5, May 1967, pp. 263-271.

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."
—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons. Also includes conference proceedings with either limited or unlimited distribution.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include final reports of major projects, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D.C. 20546