National Defence / Défense nationale

UNCLASSIFIED

DRES

SUFFIELD REPORT

NO. 446

UNLIMITED DISTRIBUTION

A NONLINEAR SIX DEGREE-OF-FREEDOM FLIGHT SIMULATION MODEL

VOLUME 2

SOFTWARE DOCUMENTATION (U)

by

A.B. Markov

PCN 031SE

May 1990

DEFENCE RESEARCH ESTABLISHMENT SUFFIELD : RALSTON : ALBERTA

DEFENSIO PER SCIENTIAM

Canada

UNLIMITED DISTRIBUTION

DEFENCE RESEARCH ESTABLISHMENT SUFFIELD
RALSTON, ALBERTA

SUFFIELD REPORT NO. 446

A NONLINEAR SIX DEGREE-OF-FREEDOM FLIGHT SIMULATION MODEL VOLUME 2 SOFTWARE DOCUMENTATION (U)

by

A.B. Markov

PCN 031SE

ACKNOWLEDGEMENT

The current versions of the simulation software are the result of an evolutionary development that has taken place since 1981. During the course of this development there have been numerous contributions to the software by staff of the Systems Section (SS) of the Defence Research Establishment Suffield (DRES), by staff of the DRES computer group, by SS R&D contractors, and by a number of other outside agencies. The author is grateful for the contributions made by all of these sources.

The contributions of the following are especially noteworthy:

1. The software is loosely based on simulation software developed by the author at the University of Toronto Institute for Aerospace Studies while working as part of the Flight Transportation Group under Prof. L. Reid.

2. Preliminary ideas for the structure and features of the software were generated in discussions between the author and Mr. K. Schilling of West Germany. Mr. Schilling was a visiting NATO fellow to DRES in the period September 1980 to December 1981.

3. A significant portion of the ROBOT-X aerodynamic model is based on wind tunnel data collected at the facilities of the High Speed Aerodynamics Laboratory of the National Aeronautical Establishment of the National Research Council.

4. The plotting software package PLTSIM is based on the PLTNLSRBX plotting package developed by Mr. J. Erling. Mr. Erling was a Research Assistant under the supervision of the author from May 1983 to September 1983.

5. A number of SS R&D contractors have utilized the software extensively and have provided valuable feedback both in identifying and resolving a number of bugs and in improving the utility of the software. In this regard, Mr. K. Lee of Ballistech Systems Inc., Montreal and Mr. B. Gagnon formerly of Boeing of Canada Ltd. made particularly significant contributions in the February 1985 to September 1986 timeframe.

6. The careful reviews of the draft manuscripts by Dr. R. Herring and Dr. S. Barton of DRES were instrumental in finalizing the format and content of the report volumes.

7. The meticulous preparation of the manuscript volumes by Ms. C. Baumann, and patience through several extensive revisions, is particularly appreciated.

ABSTRACT

Six degree-of-freedom, rigid body equations of motion are described suitable for modeling the dynamic characteristics of multistaged rocket-boosted maneuvering aerial targets such as the DRES developed ROBOT-X. These equations of motion form the core of a Fortran simulation package called FLISIM. FLISIM is currently installed on a VAX 11/780 computer and allows for modeling of vehicle thrust and structural asymmetries, time-varying mass and inertia characteristics, autopilot control laws, autopilot update rates, autopilot sensor nonidealities, nonlinear aerodynamic characteristics, variable wind conditions, turbulence, nonstandard atmospheric conditions, stage and individual motor failures, different rocket motor types, and parachute deceleration dynamics. The FLISIM software package has been developed in two versions (FLISIMV1 and FLISIMV2) using two different aerodynamic models. Both are written in VAX 11 Fortran and run under the VMS Operating System. FLISIM is fully supported with a plotting software package (PLTSIM) developed around Tektronix PLOT 10 core software.

Volume 1 describes the development of the equations of motion. Volume 2 is the FLISIM software userbook. Volume 3 contains FLISIM source code listings.

VOLUME 1 - TABLE OF CONTENTS

Page

ACKNOWLEDGEMENTS ii
ABSTRACT iii
VOLUME 1 - TABLE OF CONTENTS iv
VOLUME 2 - TABLE OF CONTENTS v
VOLUME 3 - TABLE OF CONTENTS vii
VOLUME 1 - LIST OF TABLES ix
VOLUME 1 - LIST OF FIGURES x
LIST OF SYMBOLS xi
NOTATION CONVENTIONS xxi

1. INTRODUCTION 1

2. EQUATIONS OF MOTION 4
   - 2.1 Fundamental Assumptions 4
   - 2.2 Reference Frames, Rotation Matrices and Angular Velocities 4
   - 2.3 Newton-Euler Development of the General Equations of Motion 15
   - 2.4 Wind Contributions 23

3. AERODYNAMIC MODELS 25
   - 3.1 The AERO1 Aerodynamic Model 26
   - 3.2 The AERO2 Aerodynamic Model 31
   - 3.3 Transformation of Aerodynamic Moments from F_N to F_B 35
   - 3.4 Transformation of Aerodynamic Forces from Wind Axes to Body Axes 40
   - 3.5 The Aerodynamic Models in Body Axes F_B 42
   - 3.6 Parachute Aerodynamics 50

4. MASS AND MOMENTS OF INERTIA MODEL 55

5. THRUST FORCES AND MOMENTS MODEL 65

6. WIND AND TURBULENCE MODELS 72
   - 6.1 Detailed Description of Model 1 74
   - 6.2 Detailed Description of Model 2 77
   - 6.3 Detailed Description of Model 3 78
   - 6.4 Turbulence Model 79
   - 6.5 Miscellaneous Wind Model Remarks 80

7. ATMOSPHERIC MODEL 82

VOLUME 1 - TABLE OF CONTENTS (Continued)

Page

8. AUXILIARY PARAMETERS AND EQUATIONS 83
   8.1 Vehicle Kinematic Restrictions While on Launcher 83
   8.2 Aspect Angle Equations 86
   8.3 Vehicle Acceleration 87
   8.4 Homing Beacon Geometry 91
   8.5 Servo Dynamics Model 92

9. GENERAL SOFTWARE DESCRIPTION 93

10. SUMMARY AND FUTURE WORK 97

REFERENCES - VOLUME 1 99

TABLES
FIGURES

VOLUME 2 - TABLE OF CONTENTS

Page

ACKNOWLEDGEMENT ii
ABSTRACT iii
VOLUME 1 - TABLE OF CONTENTS iv
VOLUME 2 - TABLE OF CONTENTS v
VOLUME 3 - TABLE OF CONTENTS vii
VOLUME 2 - LIST OF TABLES ix
LIST OF SYMBOLS x
NOTATION CONVENTIONS xx

1. INTRODUCTION 1

2. INSTALLATION DOCUMENTATION 3

3. FLISIMV1 4
   3.1 Structure 4
   3.2 Description of Program Modules 6
   3.3 Input Data Files 7
      3.3.1 Master Data File 7
      3.3.2 Aerodynamic Model Input Data Files 8

VOLUME 2 - TABLE OF CONTENTS
(Continued)

Page

3.4 Output Files 13
3.4.1 Formatted Output Files 13
3.4.2 Screen Output Files 14
3.4.3 Plotting Data Output Files 16
3.5 FLISIMV1 Algorithm 18
3.6 Aerodynamic Model 21
3.7 Thrust Model 22
3.8 Mass and Moment of Inertia Models 23
3.9 Servo Model 23
3.10 Homing Beacon Model 24
3.11 Wind and Turbulence Models 24
3.12 Autopilot Model 26
3.13 Differential Equation Solver 28
3.14 Miscellaneous Remarks 29
3.14.1 Variation of Parameters in Multiple Runs 29
3.14.2 Summary of Fortran Unit Number Assignments 29
3.14.3 Asymmetric Thrust Conditions 30
3.14.4 Reference Frames and Geometry 30
3.14.5 Dimension Limits 31

4. FLISIMV2 32
4.1 Structure 32
4.2 Description of Program Modules 35
4.3 Input Data Files 35
4.3.1 Master Data File 35
4.3.2 Aerodynamic Model Input Data Files 36
4.4 Output Files 36
4.5 FLISIMV2 Algorithm 36
4.6 Aerodynamic Model 37
4.7 Thrust Model 37
4.8 Mass and Moment of Inertia Models 37
4.9 Servo Model 37
4.10 Homing Beacon Model 37
4.11 Wind and Turbulence Models 38
4.12 Autopilot Model 38
4.13 Differential Equation Solver 38
4.14 Miscellaneous Remarks 38

5. PLTSIMV1 39
5.1 Structure 40
5.2 Description of Program Modules 41
5.3 Input Data Files 41

VOLUME 2 - TABLE OF CONTENTS (Continued) Page

5.4 Output Files 42
5.5 PLTSIMV1 Operation 42
5.6 PLTSIMV1 Revisions 51
5.6.1 Changing the Maximum Number of Data Points .. 51
5.6.2 Changing the Number of Variables 52
5.6.3 Organization of Arrays 54

6. VMS COMMAND PROCEDURES 54

7. SUMMARY 57

REFERENCES - VOLUME 2 58

TABLES
FIGURES

VOLUME 3 - TABLE OF CONTENTS Page

ACKNOWLEDGEMENT ii
ABSTRACT iii
VOLUME 1 - TABLE OF CONTENTS iv
VOLUME 2 - TABLE OF CONTENTS v
VOLUME 3 - TABLE OF CONTENTS vii
LIST OF SYMBOLS ix
NOTATION CONVENTIONS xix

PREFACE 1

APPENDIX A - FLISIMV1 SOURCE CODE LISTING
A.1 FLISIMV1 Modules
A.2 FLISIMCO Modules
A.3 FLISIMUS Modules
A.4 FLISIMAP Modules
A.5 AEROATMOS Modules
A.6 MATHPACKP Modules

APPENDIX B - FLISIMV2 SOURCE CODE LISTING
B.1 FLISIMV2 Modules
B.2 WTDATA Module
B.3 MATHPACKP Modules

VOLUME 3 - TABLE OF CONTENTS (Continued)

APPENDIX C - PLTSIMV1 SOURCE CODE LISTING
- C.1 PLTSIMV1 Modules
- C.2 PLOTPACK Modules

APPENDIX D - VMS COMMAND PROCEDURES
- D.1 FLISIMV1 Command Procedure
- D.2 FLISIMV2 Command Procedure

APPENDIX E - SIMULATION TEST CASE
- E.1 Description
- E.2 Master Data File
- E.3 Aerodynamic Data Files - CAN6AER2 Model
  - E.3.1 Longitudinal
  - E.3.2 Lateral
- E.4 Formatted Output File

VOLUME 2 - LIST OF TABLES

TABLE 1 FLISIMV1 PROGRAM MODULE DESCRIPTIONS

TABLE 2 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE FORMAT

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE PARAMETER DESCRIPTIONS

TABLE 4 FLISIMV1 AND FLISIMV2 FORMATTED OUTPUT FILE TABLE HEADERS

TABLE 5 FLISIMV1 AND FLISIMV2 FORMATTED OUTPUT FILES TABLE VARIABLE INDICES

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION

TABLE 7 FLISIMV1 AND FLISIMV2 ARRAY PAR ELEMENT DEFINITIONS

TABLE 8 ESTIMATED STATE VECTOR XEST: ROBOT-X FLISIMAP MODULE

TABLE 9 AUXILIARY STATE VECTOR XAUXES: ROBOT-X FLISIMAP MODULE

TABLE 10 SENSOR OUTPUT VECTOR XSEN: ROBOT-X FLISIMAP MODULE

TABLE 11 FLISIMV2 PROGRAM MODULE DESCRIPTIONS

TABLE 12 FLISIMV1 PROGRAM MODULE DESCRIPTIONS

TABLE 13 PLTSIMV1 SYMBOL TYPES

TABLE 14 PLTSIMV1 LINE TYPES

TABLE 15 PLTSIMV1 ARRAY VARTYP UNIT CODES

TABLE 16 PLTSIMV1 ARRAY ELEMENT STORAGE

LIST OF SYMBOLS

The following is a list of important symbols. Some symbols, which are defined in the text and are used only once, or are secondary quantities related to a primary quantity that is apparent from the text or from the notation conventions to follow are not included. Numbers in parentheses refer to equations.

$\vec{A}$	Aerodynamic force vector applied to the vehicle not including thrust forces and parachute forces.
$\vec{a}$, $\vec{a}_B$	Vehicle centre-of-mass acceleration vector relative to inertial space.
a	Speed of sound (m/s).
b	Reference length (span of wing for winged flight vehicles) (m).
C_D	(Total drag) / ($q_D S$), vehicle drag coefficient.
C_{D_1}, $C^N_{D_1}$	Partial drag contributions, see Volume 1, Section 3.
C_L	(Total lift) / ($q_D S$), vehicle lift coefficient.

LIST OF SYMBOLS (cont'd)

C_{L_1}, $C^N_{L_1}$	Partial lift contributions, see Volume 1, Section 3.
C_{L_α}	$\left.\frac{\partial C_L}{\partial \alpha}\right\rvert_e$ (/rad).
C^B_ℓ, C^N_ℓ	$L_{A_B}/(q_D S b)$, $L_{A_N}/(q_D S b)$.
C^B_m, C^N_m	$M_{A_B}/(q_D S \bar{c})$, $M_{A_N}/(q_D S \bar{c})$.
C_{m_1}, C^N_m	Partial pitching moment contributions, see Volume 1, Section 3.
C^B_n, C^N_n	$N_{A_B}/(q_D S b)$, $N_{A_N}/(q_D S b)$.
C_T	$T/(q_D S)$, thrust coefficient.
C_W	$mg/(q_D S)$, weight coefficient.
C^B_x	$X_B/(q_D S)$.
C^B_y	$Y_B/(q_D S)$.
C^B_z	$Z_B/(q_D S)$.
$\bar{c}$	Mean geometric chord (m).

LIST OF SYMBOLS (cont'd)

$\bar{c}_A$	Mean aerodynamic chord (m).
D	Total drag.
$\underset{\rightarrow}{F}$	External force vector acting on the vehicle centre-of-mass.
F_B	Body-fixed reference frame with origin at the vehicle centre-of-mass, see Figure 3 of Volume 1.
F_E	North pointing Earth-fixed reference frame, see Section 2.2 and Figure 2 of Volume 1.
F_I	North pointing inertial reference frame, see Section 2.2 and Figure 2 of Volume 1.
F_L	Launch site reference frame, see Section 2.2 and Figure 2 of Volume 1.
F_{LA}	Launcher reference frame, see Section 2.2 and Figure 2 of Volume 1.
F_N	Body-fixed reference frame to which nominal aerodynamic characteristics are referenced, see Figure 3 of Volume 1.

LIST OF SYMBOLS (cont'd)

F_R	Structural body-fixed reference frame with origin at nose datum time, see Figure 3 of Volume 1.
F_T	Stand-off Earth-fixed reference frame, see Figure 2 of Volume 1.
F_W, F_{WN}	Wind axes reference frames, see Figure 4 and Section 2.2 of Volume 1.
g	Acceleration due to gravity (m/s^2).
g_o	Nominal sea level acceleration due to gravity (m/s^2).
$\underset{\to}{h}$	Vehicle angular momentum vector about centre-of-mass.
h_{ASL}	Altitude of vehicle centre-of-mass above sea level.
I_{sp_i}	Specific impulse of i-th rocket motor.
I^B_{xx}, I^B_{yy}, ...	Vehicle moments of inertia about its centre-of-mass written as components in F_B.
$(L_{A_B}, M_{A_B}, N_{A_B})$	Aerodynamic moment components in F_B not including thrust and parachute moments, about centre-of-mass.

LIST OF SYMBOLS (cont'd)

$(L_{T_B}, M_{T_B}, N_{T_B})$ — Thrust moment components in F_B about centre-of-mass.

$(L_{T_B}, M_{T_B}, N_{T_B})_{cg}$ — See Section 5 of Volume 1.

$(L_{T_B}, M_{T_B}, N_{T_B})_{nz}$ — See Section 5 of Volume 1.

$\vec{M}_A$ — Aerodynamic moment vector acting about the vehicle centre-of-mass not including thrust moments or parachute moments.

$\vec{M}_P$ — Aerodynamic parachute moment vector acting about the vehicle centre-of-mass.

$\vec{M}_T$ — Thrust moment vector acting about the vehicle centre-of-mass.

m — Vehicle total mass.

m_i — Mass of the i-th airframe component.

$(m_{RE})_i$ — Mass of i-th rocket motor less propellant.

$(m_{PR})_i$ — Mass of i-th rocket motor's propellant.

N_M — Total number of rocket motors.

p_A — Atmospheric pressure.

LIST OF SYMBOLS (cont'd)

(P_B, Q_B, R_B)	Angular velocity components of F_B with respect to F_I written as components in F_B.
q_D	Dynamic pressure, $\frac{1}{2}\rho V^2$ (Pa).
$\underset{\rightarrow}{R}$	Position vector of vehicle centre-of-mass relative to F_I.
$\underset{\rightarrow}{R}_T$	Position vector of vehicle centre-of-mass relative to F_T.
$\underset{\rightarrow}{R}_{IT}$	Position vector of R_T relative to F_I.
R	See equation (7,5) of Volume 1; also the magnitude of the range vector depending on the context.
r_E	Radius of the Earth to the nominal sea level datum plane (m).
S	Reference area (projected wing planform area for winged vehicles) (m^2).
s	Distance the vehicle has moved along the launcher from its rest position (see Section 8.1 of Volume 1) (m).

LIST OF SYMBOLS (cont'd)

s_G	Launch rail guide length (see Figure 14 of Volume 1) (m).
s_n	The length of vehicle in front of the launcher while the vehicle is at rest on the launcher (see Figure 14 of Volume 1) (m).
$\vec{T}$	Thrust vector.
T_A	Atmospheric temperature (degrees Kelvin).
T_i	Thrust of the i-th rocket motor (N).
(U_B, V_B, W_B)	Components of $\vec{V}$ in F_B.
$(U_{B_E}, V_{B_E}, W_{B_E})$	Components of $\vec{V}_E$ in F_B.
$(U_{B_g}, V_{B_g}, W_{B_g})$	Components of $\vec{W}$ in F_B.
$\vec{V}$	Airspeed vector.
$\vec{V}_E$	Velocity vector of the vehicle centre-of-mass with respect to F_I and F_E.
V	Magnitude of $\vec{V}$ (m/s).
V_{xz}	$(U_B^2 + W_B^2)^{1/2}$ (m/s).

LIST OF SYMBOLS (cont'd)

$\underset{\rightarrow}{W}$	Wind velocity vector with respect to F_I.
(W_1, W_2, W_3)	Components of $\underset{\rightarrow}{W}$ in F_I (m/s).
$(X_{A_B}, Y_{A_B}, Z_{A_B})$	Aerodynamic force components in F_B, acting at the vehicle centre-of-mass not including thrust and parachute contributions (N).
$(X_{P_B}, Y_{P_B}, Z_{P_B})$	Aerodynamic parachute force components in F_B acting at vehicle centre-of-mass (N).
$(X_{T_B}, Y_{T_B}, Z_{T_B})$	Thrust vector components in F_B (N).
(x_{cm}, y_{cm}, z_{cm})	Coordinates of the vehicle centre-of-mass in F_R (m).
(x_T, y_T, z_T)	Components of $\underset{\rightarrow}{R}_T$ in F_T (see Section 8.2 of Volume 1) (m).
(x_E, y_E, z_E)	Components of $\underset{\rightarrow}{R}$ in F_E (m).
(x_I, y_I, z_I)	Components of $\underset{\rightarrow}{R}$ in F_I (m).
$[(x_{RE})_i, (y_{RE})_i, (z_{RE})_i]$	Coordinates of the centre-of-mass of the empty motor case of the i-th rocket motor in F_R (m).

LIST OF SYMBOLS (cont'd)

(x_i, y_i, z_i)	Coordinates of the centre-of-mass of the i-th airframe component in F_R (m).
$[(x_{PR})_i, (y_{PR})_i, (z_{PR})_i]$	Coordinates of the centre-of-mass of the propellant of the i-th rocket motor in F_R (m).
α_f	Fuselage reference line angle of attack of vehicle.
β	Angle of sideslip of vehicle.
θ_B	Pitch Euler angle of F_B.
ξ_A	Aspect elevation angle of vehicle relative to F_T (see Figure 2 of Volume 1).
ξ_E	Aspect azimuth angle of vehicle relative to F_T (see Figure 2 of Volume 1).
ρ	Air density (kg/m³).
ϕ_B	Euler bank angle for F_B.
ϕ	Cylindrical coordinate (see Figure 5 of Volume 1).
ψ_B	Euler azimuth angle for F_B.

LIST OF SYMBOLS (cont'd)

ψ_L	Euler azimuth angle for F_L.
ψ_{LW}	Euler azimuth angle for F_{LW}.
ψ_T	Euler azimuth angle for F_T.
$\vec{\omega}_B$	Angular velocity vector of F_B with respect to F_I.

NOTATION CONVENTIONS

$C^{C}_{A_s}$ Generalized aerodynamic derivative notation, $\left.\frac{\partial C^{C}_{A}}{\partial s}\right|_{e}$ where C_A is a nondimensional aerodynamic coefficient, specified by the subscript A, in reference frame F_C, s is a dynamic variable, and the 'e' notation indicates the aerodynamic derivative is evaluated at equilibrium.

F_A Reference frame A.

$\underline{L}_{AB}$ A rotation matrix rotating components of a vector expressed in F_B to the components of the same vector expressed in F_A.

$\vec{X}$ Vector quantity X.

$\vec{X} \times \vec{Y}$ Vector cross-product of X and Y.

$\underline{X}$ A matrix X.

$\underline{X}^A$ $\vec{X}$ expressed as components in F_A.

$\underline{X}^T$ The transpose of $\underline{X}$ or the components of a vector $\vec{X}$ expressed in the reference frame F_T(context will determine which interpretation is intended).

X* Indicates a parameter X estimated in the flight control system algorithms (used primarily in Volume 2).

$\underline{x}$ A column matrix x.

NOTATION CONVENTIONS (cont'd)

x_e A quantity x whose value is computed for an aerodynamic quasi-steady equilibrium condition.

x_s A quantity x sensed through the flight control system sensors (used primarily in Volume 2).

1. INTRODUCTION

During ROBOT-X development at the Defence Research Establishment Suffield (DRES), extensive use of design and flight dynamics software was made by DRES staff and DRES R&D contractors. This software has been used in predicting flight vehicle performance, stability and control characteristics and in developing suitable control algorithms for the autopilot system.

A critical ingredient in this software has been a six degree-of-freedom, nonlinear simulation package. This package was developed substantially at DRES in the timeframe August 1981 to August 1985. It is based on extensively modified software developed by the author at the University of Toronto Institute for Aerospace Studies as part of the work in Reference 1. This software has many features that made it particularly well suited for simulating ROBOT-X flight characteristics although it has been structured in a way that allows its application to the simulation of a much broader class of flight vehicles.

The main features of the simulation package are as follows:

a) Time varying mass and moment of inertia characteristics.
b) Time varying thrust characteristics.
c) Two standard aerodynamic models that allow input of the characteristics of a wide variety of flight configurations based on predicted, wind tunnel and flight test aerodynamic data.
d) Modeling of ripple rocket firing characteristics.
e) Modeling of control dynamics of flight vehicles that have any combination of throttle, elevator, all-flying stabilizer, ailerons and rudder.

f) Modeling of digital autopilot control algorithms including discrete sensor and control command updates.

g) Modeling of sensor nonidealities including sensor lag and bias.

h) Modeling of variable wind characteristics including discrete gusts, wind shear and turbulence.

i) An atmospheric model based on the standard ICAO atmosphere.

j) An input data file conveniently structured for parametric and sensitivity variation of critical thrust, mass and aerodynamic characteristics.

k) An interactively structured command procedure that allows convenient variation of autopilot control algorithms and control gains, and wind and turbulence models.

l) An interactive plotting package developed around Tektronix PLOT 10 (References 2 and 3).

Although the package has undergone an evolutionary development since 1981, with numerous versions and variations, its current form consists of two comprehensive versions that have combined the useful features of the others. These versions, named FLISIMV1 (FLIght SIMulation, Version 1) and FLISIMV2 (FLIght SIMulation, Version 2) differ significantly in the form of the aerodynamic model but are otherwise identical and have many subroutines in common. Both versions share a common plotting package named PLTSIMV1 (PLoT SIMulation, Version 1).

This report (Volume 2 of a three volume series) covers the user documentation for the simulation and plotting software. Volume 1 develops the simulation model rigorously and presents the modeling assumptions and equations in detail. Volume 3 gives source listings and test cases.

Section 2 summarizes software installation features.

Section 3 documents FLISIMV1.

Section 4 documents FLISIMV2.

Section 5 documents PLTSIMV1.

Section 6 documents the command procedures linking the different elements of software, initializing system parameters and providing a user-friendly, interactive environment in which to operate the software.

Listings of the various software elements and a test case are given in the appendices. It is noted that the appendices are part of Volume 3.

The documentation is of sufficient depth to allow convenient and rapid user familiarization with the software and debugging of problems encountered.

2. INSTALLATION DOCUMENTATION

The flight simulation software was developed, and is currently installed, on a VAX 11/780 computer at DRES. It runs using the DEC VMS (version 5.1) operating system. It is resident in the Systems Integration Group (SIG) flight dynamics library on disk pack DISKSIG.

FLISIMV1, FLISIMV2 and PLTSIMV1 are written in DEC Fortran Version 5.2-33 (Reference 4) and earlier compatible versions. FLISIMV1 and FLISIMV2 make use of a number of International Mathematical and Statistical Library (IMSL, Reference 5) subroutines. PLTSIMV1 makes use of Fortran callable Tektronix PLOT 10 subroutines (References 2 and 3).

The command procedure has been written in DEC Command Language (DCL, Reference 6).

The plotting software has been run successfully on VT100 Retrographics and Tektronix 4105, 4107, 4112 and 4114 terminals.

3. FLISIMV1

A Fortran source listing not including IMSL subroutines is given in Appendix A (Volume 3). This listing is extensively commented and provides additional user information to that given in this section.

A FLISIMV1 test case in given in Appendix E.

3.1 Structure

FLISIMV1 has been structured to be as modular as possible in order to enhance program readability and to facilitate modification to different flight simulation applications.

FLISIMV1 comprises seven program segments consisting of several subroutines each, as follows:

a) FLISIMV1 - Program modules unique to FLISIMV1

b) FLISIMCO - Program modules common to FLISIMV1 and FLISIMV2.

c) FLISIMUS - Program modules that occasionally require modification by the user (e.g., wind modeling subroutines).

d) FLISIMAP - Program modules that model the autopilot and sensor characteristics.

e) AEROATMOS - A number of subroutines from the DRES Systems Integration Group (SIG) library AEROATMOS are used. This library contains subroutines common to a number of DRES/SIG flight dynamics packages and has been built-up since 1981.

f) MATHPACKP - A number of subroutines from the DRES/SIG mathematical and utility library MATHPACKP are used. This library contains subroutines common to a number of DRES/SIG flight dynamics packages and has been built-up since 1981.

g) IMSL - A number of subroutines from the IMSL library (Reference 5) are used.

FLISIMV1 contains as one of its modules the main program that controls program execution.

FLISIMAP is entirely user supplied and models the characteristics of the autopilot and associated sensors. The only restriction on this module is that it be Fortran callable, that it have a calling statement compatible with that in FLISIMV1, and that it has a compatible common structure. The same calling statement will also work for FLISIMV2.

Program modules are described in detail in Section 3.2. FLISIMV1 requires three input files, as follows:

a) A master data file.
b) A longitudinal aerodynamic characteristics data file.
c) A lateral aerodynamic characteristics data file.

The input data files are documented in detail in Section 3.3.

FLISIMV1 generates the following output files:

a) A formatted output file suitable for printer output.
b) A screen output file allowing interactive assessment of simulation progress.
c) NCASE direct access files containing the data required by PLTSIMV1 in plotting the results of FLISIMV1. NCASE corresponds to the number of cases computed in a given FLISIMV1 execution.

The output data files will be documented in detail in Section 3.4.

3.2 Description of Program Modules

Including the main program and all library subroutines not intrinsic to Fortran 77, the FLISIMV1 software package consists of 48 modules grouped as follows:

a) FLISIMV1 - Three including the main program.
b) FLISIMCO - Eleven subroutines.

c) FLISIMUS - Four subroutines.

d) FLISIMAP - Ten subroutines and one block data module.

e) AEROATMOS - Six AEROATMOS library subroutines.

f) MATHPACKP - Three MATHPACKP library subroutines.

g) IMSL - Ten IMSL subroutines.

FLISIMUS may be modified by the user to change the wind characteristics, the servo dynamic model, the homing beacon model and master data characteristics from case to case in a given run.

FLISIMAP is provided by the user and may be changed as required to model the autopilot algorithms and autopilot sensor characteristics.

The modules associated with program segments FLISIMUS and FLISIMAP are described in this documentation only in the context of their use for ROBOT-X.

The program modules are described in detail in Table 1.

Listings and additional documentation on these modules are given in Appendix A.

3.3 Input Data Files

3.3.1 Master Data File

The master data file is a formatted data file that must be provided by the user. It does not include parameters unique to program segments FLISIMUS and FLISIMAP as these are user defined in form and number and are thus not conveniently amenable to standard input formats.

The data read from the master data file is associated with Fortran unit 5.

The master data parameter names and associated input formats are given in Table 2. A more detailed description of each parameter is given in Table 3.

A sample master data input file, set-up for ROBOT-X simulation, is given in Appendix E of Volume 3.

3.3.2 Aerodynamic Model Input Data Files

Two aerodynamic input data files are required, the first corresponding to the longitudinal aerodynamic characteristics and the second corresponding to the lateral aerodynamic characteristics.

FLISIMV1 assumes that these files are direct access files. They contain a number of features (e.g., a flap setting parameter) that make them useful in more general applications to that of ROBOT-X and FLISIMV1. The characteristics of these files are described in the following:

a) The longitudinal aerodynamic data file is associated with Fortran unit 15. The lateral aerodynamic data file is associated with Fortran unit 16.
b) Both the longitudinal and lateral aerodynamic files contain data that allows dimensionalizing of the aerodynamic coefficients as well as correction for centre-of-mass condi-

tions that are different from that assumed in generating the aerodynamic model. This data is contained in records 1, 2 and 3 of both files. FLISIMV1 assumes that these records are identical in both files, i.e., reference areas and lengths and centre-of-mass position data is assumed to be identical for both the longitudinal and lateral aerodynamic model files and thus only one set of parameters is used.

c) Each direct access record is 80 bytes long.

d) The record structure is as follows (only elements used in the simulation software are shown):

Record 1 - NFS
Record 2 - FLAP
Record 3 - (APAR(I), I = 1, 11), IFHS, IROLL, APAR(12)

Here

NFS	= number of flap settings (for ROBOT-X, NFS = 1).		
FLAP	= flap setting in degrees, REAL*4 (for ROBOT-X FLAP = 0 deg)		
IFHS	= types of pitch trim control (IFHS = 0 for elevator, IFHS = 1 for canard)		
IROLL	= types of roll control (IROLL = 0 for aileron, IROLL = 1 for differential stabilizer or canard)		
APAR(1)	= S	(m^2)	(SS)
APAR(2)	= b	(m)	(B)
APAR(3)	= $\bar{c}$	(m)	(C)

APAR(5)	$= h_{cg}/\bar{c}$		(AXCMND)
APAR(6)	$= (h_{y_{cg}}/\bar{c})_{aer}$		(AYCMND)
APAR(7)	$= (h_{z_{cg}}/\bar{c})_{aer}$		(AZCMND)
APAR(8)	$= (x_R)_{aer}$	(m)	(XCMN)
APAR(9)	$= (y_R)_{aer}$	(m)	(YCMN)
APAR(10)	$= (z_R)_{aer}$	(m)	(ZCMN)
APAR(11)	$= (x_R)_{\bar{c}}$	(m)	(XWGC)
APAR(12)	$= S_{base}$	(m^2)	(SSBREF)

APAR is a REAL*4 array. It is noted that APAR(4) is not used in the simulation software.

(AXCMND, AYCMND, AZCMND) are the nondimensional coordinates (normalized with the mean geometric chord $\bar{c}$) of the origin of the nominal aerodynamic reference frame F_{AERO} with respect to the leading edge of the mean geometric chord.

(XCMN, YCMN, ZCMN) are the dimensional coordinates of the same point in the structural reference frame F_R.

XWGC is the x-coordinate (in F_R) of the leading edge of the wing mean geometric chord. As a check, the relationship

XCMN = C * AXCMND + XWGC

must hold.

e) All records after and including record INITR (see modules INLOAD and/or INLAAD in Appendix A) contain the aerodynamic coefficients associated with various combinations of flap setting, Mach number, thrust coefficient and angle of attack [for ROBOT-X only one flap setting (FLAP = 0, IDFLAP = 1) and thrust coefficient (C_T = 0) are used].

f) For the longitudinal aerodynamic data file, 15 coefficients are stored per file record as

(LONAER(I), I = 1, 15)

where LONAER is a REAL*4 array. LONAER contains the following data:

LONAER(1)	C_{D_1}		
LONAER(2)	C_{L_1}		
LONAER(3)	C_m^N		
LONAER(4)	C_{D_q}	(/rad)	
LONAER(5)	C_{L_q}	(/rad)	
LONAER(6)	$C^N_{m_q}$		
LONAER(7)	$C_{D_{\dot{\alpha}}}$	(/rad)	
LONAER(8)	$C_{L_{\dot{\alpha}}}$	(/rad)	
LONAER(9)	$C^N_{m_{\dot{\alpha}}}$	(/rad)	
LONAER(10)	$C_{D_{\delta_E}}$	(/rad)	: δ_E positive trailing edge of control surface down.
LONAER(11)	$C_{L_{\delta_E}}$	(/rad)	
LONAER(12)	$C^N_{m_{\delta_E}}$	(/rad)	
LONAER(13)	$C_{D_{i_H}}$	(/rad)	: i_H positive trailing edge of control surface down.
LONAER(14)	$C_{L_{i_H}}$	(/rad)	

LONAER(15) $C^N_{m_{i_H}}$ (/rad)

g) For the lateral aerodynamic data file, 15 coefficients are stored per file record as

(LATAER(I), I = 1, 15)

where LATAER is a REAL*4 array. LATAER contains the following data:

LATAER(1)	$C^N_{y_\beta}$	(/rad)	
LATAER(2)	$C^N_{\ell_\beta}$	(/rad)	
LATAER(3)	$C^N_{n_\beta}$	(/rad)	
LATAER(4)	$C^N_{y_p}$	(/rad)	
LATAER(5)	$C^N_{\ell_p}$	(/rad)	
LATAER(6)	$C^N_{n_p}$	(/rad)	
LATAER(7)	$C^N_{y_r}$	(/rad)	
LATAER(8)	$C^N_{\ell_r}$	(/rad)	
LATAER(9)	$C^N_{n_r}$	(/rad)	
LATAER(10)	$C^N_{y_{\delta_A}}$	(/rad)	: δ_A positive right aileron up, left down ($\delta_A = (\delta_{A_L} - \delta_{A_R})/2$).
LATAER(11)	$C^N_{\ell_{\delta_A}}$	(/rad)	

LATAER(12) $C^N_{n_{\delta_A}}$ (/rad)

LATAER(13) $C^N_{y_{\delta_R}}$ (/rad) : δ_R positive for trailing edge of rudder to the left.

LATAER(14) $C^N_{\ell_{\delta_R}}$ (/rad)

LATAER(15) $C^N_{n_{\delta_R}}$ (/rad)

3.4 Output Files

3.4.1 Formatted Output Files

FLISIMV1 and FLISIMV2 generate one formatted output file in a form suitable for printer hardcopy. This file is associated with Fortran unit 6.

The file contains the following:

a) The master data corresponding to each case including changes introduced by subroutine INPCHA from case to case.
b) Output resulting from user defined modules.
c) A response summary table that is always outputted. This table includes

t(sec), V(m/s), Θ(deg), β(deg), x_I(m), y_I(m), h_{AGL}(m)

d) Up to 20 user selectable output tables that may be suppressed using the array IOUTS. If IOUTS(I) = 1, the i-th table is outputted. If IOUTS(I) = 0, the output of the I-th table is suppressed (also see Section 3.3.1). The content of each of the twenty output tables is described in Tables 4 and 5. Table parameters are outputted each RDELT seconds for rocket motor thrust = 0 and each RDTINT * ISPSO seconds for thrust ≠ 0 (RDTINT = RDELT/ISPSO).

e) At the end of each case, a mission summary including

ICASE - user defined case designator
KSTAG - number of rocket motor stages fired
AVVEL - average airspeed (m/s)
AVMACH - average Mach number
AVALT - average altitude (m)
RMAVEL - maximum airspeed (m/s)
RMAMAC - maximum Mach number
RMAALT - maximum altitude (m)
SX(1) - flight time
SX(47) - range at end of simulation (m)
(TISTAG(I), I = 0 - stage firing times (sec)

f) At the end of all cases, a case summary that includes ICASE, KSTAG, AVVEL, AVMACH, AVALT, RMAVEL, RMAMAC, RMAALT, SX(1), SX(47) in table form.

Items (a) to (e) inclusive are outputted for each case. Item (f) is outputted only once at the successful completion of all cases.

3.4.2 Screen Output File

The program has provision for outputting to Fortran unit 7 a limited number of key parameters as program execution takes place.

This is intended to provide the user an interactive monitor on the progress of the simulation for runs submitted in the interactive mode. In the event that the run is submitted as a batch job, this output may be assigned to a dummy file or to the batch job log file. This is further discussed in Section 6 where the FLISIMV1 command procedure is documented.

Parameters are outputted to unit 7 each RDELT seconds for rocket motor thrust = 0 and each RDTINT * ISPSO1 seconds for rocket motor thrust ≠ 0 (RDTINT = RDELT/ISPSO). These parameters are as follows:

T	t	(sec)	time
VV	V	(m/s)	airspeed
MACH NO	M_a		Mach number
THETBB	Θ	(deg)	vehicle pitch Euler angle
PHI	Φ	(deg)	vehicle roll Euler angle
BETA	β	(deg)	angle of sideslip
ALPHAF	α_f	(deg)	angle of attack
XII	x_I	(m)	x inertial position
YII	y_I	(m)	y inertial position
HAGL	h_{AGL}	(m)	altitude above launch site
RIHH	i_H	(deg)	canard deflection
DELTAA	δ_A	(deg)	aileron deflection
HINT	h_{int}	(sec)	DGEAR chosen integration time step (see Reference 5)

3.4.3 Plotting Data Output File

Direct access output files for the response data to be read by the plotting package PLTSIM are also generated, one for each case of a given run. Within the program these files are associated with Fortran unit numbers 21 and on (the first with unit number 21, the second with unit number 22 and so forth up to and including 20 + NCASE).

Each record of this file is of length IVARL*2 and the blocksize is IVARL*4. Here IVARL is the number of parameters, including time, to be recorded for subsequent plotting by PLTSIM. For the current version of the program for ROBOT-X, IVARL = 182.

The initial size of the file is IRELIM records where

$$\text{IRELIM} = (\text{IDINT}(\text{TTMAX}/\text{RDTINT}) + 1)/10$$
$$\text{RDTINT} = \text{RDELT}/\text{ISPSO}$$

and RDELT, ISPSO and TTMAX are inputted from the master data input file (Section 3.3.1) as defined by the user.

The associated variable for these direct access files is IASV.

If the simulation for a given case terminates prematurely (for any reason) the direct access file is complete up to and including the last complete time point and is available for plotting through PLTSIM.

Each direct access record other than record one contains the complete set of parameters (SX(I), I = 1, IVARL) associated with the time point SX(1). Record two contains the first time point, record three the second and so forth up to IOUTC time points. SX is a single precision real array whose contents are defined in detail in Table 6. For the current versions of FLISIMV1 and FLISIMV2, SX is dimensioned as SX(200) although for the ROBOT-X application IVARL = 182, i.e., only the first 182 array elements of SX are used.

Record 1 contains a number of parameters that identify the plotting file and allow PLTSIM to read and plot the data as requested by the user. These parameters, in order of appearance on record 1 are as follows:

IRELIM	INTEGER*4	See previous paragraphs this section.
IVARL	INTEGER*4	See previous paragraphs this section.
(NLABEL(I), I=1, 20)	INTEGER*4	See Section 3.3.1.
ICASE	INTEGER*4	See Section 3.3.1.
IOUTC	INTEGER*4	Total number of time points recorded for the case.
RDELT	REAL*8	See Section 3.3.1 (sec).
ALTØ	REAL*8	See Section 3.3.1 (m).
KSTAG	INTEGER*4	See Section 3.4.1.
AVVEL	REAL*4	See Section 3.4.1.
AVMACH	REAL*4	See Section 3.4.1.
AVALT	REAL*4	See Section 3.4.1.
RMAVEL	REAL*4	See Section 3.4.1.
RMAMAC	REAL*4	See Section 3.4.1.
RMAALT	REAL*4	See Section 3.4.1.
SX(1)	REAL*4	See Section 3.4.1.
SX(47)	REAL*4	See Section 3.4.1.
(TISTAG(I), I=1, KSTAG)	REAL*8	See Section 3.4.1.

3.5 FLISIMV1 Algorithm

The execution algorithm used in FLISIMV1 is as follows:

1. Read the master data input file.
2. Initialize aerodynamic data file interpolation subroutines INLAAD, INLOAD and SETUP and read in coefficient reference lengths and areas, and moment origin reference geometry.
3. Initialize wind modeling subroutine WINDI.
4. Initialize case counter ICA to 0.
5. Begin case loop (Label 1100).
6. Advance case counter by one.
7. Call INPCHA. This subroutine contains common statements for all master data input file parameters and allows the user to redefine master data parameters on a case by case basis.
8. Output master data parameters to the formatted output file.
9. Compute initial airspeed components in body-axes given the initial Euler angles, the initial linear position, the initial ground velocity components in F_I and the initial wind velocity components in F_I.
10. Define the autopilot update time

    RDTINT = RDELT/ISPSO.
11. Open the direct access file onto which the unformatted plotting output will be stored. This file is associated with Fortran unit

    ILUN = ICA + 20

    and the associate variable IASV. Initialize IASV to 2.

12. Initialize the mass computation subroutine MCHRØ.
13. Compute the initial NRE rocket motor thrusts and store in array ATT, and the initial body-axes thrust components (TTXBB, TTYBB, TTZBB) by calling ROCTHR.
14. Call ROCMOM to compute the initial rocket motor induced moments in F_B as (LLBBRR, MMBBRR, NNBBRR).
15. Initialize the output time step counter IOUTC to 0.
16. Define the initial integration step size to

    HINT = RDTINT/10.

17. Define the maximum number of response output time points as

    NOMAX = IDINT (TTMAX/RDTINT) + 2.

18. Initialize DGEAR parameter index to 1 (see Reference 5).
19. Initialize the autopilot model by calling PILOT with T = -100.
20. Output response summary table header to the formatted output file.
21. Initialize average and maximum characteristic parameters.
22. Define initial autopilot control commands (call PILOT, T = T1).
23. Define initial actual control positions by CALL SERVO.
24. Begin simulation loop 401 with counter ICOUNT.
25. If ICOUNT = 1, omit integration loop 52.
26. Set output time point to RDTINT * ISPSO1 during or near rocket motor stage firings and to RDTINT * ISPSO otherwise.
27. Begin integration loop 52.
28. Advance T2 to

T2 = T2 + RDTINT.

29. Execute the differential equation solver DGEAR, and store resulting response at time T2 in array TX. On exit from DGEAR, T1 = T2.
30. If DGEAR error parameter IER is greater than 128 (see Reference 5), exit simulation and integration loops and go to 2001. Output key DGEAR parameters for error analysis to formatted output file. Proceed to next case if ICA < NCASE. Otherwise end program run.
31. Update autopilot control commands (PILOT).
32. Update servo positions (SERVO).
33. Compute thrust forces and moments (ROCTHR and ROCMOM).
34. Compute vehicle mass characteristics at T = T2.
35. End integration loop 52 if T2 = next output time step. Otherwise return to step (27).
36. Compute output parameters SX (see Table 5) and convert radian parameters to degrees.
37. Update coverage totals and maximum of selected parameters.
38. Output critical parameters to screen.
39. Output response summary table data for current time points to the formatted output file.
40. Increment the output counter IOUTC to

    IOUTC = IOUTC + 1

    (now indicating current number of completed output points).
41. If the vehicle's altitude ASL is less than ALTST, perform a linear interpolation for all parameters in SX to ALTST.
42. Set IASV to IASV = IOUTC + 1.
43. Write (SX(I), I = 1, IVARL) to the direct access unformatted plotting file's record IASV.

44. Update record 1 of this file with current case data.
45. Exit simulation loop (to to 801) if T2 ≥ TTMAX or if the altitude ASL is < ALTST. Otherwise return step 24, or if the pitch attitude absolute value is >85°, i.e., the pitch singularity at 90° is approached.
46. If IOUTC ≠ 1, compute case averages AVVEL, AVMACH and AVALT.
47. Call PRINOP to output to formatted output file.
48. Output mission summary to formatted output file.
49. Store case summary in array RUNSUM.
50. Update record one of direct access unformatted plotting data file.
51. If ICA < NCASE, repeat case loop (return to step 5).
52. Output case summaries to the formatted output file.
53. STOP execution.

3.6 Aerodynamic Model

The aerodynamic model used in FLISIMV1 is based on the AERO1 model (see Section 3 of Volume 1).

Provision is made for parametrically varying the input aerodynamic characteristics. This may be conveniently done through variation of selected parameters in the array ACPAR or through the arrays RMLON, RBLON, RMLAT and RBLAT. All of these arrays are defined in the master data input file (see Table 3).

Data is provided in the form of look-up tables, and linear interpolation is used to determine intermediate variables. The interpolation independent variables are thrust coefficient C_T, Mach number M_a and fuselage reference lines angle of attack α_f.

Provision is made for modeling the drogue and main canopy aerodynamics through parameters ACPAR(19) through to ACPAR(22). If IDROGU = 1 and IMAINP = 0, then the drogue parachute is assumed to be deployed. If IDROGU = 1 and IMAINP = 1, then the main canopy is assumed to be deployed except if $C_{D_{main}} \leq 0$ in which case a main canopy deployment failure is assumed (the vehicle comes down on the drogue). The flags IDROGU and IMAINP are passed from the autopilot module SUBROUTINE AUTCOM through COMMON PARDYN.

3.7 Thrust Model

The thrust model used in FLISIMV1 is defined in Section 6 of Volume 1. The thrusts are defined through SUBROUTINE ROCTHR and thrust moments through SUBROUTINE ROCMOM.

The thrust characteristics of an individual rocket motor are defined in arrays TIME(I) and THRUS(I). Off-nominal characteristics for TIME and THRUS may be obtained through the multipliers SCAT and SCATT. The NPOINT TIME and THRUS data points and SCAT and SCATT are defined in the master input data file.

Linear interpolation is used to obtain intermediate values in the look-up tables. TIME and THRUS need not have evenly spaced points.

The array TISTAG(I) defines the time of ignition of the i-th rocket motor stage. This array is checked through the KSTAG stages fired up to and including time T. The actual time of ignition is defined in the autopilot module AUTCOM and passed through ROCTHR through COMMON REPARM. KSTAG is also set in AUTCOM and is passed to ROCTHR through its calling argument list.

As well as total body-axes thrusts, ROCTHR also computes the individual thrust of each of the NRE rocket motors. These individual thrusts are then used in ROCMOM to compute the rocket motor thrust moments.

3.8 Mass and Moment of Inertia Models

The mass and moment of inertia models are defined in Section 4 of Volume 1.

Computation of initial mass and moment of inertia characteristics are conducted through a preliminary call to MCHRØ. Subsequently to this initial call, time varying characteristics are computed by using the entry point MCHR.

As for the thrust model, the time of ignition of the i-th rocket motor state TISTAGE(I) is passed from the autopilot module SUBROUTINE AUTCOM through the common statement labeled REPARM.

3.9 Servo Model

Control servo dynamics are modeled as first order lags (see Section 4 of Volume 1) in SUBROUTINE SERVO.

The time constants, in seconds, of the five servo channels are defined in PAR(1) through PAR(5) (see Table 7). They are as follows:

TTDTT	Throttle time constant.
TTDEE	Elevator time constant.
TTIHH	Canard time constant.
TTDAA	Aileron time constant.
TTDRR	Rudder time constant.

3.10 Homing Beacon Model

The homing beacon signal is modeled through SUBROUTINE BEACON. In its current form this subroutine models a nondirectional beacon. It provides an output ETA that represents the bearing of the beacon relative to the vehicle. ETA is defined as per standard manned aircraft convention, i.e., it is measured clockwise from the nose of the aircraft and is in the range

$$0 \leq ETA < 2\pi.$$

3.11 Wind and Turbulence Models

The mean wind and turbulence characteristics are modeled as per Section 4 of Volume 1. This model is implemented through SUBROUTINES WINDI and TWINDR.

SUBROUTINE TWINDR computes the body-axes components of the total rate of change derivative of wind velocity as seen by the aircraft centre-of-mass using a centred finite difference scheme. The finite difference steps for x_I, y_I, altitudes and time are defined in the master input data file (see Table 3) through array elements PAR(6) to PAR(9) respectively (see Table 7) and are passed to TWINDR through COMMON RPET2.

The wind velocity components in body-axes are computed by calling entry point WIND in SUBROUTINE WINDI. WINDI must be called only once, and this call must be made prior to any calls to TWINDR or WIND.

The call to WINDI initializes wind model parameter values and opens and reads into arrays the user defined turbulence files. These three turbulence files are assumed to be sequential, unformatted files, and each record is 1000 bytes long. For ROBOT-X simulations, the turbulence files were limited to have a maximum of 10000 data points and were generated using filtered white noise through the digital filter algorithm described in Reference 7 and implemented through the DRES software package FILTERV2.

These turbulence files are generated off-line to the simulation and may be generated using any suitable user selected technique. The first record, of each file contains a number of specification parameters associated with the algorithm of Reference 7 as follows:

1. NFIL
2. MFIL
3. FCUT - Cut-off frequency of the filter used to colour the white noise (Hz).
4. ASEED - Random number seed used to generate the white noise.
5. FSAMP - Sample frequency of the filtered white noise stream (Hz).
6. TTOT - Total time of the filtered white noise stream (sec).

The three turbulence files are associated with Fortran unit numbers 10, 11 and 12 respectively.

The flag ITWIND, inputted as PAR(10), determines the type of wind model as follows:

1. ITWIND = 0 — No wing or turbulence.
2. ITWIND = 1 — Shear and downdraft profiles plus turbulence.
3. ITWIND = 2 — Constant mean wind plus turbulence.
4. ITWIND = 3 — User defined wind profiles as a function of h_{ASL}.

The characteristics of the wind model are determined through the master input data file by setting parameters in array PAR (see Table 7). The shear and downdraft profiles are as per the model specified in Section 4 of Volume 1. This model was originally developed as part of the work described in Reference 1.

The downdraft portion of the model allows for multiple downdraft cells. The arrays associated with this are currently dimensioned to allow up to 2 cells.

Within the constraints of its calling and common structure, SUBROUTINE WINDI is user definable, thus permitting changes and/or additions to the current wind model. Such changes could imply changes to the number and/or meaning of some of the elements of array PAR. The current dimension limit for this array is 200 elements.

3.12 Autopilot Model

The subroutines associated with modeling the autopilot and sensor characteristics are all user-defined and provided as part of model FLISIMAP.

FLISIMV1 calls SUBROUTINE PILOT once every RDTINT seconds, i.e., every autopilot update cycle. SUBROUTINE PILOT is the only FLISIMAP subroutine that must be available to FLISIMV1 with compatible calling arguments and common structure. All other subroutines that are given as part of FLISIMAP are specific to the ROBOT-X autopilot model.

For ROBOT-X use the FLISIMAP model is of sufficient complexity to model the following features:

1. All control modes and algorithms including transition algorithms.
2. All estimation algorithms including the complementary filter algorithms and air data algorithms.
3. Computational lags.
4. Sensor and measurement nonidealities for most sensor types and radios in use or under consideration for use in ROBOT-X.
5. Angle-of-attack limiting logic.
6. Parachute deployment logic.
7. Control deflection limiting logic.

Other features that are not currently incorporated, but which could be modeled include integer arithmetic and sensor noise.

The ROBOT-X FLISIMAP subroutines are structured such that the majority of the parameters that are altered by the user are defined in the block data subroutine AUTPIL.

The definitions assumed for the estimated state vector XEST, the auxiliary state vector XAUXES and the sensor vector XSEN are given in Tables 8, 9 and 10 respectively. The estimated state vector components correspond to the state vector X in use in the differential equation model. Both X(12) and XEST(12) refer to altitude referenced above the launch site elevation.

3.13 Differential Equation Solver

The IMSL subroutine DGEAR is used to solve the twelfth order nonlinear system of differential equations. DGEAR is documented in Reference 5.

DGEAR allows a number of solution modes as determined by the parameters METH and MITER (see Table 3). METH = 1 sets DGEAR to use an Adams-Moulton predictor-corrector method while METH = 2 sets DGEAR to use Gear's backward differentiation method (Reference 8).

For ROBOT-X simulations, the following DGEAR parameters were found to be effective:

1. METH = 1
2. MITER = 0
3. TOL = 1.0×10^{-4}.

For simulations in which speed of computation is essential, DGEAR is not particularly efficient, in part due to its variable step size and complexity of options. A fixed step size fourth order Runge-Kutta method is usually satisfactory for flight vehicle simulation applications and could be used by replacing the call to DGEAR in FLISIMV1 with a corresponding call to a Runge-Kutta routine.

3.14 Miscellaneous Remarks

3.14.1 Variation of Parameters in Multiple Runs

The structure of FLISIMV1 and FLISIMV2 permits the sequential simulation of multiple cases in a given run by setting NCASE to a value greater than 1 (see Table 3).

Two methods are provided for varying parameters from case to case. The first consists of making changes associated with the master data file by modifying the user-defined SUBROUTINE INPCHA. The case counter ICA can then be used to make changes for specific cases through user-written conditional statements. The changes are passed on to other program modules through the common statements.

The second method, intended for changes to data not in the master input data file (e.g., in the user-defined autopilot modules), is implemented by making the case counter ICA available to all user-defined subroutines. The user may then implement case specific logic based on ICA.

It should be noted that the master input data file is read only once for a given run of FLISIMV1 or FLISIMV2 regardless of the number of cases in the run.

3.14.2 Summary of Fortran Unit Number Assignments

FLISIMV1, in its current configuration, utilizes the following Fortran unit number assignments:

1.	Master input data file	- Unit #5
2.	Formatted output file	- Unit #6
3.	Screen output	- Unit #7
4.	Turbulence input file 1	- Unit #10
5.	Turbulence input file 2	- Unit #11
6.	Turbulence input file 3	- Unit #12
7.	Longitudinal aerodynamic data	- Unit #15
8.	Lateral aerodynamic data	- Unit #16
9.	Plotting files	- Unit #21 through to Unit #20 + NCASE

3.14.3 Asymmetric Thrust Conditions

Asymmetric thrust conditions may be modeled in three ways:

1. Through the master data input file array RDAT by defining a rocket motor configuration that is asymmetric to the degree desired.
2. By defining the vehicle mass characteristics such that the resultant thrust axis is not through the centre-of-mass.
3. By setting the array element of IFIRE (see Table 3) corresponding to a specific rocket motor to the negative of its value. This simulates a total failure of that motor.

3.14.4 Reference Frames and Geometry

Volume 1 describes in detail the reference frame conventions and associated geometry.

3.14.5 Dimension Limits

A number of limits are implied by the dimensions chosen for selected arrays. Many of these are arrays that are input from the master data file, and whose current dimensions are given in Table 3.

Appropriate modifications must be made to the program before these limits may be exceeded.

A partial summary of these limits is given below:

1. Up to 10 rocket motor stages may be specified with up to a total of 25 rocket motors.
2. The thrust versus time data arrays may have up to 201 points.
3. Not counting rocket motors, up to 20 components may be defined in specifying the vehicle mass and moment of inertia characteristics.
4. Up to and including 200 parameters may be defined and stored as a function of time as the simulation progresses. In the current configuration of the software, 182 elements are being utilized.
5. Up to and including 30 parameters may be input as part of the array ACPAR.
6. Up to and including 200 parameters may be input as part of the array PAR.
7. There can be up to 20 sensor variables and 60 auxiliary variables as part of the autopilot module.

4. FLISIMV2

A Fortran source listing not including IMSL subroutines is given in Appendix B. Program modules that are common with FLISIMV1 are listed in Appendix A and are not repeated in Appendix B.

The listing is extensively commented and provides additional user information to that given in this section.

4.1 Structure

FLISIMV2 algorithms are similar to those of FLISIMV1. The two programs share many subroutines, and are generally user transparent.

The main differences between FLISIMV1 and FLISIMV2 arise in the form of the aerodynamic model that is utilized. FLISIMV2 was developed specifically for directly utilizing wind tunnel aerodynamic data that was available for ROBOT-X from tests conducted at the National Aeronautical Establishment in 1982, 1984 and 1985 in the 5-foot blowdown trisonic wind tunnel facility. Although FLISIMV1 can also utilize an aerodynamic model based on wind tunnel data, this model is developed in a way that eliminates some of the nonlinearities associated with the measured data. FLISIMV2 was thus developed in order to partially address these limitations through more direct use of wind tunnel data. It has been primarily used to validate the less sophisticated aerodynamic model used in FLISIMV1.

FLISIMV2 has been structured to be as modular as possible in order to facilitate program readability and modification to different flight simulation applications.

Eight program segments consisting of several subroutines each constitute FLISIMV2, as follows:

a) FLISIMV2 - Program modules unique to FLISIMV2.

b) FLISIMCO - Program modules common to FLISIMV1 and FLISIMV2.

c) FLISIMUS - Program modules that occasionally require modification by the user (e.g., wind modeling subroutines).

d) FLISIMAP - Program modules that model the autopilot and sensor characteristics.

e) AEROATMOS - A number of subroutines from the DRES/SIG library AEROATMOS are used. This library contains subroutines common to a number of DRES/SIG flight dynamics packages and has been built up since 1981.

f) MATHPACKP - A number of subroutines from the DRES/SIG mathematical and utility library MATHPACKP are used. This library contains subroutines common to a number of DRES/SIG flight dynamics packages and has been built up since 1981.

g) WTDATA - Subroutine WTDTCON from the DRES/SIG wind tunnel data analysis library WTDATA is utilized. This library contains subroutines common to a number of DRES/SIG software packages and has been built up since 1982.

h) IMSL - A number of subroutines from the IMSL library (Reference 5) are used.

FLISIMV2 contains, as one of its modules, the main program that controls program execution.

FLISIMCO, FLISIMUS, FLISIMAP, AEROATMOS and MATHPACKP modules have been extensively documented in Section 3 with FLISIMV1 documentation. It is stressed that the user-defined modules FLISIMUS and FLISIMAP can be used with either FLISIMV1 or FLISIMV2.

FLISIMV2 requires several input files, as follows:

a) A master data file.
b) A longitudinal aerodynamic characteristics data file.
c) A lateral aerodynamic characteristics data file.
d) Wind tunnel data files.

The input files associated with (a), (b) and (c) are organized identically to those utilized in FLISIMV1, and are documented in Section 3.3. The number of wind tunnel data files required will depend on the particulars of the aerodynamic model. In the current version of FLISIMV2 for WTDATA subroutines configured for ROBOT-X data analysis, 30 files are required.

The wind tunnel data input files are documented in Section 4.3.

FLISIMV2 generates the same output files as FLISIMV1 (see Section 3.4). These are as follows:

a) A formatted output file suitable for printer output.
b) A screen output file allowing interactive assessment of simulation progress.
c) NCASE direct access files containing the data required by PLTSIM in plotting the results of FLISIMV2. NCASE corres-

ponds to the number of cases computed in a given FLISIMV2 execution.

4.2 Description of Program Modules

Including the main program and all library subroutines not intrinsic to VAX Fortran, the FLISIMV2 software package consists of 51 modules grouped as follows:

a) FLISIMV2 - Three including the main program.
b) FLISIMCO - Eleven subroutines.
c) FLISIMUS - Four subroutines.
d) FLISIMAP - Ten subroutines and one block data module.
e) AEROATMOS - Six AEROATMOS library subroutines.
f) WTDATA - One WTDATA library subroutine.
g) MATHPACKP - Five MATHPACKP library subroutines.
h) IMSL - Ten IMSL subroutines.

Program modules associated with FLISIMV2 are described in Table 11. Only those modules not associated with FLISIMV1 are included.

4.3 Input Data Files

4.3.1 Master Data File

The master data file is a formatted data file that must be provided by the user. FLISIMV2 uses an identical master data file to that used by FLISIMV1 (see Section 3.3.1 and Tables 2 and 3).

4.3.2 Aerodynamic Model Input Data Files

Several aerodynamic data input files are required.

The first two files correspond, respectively, to the longitudinal and lateral aerodynamic characteristics, and are identical to the files documented in Section 3.3.2 for FLISIMV1. The remaining files are wind tunnel test data files and are unique to FLISIMV2.

The input format currently required may be inferred from subroutine WTDTCON, and will depend on the format of the wind tunnel data that is available.

4.4 Output Files

The FLISIMV2 output files are identical to those generated by FLISIMV1. The latter are documented in Section 3.4 (subsections 3.4.1, 3.4.2 and 3.4.3).

4.5 FLISIMV2 Algorithm

The execution algorithm used in FLISIMV2 is the same as that used in FLISIMV1 (see Section 3.5) with the exception that step 2 should read as follows:

2. Initialize aerodynamic data file interpolation subroutines INLAAD, INLOAD, SETUP and WTDTCON and read in coefficient reference lengths and areas and moment origin reference geometry.

4.6 Aerodynamic Model

The aerodynamic model used in FLISIMV2 is based on the AERO2 model (see Section 3 of Volume 1).

A number of aerodynamic model parametric variation considerations documented with regards to FLISIMV1 in Section 3.6 also apply to FLISIMV2.

4.7 Thrust Model

The thrust model used in FLISIMV2 is identical to that used in FLISIMV1. The latter is documented in Section 3.7.

4.8 Mass and Moment of Inertia Models

The mass and moment of inertia model used in FLISIMV2 is identical to that used in FLISIMV1. The latter is documented in Section 3.8.

4.9 Servo Model

The model of control servo dynamics used in FLISIMV2 is identical to that used in FLISIMV1. The latter is documented in Section 3.9.

4.10 Homing Beacon Model

The homing beacon model is identical to that used in FLISIMV1 (see Section 3.10).

4.11 Wind and Turbulence Models

All of the discussion of Section 3.11 regarding FLISIMV1 wind and turbulence models also applies to FLISIMV2.

4.12 Autopilot Model

All of the discussion of Section 3.12 regarding the use of the FLISIMAP autopilot model with FLISIMV1 also applies to FLISIMV2.

4.13 Differential Equation Solver

The discussion of Section 3.13 with regards to FLISIMV1 also applies to FLISIMV2.

4.14 Miscellaneous Remarks

All of the miscellaneous remarks of Section 3.14 for FLISIMV1 also apply to FLISIMV2.

In addition to the Fortran unit number assignments of Section 3.14.2, FLISIMV2 assigns unit numbers 21 through to 51 to the wind tunnel data files. These are only open while the aerodynamic data is loaded into dynamic arrays, and thus no unit number conflict arises with the plotting files associated with unit numbers 21 through 20 + NCASE.

5. PLTSIMV1

The PLTSIMV1 (PLoT SIMulation, Version 1) was originally developed in 1983 for plotting support for an early ROBOT-X six degree-of-freedom code referred to as NLSRBX. In this early form PLTSIMV1 was referred to as PLTNLSRBX.

This software has evolved into a user-friendly package that is fully integrated and compatible with FLISIMV1 and FLISIMV2.

PLTSIMV1 is written in VAX-11 Fortran (Reference 4) and has been developed around Fortran callable PLOT 10 graphics subroutines (References 2 and 3). No knowledge of FORTRAN or of PLOT 10 is required to run this interactive program.

PLTSIMV1 must be used with a terminal that has graphics capability. It has been successfully demonstrated on Tektronix 4105, 4107, 4112 and 4114 terminals as well as DEC VT100 terminals with the Retrographics feature.

In general, the PLTSIMV1 software provides all of the information necessary to specify a plot in an interactive fashion through a computer terminal. Prompts are given for all of the required parameters, and thus the user should not have to regularly refer to this manual to successfully operate the program.

A Fortran source listing not including PLOT 10 subroutines is given in Appendix C.

5.1 Structure

PLTSIMV1 has been structured to be as modular as possible in order to facilitate program readability and modification to different flight simulation applications.

Two program segments consisting of several subroutines each constitute PLTSIMV1, as follows:

a) PLTSIMV1 - A number of Fortran modules unique to PLTSIMV1.

b) PLOTPACK - A number of subroutines from the DRES/SIG plotting library are used. This library contains subroutines common to a number of DRES/SIG flight dynamics plotting packages and has been built-up since 1981.

c) PLOT 10 - A number of subroutines for PLOT 10 (References 2 and 3) software are used.

PLTSIMV1 contains, as one of its modules, the main program that controls program execution.

Program modules are described in more detail in Section 5.2.

PLTSIMV1 requires up to ten input files that define the up to ten simulation cases to be plotted. These input data files are documented in Section 5.3.

PLTSIMV1 outputs all plotting commands to the user's screen. In the current version of the program, no other output files are generated.

5.2 Description of Program Modules

Including the main programs and all library subroutines not intrinsic to Fortran 77, but not including PLOT 10 software modules, the PLTSIMV1 software package consists of 16 modules grouped as follows:

a) PLTSIMV1 - Two modules including the main program.

b) PLOTPACK - Fourteen subroutines.

PLTSIMV1 calls 32 different PLOT 10 subroutines and three PLOT 10 functions. These PLOT 10 functions and subroutines call other PLOT 10 software as required.

By making a number of minor modifications, PLTSIMV1 may be modified by the user to handle greater numbers of data points per plotting file as well as different types and number of plotting variables.

The program modules are described in detail in Table 12.

5.3 Input Data Files

The input data files for PLTSIMV1 are the output plotting files generated by FLISIMV1 and/or FLISIMV2. These files are direct access unformatted files whose structure is fully documented in Sections 3.4.3 and 4.4.

Up to ten plotting files may be simultaneously utilized by PLTSIMV1. The software is transparent as to the source of these files, i.e., any combination of up to 10 FLISIMV1 and FLISIMV2 files may be utilized.

The input data files are read on Fortran unit numbers 21 through to 30 inclusive.

5.4 Output Files

PLTSIMV1 outputs all plotting commands to the user's screen. In the current version of the program no other output files are generated.

5.5 PLTSIMV1 Operation

PLTSIMV1 first asks the user if he would like to plot. Typing "Y" and pressing return will lead to the next stage of the program. An answer "N" will cause program termination. Any other action will prompt an inappropriate response message and a repetition of the initial question.

The program then asks, "How many files do you want access to during this plotting run?". This refers to the number of direct access files to which the command procedure has assigned logical unit numbers. If, say, the user answers "5", the program assumes that it can read data in from five files with logical units numbers 21 through 25 (20 is added to the file number to obtain the unit number). In its prompts the program will refer to these files as 1 to 5. While the user can indicate that he wants access to up to 99 data files during the plotting run, he can read no more than 10 of these for one plot. Also, the command procedure used to run this program in conjunction with the flight simulation software can set up logical unit numbers for no more than 10 files (representing logical unit numbers from 21 to 30).

The user will then be asked, "How many files do you want to read for the next plot?". Continuing with the previous example, let's assume the user replies "3". The program will then ask, "Input the numbers 1 to 5 of the 3 files you want to read.". (The numbers here underlined will vary with the answers to the previous questions.) If, say, the user responds with the numbers "4", "1" and "3", the program will subsequently read in data from logical unit numbers, 24, 21 and 23. It will refer to the corresponding plotting files as numbers 4, 1 and 3, in that order, in the rest of the prompts until the files are changed. This gives the user flexibility in reading data from any of the files that have been set up with the command procedure. Had a number greater than "5" been given in response to the second last question, it would have been ignored and a default value of "5" assumed. Thus the program reads all the files set up with the command procedure, and does not ask for the numbers of individual files.

A listing of all of the program variables is then printed on the screen. The user types in the number of the variable to be plotted as the abscissa. The program will list this variable's range for each of the files specified above, and ask the user for the range of values to be graphed. The minimum and maximum values of the abscissa variable within the user selected range determine the data limits for the x-axis. Thus the user does not select the abscissa range directly. If the prompts list an abscissa range from "0" to "40", and the user selects a range from "20" to "999" for graphing, the program will truncate the excess range from "40" up.

If the user inputs a range from "9" to "9", the program, as indicated, will allow the user to reselect the abscissa variable. Thus, if the variable chosen is found to have no non-zero values, it need not be graphed.

Otherwise, the program then checks each of the files for the number of data points which fall within the user selected range, and asks for the increment between the points to be plotted. If the user answers "2", only every second data point will be displayed on the graph. An answer of "99" will allow the user to reselect the range. This is useful if the file has an insufficient number of data points within the initially selected range to make a plot meaningful.

Next, the program will calculate the maximum number of variables that can be plotted from the specified files without the number of curves exceeding ten. In our example, the maximum number of variables is three, since with three files, this would lead to nine curves.

Given this information, prompts ask the user to input the actual number of variables he does want to plot. If his answer is greater than the indicated maximum, it is ignored and the question is repeated.

Otherwise, the program redisplays the list of variables on the screen, and prompts the user to input the number of the variables that he wants plotted.

The computer reads all the numerical data that is typed into the terminal with list-directed formatting. Thus numbers need not occupy any particular space on a line, nor any particular line. The only requirement is that sets of numbers be entered with their elements separated by at least one space, and/or one comma. If the necessary number of values have not been added, the computer will wait, even after a pressing of the "RETURN" key.

Character data (a "Y" or "N" response, for example) has to entered in the first space of the first line following the prompt, or it will not be read.

Continuing, the program asks the user if he wants different symbol types. A "Y" response will produce a listing of the available symbols and their corresponding numbers (see Table 13). Prompts ask the user to enter the numbers of the desired symbols for each file in turn. A similar process is repeated for line types (Table 14). If the user does not use these two options, no symbols will appear on plotted points, and all lines will be solid.

The plotting package will now display the graph on the screen. A chart showing the line and symbols, and their corresponding files and variables, will be listed above the graph to provide identification to the curves. The files will be identified with case numbers, read from the first record of the plotting files. The program can display case numbers up to 9999, printing two asterisks ("**") if the case number is larger. The user has an opportunity to examine the graphs for possible changes, or to make a hard copy.

By pressing any alphanumeric key other than "N", the user can move to the 'Option' part of the program. A menu of possible changes will be displayed. This feature is designed so that the user, after obtaining a basic plot with a minimum number of prompts, can view it, and decide what changes would be appropriate. He then inputs the appropriate option number. The user can go on making changes to the graph (or drawing new ones) indefinitely. Use of the option feature simply causes the program to recircle back through itself.

The PLTSIMV1 menu options are described below by option number.

1. New Graph

   All of the graph parameters will revert to their initial default values, and the graph specification process will start again with the question "How many files do you want to read for the next plot?". The program will then proceed through all of the questions described previously.

2. Change Abscissa Variable

   This option allows the user to graph the ordinate variables of the last graph against a different abscissa. Questions relating to the new abscissa variable and its range, and the increment between symbols placed on plotted points, will be repeated. Previously specified line and symbol types will be retained. Options 7, 8, or 11, if applicable, will be nullified.

3. Change Ordinate Variables

   This option retains the absicssa and its corresponding range from the last graph, and asks the user how many new variables are to be plotted against it. The variable list is redisplayed and the user inputs the numbers of these new variables. Line and symbol types have to be respecified only if the number of variables changes. Option 11, if applicable, will remain in effect, but options 7 and 8 will not.

4. Change the Range of the Abscissa

Only questions relating to the range of the abscissa will be repeated, and the prompts will display the previously chosen range. If symbols have been asked for, the interval between their placement on plotted points must be respecified, since the number of plotted points may change substantially with a change of range. No other characteristics of the graph will be altered, and options 7, 8 and 11 remain in effect.

5. Change Line Types

The user can rechoose line types after displaying the graph.

6. Change Symbol Types

The user can rechoose symbol types after displaying the graph.

7. Create a Second "Y" Axis

This option allows the user to show variables of two widely different magnitudes effectively on one graph by putting a numerical scale on each of the two vertical edges. After prompting, the user tells the program which curves will be referred to the second scale (on the right side). Then the program routine which calculates y-axis data ranges, and thereby the appropriate axis scale, is free to determine values for each group of curves separately. This prevents curves whose data values are relatively small from being shown only as a cluster of points along the x-axis because of incompatible magnitudes.

The program first asks the user how many of the curves he wants referred to the second scale (unless there are only two curves, in which case the program assumes an answer of one). An answer greater than the total number of curves will cause the program to revert to a display of the option 'menu', and cancellation of this option choice. Otherwise a table is displayed showing all of the curves by file number and variable, and listing a two digit number for each. The user inputs as many of these numbers as there are to be curves referenced to the second scale. The first digit of these numbers refers to a particular file, and the second to a particular variable.

The program will then display the new graph on the screen, with all of the other characteristics of the previous graph retained. It will place an "L" or an "R" (indicating left or right) at the end of the sample line of each curve label, so the viewer knows which scale to refer each curve to.

If there is only one curve on the graph, use of option 7 will cause it to be displayed with only a right hand y-axis scale. The program will draw neither an x-axis nor any grid lines, and will ask none of the above questions.

8. Add a Title

Prompts will tell the user to type in the desired title followed by a semi-colon (';') to indicate the end. The program will automatically count the number of characters in the title (the maximum of which is 80), and center it above the graph. Graph screen limits and variable label placement will be adjusted accordingly. The user can replace one title with another by simply using option 8 twice.

9. Eliminate Label Round-Off Error

The PLOT 10 plotting package determines internally the values of major tick mark labels, and the number of character spaces allotted to print them. On occasion, it will not allot enough spaces, causing label round-off. This condition can be easily detected because adjacent labels will be either identical or appear unevenly incremented.

To correct this, option "9" will automatically increase the maximum possible width of the axis labels by two spaces, by incrementing the appropriate variable in the PLOT 10 parameter table. The graph should then be displayed with the ambiguity resolved. Unfortunately, for reasons peculiar to the PLOT 10 package itself, the graph also occasionally returns unchanged.

10. Change Major Tick Mark Intervals

The PLOT 10 plotting package determines automatically the number of major tic mark intervals for each axis. If the user wishes to change them, he can select option 10, and it will display the present number of intervals along with prompts asking the user how many he would prefer.

11. Replot Date in Time Sequence

Use of this option will produce a change only if time is not the abscissa variable.

The data files are ordered chronologically, since the flight simulation program outputs data in records representing sequential moments in time. In initially producing the graph, the program searches out those records of the file in which the abscissa variable falls within the user selected range. It then sorts the data to be graphed from these records in order of the value of the abscissa variable. This means that on the graph the plotting package will join the data points from left to right by the plotted lines, which is not necessarily the order in which they were output by the simulation program.

Option 10 causes the appropriate data to be reread and then suppresses the sorting of it into the order of the abscissa variable. Thus the plotted data points will be joined in a chronological order (the order in which they were output). This is physically more meaningful for some types of data. For other types of data it will produce 'switchbacks' that make the graph difficult to read.

The effect of this option will be retained even through subsequent use of options 3, 4, 5, 6, 9 and 12. Use of any other option will nullify its effect.

12. Change Data Files

This option allows the user to change the files from which the variables that he has specified are read. The program calculates the maximum number of files from which it can read the chosen variables without the resulting number of curves exceeding ten. Displaying this maximum (or the number of files that have been specified as accessed during this plotting run, whichever is the lesser), the program asks the user how many new files he actually wants to read. If his answer is greater than the maximum, his answer is ignored and the maximum assumed.

The program then asks the user to input the numbers of these files. Because new data is being read for the abscissa, the user must respecify its range and the increment between symbols, if any. The program will give the appropriate prompts. Line and symbol types have to be rechosen only if the number of data files to be read is changed. Option 11, if applicable, will remain in effect, but options 7 and 8 will not.

5.6 PLTSIMV1 Revisions

As noted previously, the PLTSIMV1 plotting software can be adapted to plot data from any simulation program that outputs information into a direct access file, if each record represents a sequential moment in time. The following section outlines how to do this, or how to adjust PLTSIMV1 to reflect changes of the FLISIMV1 or FLISIMV2 programs. Thus this section is intended only for users who may wish to revise PLTSIMV1. A knowledge of VAX 11 Fortran is necessary, and the user should also consult the comments in the program listing.

5.6.1 Changing the Maximum Number of Data Points

The simplest revision that the user can make to the PLTSIMV1 program is to change the maximum number of records that it can read from a direct access file. This may be necessary if the user simulates extraordinarily long flights or decreases the time intervals at which it outputs data. The present maximum of 3000 records is large enough for most runs, but is necessarily a compromise between flexibility and the desire to minimize memory requirements. To alter the maximum, the user must change the appropriate dimension (that is to say the one that is 3000 at present) of the arrays XDATA, YDATA, AERDAT, and LRANGE in subroutines PLOT, ABSRANGE, SORT, and LIMITSORT.

The first record of each direct access file contains file and case parameters rather than actual data, and the program reads this record separately. The parameter in the twenty-fourth field of the first record, read into IOUTC, is equal to the number of filled data records in the file being read. The array IOUTC, each element of which contains the length of one of the different plotting files, determines the number of records that the program reads when storing actual data in the array AERDAT. It is thus used as the terminal parameter in the DO 110 and DO 330 loops in the subroutine PLOT. If IOUTC is greater than the first dimension of AERDAT (which is 3000 at present), the program will prompt the user for an increment and/or bound change that will restrict the number of points to 3000. The value of IOUTC will therefore indicate whether the number of data points that the program can handle must be increased.

5.6.2 Changing the Number of Variables

Each record of the direct-access file contains one value for each of the variables output by the flight simulation program. If the number of the variables is changed, the size of the array SX, which stores their values during the reading of each record, and the variable list that is displayed on the screen by the subroutine PRINT, must be changed accordingly.

The integer array VARTYP contains number codes identifying the units of the variables. The codes for PLTSIMV1 are listed in Table 15. If the number of variables changes, the size of the VARTYP array, and its data initialization, must also be changed. The VARTYP codes refer to the two-dimensional array LBUNIT. The elements of LBUNIT that have the code number as their second subscript (namely LBUNIT(X,code), where X varies from 1 to 9) contain numbers that are ASCII designations for the characters printed on the graph as unit labels.

The magnitude of the first dimension of LBUNIT, which for PLTSIMV1 is 9, determines the maximum number of characters in the longest unit label. The numbers contained in LBUNIT are inputs to the NOTATE and HLABEL subroutines of the PLOT 10 package for the printing of units on the plots. The data statement that assigns these numbers is arranged so that the values representing one label occupy one line. If new variables involve new units, the appropriate additions will have to be made to LBUNIT.

The array VRNAME contains the names of all of the simulation variables as character data, and corresponds to the variables of Table 6. The maximum name length for PLTSIMV1 is 6 characters, which exceeds the limit of 4 characters per REAL*4 variable in Fortran. Thus each name is split into 2 unequal sections of 4 and 2 characters, reflected by the size of 2 for the first dimension of the array VRNAME. The second dimension equals the number of variables. Blank spaces fill up the elements of VRNAME where the name is shorter than the maximum.

The contents of VRNAME are used in the prompts for parameter input.

The elements of VRNAME are also used to print variable labels on the plotted graphs, although the PLOT 10 package requires that their contents be converted to ASCII code before input to the NOTATE and NLABEL subroutines. This conversion is done by the KAM2AS utility function, a PLOT 10 routine, immediately preceeding the calls to the PLOT 10 subroutines. This function requires real or integer variables as input arguments, preventing the creation of VRNAME as a character array.

5.6.3 Organization of Arrays

The program can plot up to 10 curves per graph, which, as shown in Table 16, allows for any combinations in terms of number of files and variables. The file numbers represent the files chosen for a particular plot in the order they were chosen (this is also the case for the variable numbers). The indicated curve number determines, for certain arrays, in which element information for those combinations of file and variable numbers is stored. Other arrays (as listed) hold information that is unique only to files (rather than curves), and thus the file number determines array element storage.

6. VMS COMMAND PROCEDURES

FLISIMV1, FLISIMV2 and PLTSIMV1 are part of a growing SIG flight dynamics library. This library supports software packages that span a broad range of activities associated with unmanned aerial system flight dynamics and design and include the following:

a) Preliminary configuration aerodynamic prediction software packages (subsonic and supersonic).

b) Control law synthesis software including frequency response and open/closed-loop natural modes.
c) Ballistic vehicle six DOF simulation software.
d) Flight vehicle six DOF simulation software.
e) Control surface hinge moment prediction software.
f) Flight test data reduction and analysis software including TM and radar data reduction.
g) Wind tunnel data reduction and analysis software.
h) Plotting packages associated with all of these software elements.

These software packages have evolved into a comprehensive capability that is currently being integrated as part of an interactive, user friendly "super package" under the control of a command language linking all of the elements.

As part of this process, command procedures have been developed for FLISIMV1, FLISIMV2 and PLTSIMV1 that interactively take care of the following main functions:

a) Compilation and linking of library and user modules.
b) Input and output unit number and file name assignment.
c) Execution of jobs either interactively or as batch jobs at the user's discretion.
d) Assignment of files to plotting software and access to plotting software at the user's discretion.
e) Specification of file protection attributes.
f) Some error recovery capability.

The command procedures for FLISIMV1 and FLISIMV2 are given in Appendix D and are called FLISIMV1.COM and FLISIMV2.COM. They are written under VAX 11/780 DCL (Reference 6). Each procedure allows access to PLTSIMV1, at the user's option, either after accessing the simulation software or directly if simulation data files are already available.

The command procedures are highly interactive and prompt the user for specification of options and file names. They are set-up for execution on the Military Engineering Section/Ordnance Detection Group VAX 11/780. As such they are configured for accessing the software based on specific disk and directory names. Use of these procedures on other systems supporting the VMS O/S will require changes in disk drive assignments and possibly in the directory names.

All SIG generated and/or supported software is located on the DISKSIG:[SIG1.FLIDYN] directory and its subdirectories. These subdirectories include the following:

a) DISKSIG:[SIG1.FLIDYN.FOR] - Fortran source software that is not part of specialized libraries.
b) DISKSIG:[SIG1.FLIDYN.OBJ] - Object modules.
c) DISKSIG:[SIG1.FLIDYN.EXE] - Executable modules.
d) DISKSIG:[SIG1.FLIDYN.LIB] - Directory containing SIG software object module libraries.

It is anticipated that the FLISIMV1 and FLISIMV2 command procedures will continue to evolve as they become fully integrated into the full complement of software packages available in the flight dynamics library. As such the current procedures are considered to be interim procedures and will not be documented in detail here.

REFERENCES - VOLUME 2

1. Markov, A.B., **The Landing Approach in Variable Winds: Curved Glidepath Geometries and Worst-Case Wind Modeling**, UTIAS Report 254, December 1981.

2. Anonymous, **PLOT 10 Terminal Control System, User Manual**, Tektronix Inc., 1977.

3. Anonymous, **PLOT 10 Advanced Graphing II, User Manual**, Tektronix Inc., 1977.

4. Anonymous, **VAX 11 FORTRAN Language Reference Manual**, Digital Equipment Corporation AA-DØ34COTE, April 1982.

5. Anonymous, **IMSL Library Reference Manual**, Vols. 1-3, Edition 8, IMSL, Inc., June 1980.

6. Anonymous, **VAX/VMS DCL Dictionary**, Digital Equipment Corporation AA-2200A-TE, September 1984.

7. Reid, L.D. and Graf, W.O., **ROBOT-X Preliminary Control Law Synthesis - Phase 2**, Final Report, DSS File No. 01SG.97702-R-3-6715, November 1983. UNCLASSIFIED.

8. Gear, C.W., **Numerical Initial Value Problems in Ordinary Differential Equations**, Prentice-Hall, 1971.

TABLE 1 FLISIMV1 PROGRAM MODULE DESCRIPTIONS

PROGRAM SEGMENT	MODULE NAME	ADDITIONAL ENTRY POINTS	TYPE	DESCRIPTION
FLISIMV1	AEROD	None	Subroutine	Used to define the aerodynamic forcing functions.
	FLISIMV1	None	Main Program	Controls program execution.
	XD	None	Subroutine	Defines the right hand side of the first order differential equations modeling the vehicle's six degree-of-freedom rigid body equations of motion.
FLISIMCO	CMCORR	None	Subroutine	Corrects aerodynamic data for changes caused by centre-of-mass shifts.
	INTRAN	None	Subroutine	Performs a rotational transformation on a mass moment of inertia matrix.
	MCHRØ	MCHR	Subroutine	Computes the vehicle's moments of inertia, centre-of-mass position and total mass.
	PRIN	None	Subroutine	Prints formatted output file data.
	PRINOP	None	Subroutine	Controls formatted output file data tables; prints table headers.
	ROCTHR	None	Subroutine	Computes the thrust generated by each of the NRE rocket motors; computes total thrust components in vehicle body-axes.
	ROCMOM	None	Subroutine	Computes the thrust moments generated by each of the NRE rocket motors and returns the results in body-axes components.
	SLLBBC	None	Subroutine	Defines the rotation matrix $\underline{L}_{BBC}$.

TABLE 1 FLISIMV1 PROGRAM MODULE DESCRIPTIONS (cont'd)

PROGRAM SEGMENT	MODULE NAME	ADDITIONAL ENTRY POINTS	TYPE	DESCRIPTION
FLISIMCO (cont'd)	SLLBI	None	Subroutine	Defines the rotation matrix $\underline{L}_{BI}$.
	TWINDR	None	Subroutine	Computes wind total time derivative.
	XDDDJ	None	Subroutine	Dummy subroutine required by IMSL subroutine DGEAR.
FLISIMUS	BEACON	None	Subroutine	Defines homing beacon characteristics.
	INPCHA	None	Subroutine	Allows user defined changes to master data from case to case in a given run.
	SERVO	None	Subroutine	Defines the control servo dynamics.
	WINDI	WIND	Subroutine	Defines the wind velocity components acting at the vehicle centre-of-mass expressed in F_I.
FLISIMAP	ACCEL	None	Subroutine	Defines the autopilot accelerometer model.
	ANRATE	None	Subroutine	Defines the autopilot angular rate sensor model.
	AUTCOM	None	Subroutine	Defines autopilot parameters associated with mission profile (includes all waypoints); models flight director characteristics.
	AUTPIL	None	Block data subroutine	Defines all constants associated with autopilot algorithms.
	ESTIM	None	Subroutine	Computes all parameters estimated by the autopilot, i.e. all parameters not directly available from sensor data.

TABLE 1 FLISIMV1 PROGRAM MODULE DESCRIPTIONS (cont'd)

PROGRAM SEGMENT	MODULE NAME	ADDITIONAL ENTRY POINTS	TYPE	DESCRIPTION
FLISIMAP (cont'd)	MAGNET	None	Subroutine	Defines the autopilot magnetometer model.
	PADATA	None	Subroutine	Defines the primary pitot-static pressure transducer characteristics.
	PILOT	None	Subroutine	Controlling subroutine for modules associated with the autopilot model FLISIMAP. Must be available to FLISIMV1 (or FLISIMV2) in at least a dummy form. All other modules in FLISIMAP are required by PILOT in association with the ROBOT-X autopilot model.
	RADALT	None	Subroutine	Defines the autopilot radar altimeter model.
	SENSOR	None	Subroutine	Defines the autopilot sensor data vector used in the control algorithms.
	VGYRO	None	Subroutine	Defines the autopilot vertical gyro model.
AEROATMOS	ATMONS	None	Subroutine	Defines nonstandard atmospheric characteristics based on user requirements.
	ATMOS	None	Subroutine	Defines characteristics of ICAO standard atmosphere.
	GLININ	None	Subroutine	Interpolation subroutine required by INLAAD and INLOAD.
	INLAAD	LILAAD	Subroutine	Defines lateral aerodynamic characteristics based on look-up table data.
	INLOAD	LILOAD	Subroutine	Defines longitudinal aerodynamic characteristics based on look-up table data.

TABLE 1 FLISIMV1 PROGRAM MODULE DESCRIPTIONS (cont'd)

PROGRAM SEGMENT	MODULE NAME	ADDITIONAL ENTRY POINTS	TYPE	DESCRIPTION
AEROATMOS (cont'd)	SETUP	None	Subroutine	Configures aerodynamic data arrays required by INLAAD and INLOAD.
MATHPACKP	ESLINS	None	Subroutine	Performs a linear interpolation of the data in the dependent variable given a value of the evenly spaced independent variable.
	INTPL	None	Subroutine	Performs a linear interpolation of the data in the dependent variable given a value of the independent variable. The independent variable need not be uniformly spaced. Double precision.
	MATOUT	None	Subroutine	A general utility routine for formatted output of a two dimensional matrix of arbitrary order n x m.
IMSL	DGEAR	None	Subroutine	An IMSL routine (Ref. 5) used to solve six DOF first order differential equations.
	DGRCS	None	Subroutine	Required by DGEAR.
	DGRIN	None	Subroutine	Required by DGEAR.
	DGRPS	None	Subroutine	Required by DGEAR.
	DGRST	None	Subroutine	Required by DGEAR.
	LINV2F	None	Subroutine	A IMSL matrix inversion routine (Ref. 5).
	UERSET	None	Subroutine	Required by DGEAR, LINV2F and VMULFF.

TABLE 1 FLISIMV1 PROGRAM MODULE DESCRIPTIONS (cont'd)

PROGRAM SEGMENT	MODULE NAME	ADDITIONAL ENTRY POINTS	TYPE	DESCRIPTION
IMSL (cont'd)	UERTST	None	Subroutine	Required by DGEAR, LINV2F and VMULFF.
	UGETIO	None	Subroutine	Required by DGEAR, LINV2F and VMULFF.
	VMULFF	None	Subroutine	IMSL matrix multiplication utility routine (Ref. 5).

TABLE 2 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE FORMAT

DATA ELEMENTS	FORMAT
NLABEL(I), I=1, 20	20A4
NCASE, IVARL	12I5
ICASE, NTOTAL	12I5
IOUTS(I), I=1, 20	12I5
ALTØ, ALTST	4G15.6
TTMAX, RDELT, TOL	4G15.6
ISPSO, ISPSO1, METH, MITER	12I5
XIIDØ, YIIDØ, HDØ, (XØ(I), I=4, NTOTAL)	4G15.6
IAPAR	12I5
ACPAR(I), I=1, IAPAR	4G15.6
((CLIM(I,J), J=1,2), I=1,5)	2G15.6
IPAR	12I5
(PAR(I), I=1, IPAR)	4G15.6
NPOINT	12I5
SCAT, SCATT, ATCCDD	4G15.6
TIME(1), THRUS(1)	4G15.6
• • •	• • •
TIME(NPOINT), THRUS(NPOINT)	4G15.6
NSTCNF, NRE, NC	12I5
(NMOCNF(I), I=1, NSTCNF)	12I5

TABLE 2 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE FORMAT (cont'd)

DATA ELEMENTS	FORMAT
(RDAT(1,J), J=7,13)	8F10.4
•	•
•	•
•	•
(RDAT(NRE,J), J=7,13)	8F10.4
RMLON(1), RBLON(1), RMLAT(1), RBLAT(1)	4G15.6
•	•
•	•
•	•
RMLON(15), RBLON(15), RMLAT(15), RBLAT(15)	4G15.6
NATMOS	12I5
ALTA(1), TMPRA(1), PRESA(1)	4G15.6
•	•
•	•
•	•
ALTA(NATMOS), TMPRA(NATMOS), PRESA(NATMOS)	4G15.6

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE PARAMETER DESCRIPTIONS

(In order of appearance in master data input file, see Table 2)

PARAMETER	DECLARATION DIMENSION(S)	TYPE	DESCRIPTION
NLABEL	20	INTEGER*4	User defined descriptor for particular run or for the vehicle configuration. Maximum length of descriptor 80 alphanumeric characters.
NCASE	--	INTEGER*4	Number of cases in a given run.
IVARL	--	INTEGER*4	Number of different parameters recorded and stored for each simulation time point (for the current version of the program, IVARL=182).
ICASE	--	INTEGER*4	A user defined case descriptor.
NTOTAL	--	INTEGER*4	The order of the simulation equations (NTOTAL=12).
IOUTS	20	INTEGER*4	A parameter controlling tabular data output: IOUTS(I)=1 means I-th table is outputted. IOUTS(I)=0 means the output of the I-th table is suppressed
ALTØ	--	REAL*8	Elevation above sea level (ASL) of the origin of the Earth-fixed reference frame F_I (m).
ALTST	--	REAL*8	Altitude ASL of the centre-of-mass of the vehicle at which the simulation for a given case is to stop.
TTMAX	--	REAL*8	Maximum time for which the simulation for a given case is to continue (sec).

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA
INPUT FILE PARAMETER DESCRIPTIONS (cont'd)

(In order of appearance in master data input file, see Table 2)

PARAMETER	DECLARATION DIMENSION(S)	TYPE	DESCRIPTION
RDELT	--	REAL*8	Time interval for each output step when rocket thrust is off (sec).
TOL	--	REAL*8	IMSL subroutine DGEAR tolerance parameter (typically TOL = 1.0 x 10^{-4}), see Ref. 5.
ISPSO	--	INTEGER*4	Number of autopilot updates per RDELT seconds. If the autopilot update time interval is RDTINT, then RDTINT = RDELT/ISPSO.
ISPSO1	--	INTEGER*4	Number of autopilot updates per output time point for thrust not equal to zero. Normally ISPSO1 ≤ ISPSO.
METH	--	INTEGER*4	IMSL subroutine DGEAR method parameter. METH=0 Adams-Moulton predictor-corrector. METH=1 Gear's backward differentiation method. See also Ref. 5.
MITER	--	INTEGER*4	IMSL subroutine DGEAR Jacobian computation method parameter. MITER=1 Computation is made based on a subroutine provided by the user (XDDDJ). This mode is not available as XDDDJ is dummy. MITER=2 MITER=3

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE PARAMETER DESCRIPTIONS (cont'd)

(In order of appearance in master data input file, see Table 2)

PARAMETER	DECLARATION DIMENSION(S)	TYPE	DESCRIPTION
XIIDØ	--	REAL*8	$\dot{x}_I(t_0)$ (m/s)
YIIDØ	--	REAL*8	$\dot{y}_I(t_0)$ (m/s)
HDØ	--	REAL*8	$\dot{h}(t_0)$ (m/s)
XØ	27	REAL*8	$x(t_0)$; XØ(4) = P(0) (deg/s) XØ(5) = Q(0) (deg/s) XØ(6) = R(0) (deg/s) XØ(7) = $\Phi(0)$ (deg) XØ(8) = $\Theta(0)$ (deg) XØ(9) = $\Psi(0)$ (deg) XØ(10) = $x_I(0)$ (m) XØ(11) = $y_I(0)$ (m) XØ(12) = $h(0)$ (m) altitude above launch site elevation
IAPAR	--	INTEGER*4	The number of parameters to be inputted in array ACPAR (IAPAR≤20); currently IAPAR=23.
ACPAR	30	REAL*8	An array containing a number of parameters required by the program. ACPAR(1) $x_{I_{BE}}$ (m) ACPAR(2) $y_{I_{BE}}$ (m) ACPAR(3) $h_{ASL_{BE}}$ (m) ACPAR(4) s_{Laun} (m) ACPAR(5) $s_{T\text{-}bolt}$ (m) ACPAR(6) Ψ_T (deg) ACPAR(7) x_{TI} (m) ACPAR(8) y_{TI} (m) ACPAR(9) z_{TI} (m)

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE PARAMETER DESCRIPTIONS (cont'd)

(In order of appearance in master data input file, see Table 2)

PARAMETER	DECLARATION DIMENSION(S)	TYPE	DESCRIPTION
ACPAR (cont'd)			ACPAR(10) $f_{C_{D_{ex}}}$: excresance drag factor, $C_{D_{ex}}=f_{C_{D_{ex}}}(C_{D_o}-C_{D_{base}})$
			ACPAR(11) $f_{C_{D_{base}}}$: base drag factor, $C_{D_{base}}=f_{C_{D_{base}}}\ 0.13\ S_{base}/S$
			ACPAR(12) S_{base} (m^2)
			ACPAR(13) $C_{D_{os}}$: drag coefficient offset, $C_D=C_D+C_{D_{os}}$
			ACPAR(14) $C_{L_{os}}$: lift coefficient offset, $C_L=C_L+C_{L_{os}}$
			ACPAR(15) $C_{m_{os}}$: pitching moment coefficient offset, $C_m=C_m+C_{m_{os}}$
			ACPAR(16) $C_{y_{os}}$: side force coefficient offset, $C_y=C_y+C_{y_{os}}$
			ACPAR(17) $C_{\ell_{os}}$: rolling moment coefficient offset, $C_\ell=C_\ell+C_{\ell_{os}}$

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE PARAMETER DESCRIPTIONS (cont'd)

(In order of appearance in master data input file, see Table 2)

PARAMETER	DECLARATION DIMENSION(S)	TYPE	DESCRIPTION
ACPAR (cont'd)			ACPAR(18) $C_{n_{os}}$: yawing moment coefficient offset, $C_n = C_n + C_{n_{os}}$
			ACPAR(19) $C_{D_{drogue}}$: drag coefficient of drogue parachute
			ACPAR(20) $C_{D_{main}}$: drag coefficient of main canopy
			ACPAR(21) S_{drogue}: (m²) reference area of drogue
			ACPAR(22) S_{main}: (m²) reference area of main canopy
			ACPAR(23) ℓ_B: (m) length of fuselage
CLIM	(5,2)	REAL*8	Array defining physical limits of control mechanism deflections and/or motions.
			CLIM(1,1) $\delta_{E_{min}}$ (deg)
			CLIM(1,2) $\delta_{E_{max}}$ (deg)
			CLIM(2,1) $\delta_{T_{min}}$ (throttle fraction)
			CLIM(2,2) $\delta_{T_{max}}$ (throttle fraction)
			CLIM(3,1) $\delta_{i_{H_{min}}}$ (deg)
			CLIM(3,2) $\delta_{i_{H_{max}}}$ (deg)
			CLIM(4,1) $\delta_{A_{min}}$ (deg)

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE PARAMETER DESCRIPTIONS (cont'd)

(In order of appearance in master data input file, see Table 2)

PARAMETER	DECLARATION DIMENSION(S)	TYPE	DESCRIPTION
CLIM (cont'd)			CLIM(4,2) $\delta_{A_{max}}$ (deg) CLIM(5,1) $\delta_{R_{min}}$ (deg) CLIM(5,2) $\delta_{R_{max}}$ (deg)
IPAR	--	INTEGER*8	Number of user defined parameters for wind modeling subroutines
PAR	200	REAL*8	Array containing user defined parameters used in a number of user defined subroutines (SERVO, WINDI, etc.), see Table 7.
NPOINT	--	INTEGER*4	Number of time points in individual rocket motor thrust mode (NPOINT≤201).
SCAT	--	REAL*8	Factor permitting scaled deviation of time points of thrust model from nominal, i.e. TIME(I) = SCAT*TIME(I).
SCATT	--	REAL*8	Factor permitting scaled deviation of rocket motor thrust model from nominal, i.e. THRUS(I) = SCATT*THRUS(I).
ATCCDD	--	REAL*8	$-\partial C_D/\partial(T_i/T_{i_{max}})$, i.e. drag coefficient reduction due to thrust of a single rocket motor.
TIME	201	REAL*8	Array containing time points for rocket motor thrust model; these points do not have to be uniformly spaced (sec).
THRUS	201	REAL*8	Array containing thrust of rocket motor at time points corresponding to TIME(I) (N).
NSTCNF	--	INTEGER*4	Number of stages wired into configurator board (NSTCNF≤10).
NRE	--	INTEGER*4	Total number of rocket motors (NRE≤25).
NC	--	INTEGER*4	Number of vehicle components that do not change their mass characteristics with time not including the rocket motor casing (NC≤20).

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE PARAMETER DESCRIPTIONS (cont'd)

(In order of appearance in master data input file, see Table 2)

PARAMETER	DECLARATION DIMENSION(S)	TYPE	DESCRIPTION
NMOCNF	10	INTEGER*4	Number of motors wired for firing in the i-th rocket motor stage.
IFIRE	(10,5)	INTEGER*4	IFIRE(I,J) designates which of the NRE motors are fired for the I-th stage (I=1, NSTCNF; J=1, NMOCNF(I)).
RMASS	20	REAL*8	RMASS(I) = mass of i-th airframe component (not including rocket motors), I=1, NC (kg).
RIIXX	20	REAL*8	RIIXX(I) = I_{xx} of i-th airframe component (not including rocket motors), I=1, NC, about a reference frame whose origin is at the centre-of-mass of the component (kgm^2).
RIIYY	20	REAL*8	I_{yy} of i-th airframe component, also see RIIXX (kgm^2).
RIIZZ	20	REAL*8	I_{zz} of i-th airframe component, also see RIIXX (kgm^2).
RIIXY	20	REAL*8	I_{xy} of i-th airframe component, also see RIIXX (kgm^2).
RIIXZ	20	REAL*8	I_{xz} of i-th airframe component, also see RIIXX (kgm^2).
RIIYZ	20	REAL*8	I_{yz} of i-th airframe component, also see RIIXX (kgm^2).
RX	20	REAL*8	Array containing x cylindrical coordinates of the centre-of-mass of the NC airframe components (m).
RR	20	REAL*8	Array containing r cylindrical coordinates of the centre-of-mass of the NC airframe components (m).
RPHI	20	REAL*8	Array containing ϕ cylindrical corrdinates of the centre-of-mass of the NC airframe components (deg).
RTHETA	20	REAL*8	
RDELTA	20	REAL*8	
PRMASS	--	REAL*8	Mass of propellant of each rocket motor (kg).

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE PARAMETER DESCRIPTIONS (cont'd)

(In order of appearance in master data input file, see Table 2)

PARAMETER	DECLARATION DIMENSION(S)	TYPE	DESCRIPTION
PRIIXX	--	REAL*8	I_{xx} of propellant of each rocket motor about a reference frame whose origin is at the centre-of-mass of the rocket motor (kgm^2).
PRIIYY	--	REAL*8	I_{yy} of propellant of each rocket motor, also see PRIIXX (kgm^2).
PRIIZZ	--	REAL*8	I_{zz} of propellant of each rocket motor, also see PRIIXX (kgm^2).
PRIIXY	--	REAL*8	I_{xy} of propellant of each rocket motor, also see PRIIXX (kgm^2).
PRIIXZ	--	REAL*8	I_{xz} of propellant of each rocket motor, also see PRIIXX (kgm^2).
PRIIYZ	--	REAL*8	I_{yz} of propellant of each rocket motor, also see PRIIXX (kgm^2).
REMASS	--	REAL*8	One-to-one correspondence with PRMASS to PRIIYZ but for empty rocket motors, i.e. rocket motors without propellant.
⋮	⋮	⋮	
REIIYZ	--	REAL*8	
RIISP	--	REAL*8	Specific impulse of one rocket motor (sec).
RLLNZ	--	REAL*8	$\partial L_R/\partial T_i$, rolling moment produced by rocket motor nozzle flutes about rocket motor x-axis per unit thrust.
RDAT	(25,13)	REAL*8	Array containing a number of rocket motor specific parameters. RDAT(I,J): Index I corresponds to i-th rocket motor, I≤NRE; index J corresponds to the 13 parameter for the i-th rocket motor, as follows: J=1,2,3 correspond to (x,r,ϕ) cylindrical coordinates of centre-of-mass of propellant of i-th rocket motor (m,m,deg).

TABLE 3 FLISIMV1 AND FLISIMV2 MASTER DATA INPUT FILE PARAMETER DESCRIPTIONS (cont'd)

(In order of appearance in master data input file, see Table 2)

PARAMETER	DECLARATION DIMENSION(S)	TYPE	DESCRIPTION
RDAT (cont'd)			J=4,5,6 correspond to (x,r,ϕ) cylindrical coordinates of centre-of-mass of empty (no propellant) i-th rocket motor (m,m,deg). J=7,8,9 correspond to (x,r,ϕ) cylindrical coordinates of thrust point of action of i-th rocket motor (m,m,deg). J=10,11 correspond to $(\theta_{TH_i}, \delta_{TH_i})$. Euler angles of thrust reference frame of i-th rocket motor (deg). J=12,13 correspond to $(\theta_{R_i}, \delta_{R_i})$. Euler angles of mass reference frames of i-th rocket motor (assumption has been made that empty and propellant (θ,δ) are indentical (deg).
RMLON RBLON	15 15	REAL*4 REAL*4	These arrays contain the parameters required to parametrically modify the nominal longitudinal aerodynamic parameters, i.e. LONAER(I)=RMLON(I)*LONAER(I)+RBLON(I).
RMLAT RBLAT	15 15	REAL*4 REAL*4	As above except for the nominal lateral aerodynamic parameters, i.e. LATAER(I)=RMLAT(I)*LATAER(I)+RBLAT(I).
NATMOS		INTEGER*4	Number of user supplied data lines provided defining percent deviations from nonstandard atmosphere.
ALTA	50	REAL*8	Array containing altitudes ASL at which deviations from nonstandard conditions are defined.
TMPRA	50	REAL*8	Deviation from standard temperature in percent at the i-th altitude ASL.
PRESA	50	REAL*8	Deviation from standard pressure in percent at the i-th altitude ASL. Deviations for TMPRA and PRESA are defined in the additive sense, e.g. $TEMP=TEMP_{Standard}+TEMPRA/100*TEMP_{Standard}$.

TABLE 4 FLISIMV1 AND FLISIMV2 FORMATTED OUTPUT FILE TABLE HEADERS

OUTPUT TABLE	CONTENTS
0	TIME(SEC) VV(M/S) THETBB(DEG) BETA(DEG) XII(M) YII(M) HAGL(M)
1	T(SEC) UBB(M/S) WBB(M/S) Q(DEG/S) THETBB(DEG) XII(M) HAGL(M) AXBB/G AZBB/G
2	T(SEC) ALPHAF(DEG) V(M/S) MACHN CCLL CCDD CCM CCYBB CCLBB CCNBB
3	T(SEC) DELTTT DLEE(DEG) RIHH(DEG) DAA(DEG) DRR(DEG) IDFLAP DELTTC DLEC(DEG) RIHC(DEG) DAAC(DEG) DRRC(DEG)
4	T(SEC) HAGL(M) WW1(M/S) WW2(M/S) WW3(M/S) WW1D(M/S/S) WW2D(M/S/S) WW3D(M/S/S)
5	T(SEC) MASS(KG) IXX(KG*M*M) IYY(KG*M*M) IZZ(KG*M*M) IXY(KG*M*M) IXZ(KG*M*M) IYZ(KG*M*M) XCM(M) YCM(M) ZCM(M)
6	T(SEC) TTXXRR(NT) TTYYRR(NT) TTZZRR(NT) LLBBRR(NT*M) MMBBRR(NT*M) NNBBRR(NT*M)
7	T(SEC) HAGL(M) TEMP(C) RHO(KG/(M*M*M)) RRHOR(M) SOFS(M/S) PRES(PA) QDYN(PA) G(M/S/S)
8	T(SEC) VBB(M/S) BETA(DEG) PBB(DEG/S) RBB(DEG/S) PHIBB(DEG) SIBB(DEG) YII(M) AYBB/G
9	T(SEC) GAEE(DEG) SIEE(DEG) RR(M) ZEEII(DEG) ZAAII(DEG) ZEETT(DEG) ZAATT(DEG)
10	T(SEC) PHIBST(D) THETBST(D) SIBBST(D) XIIST(M) YIIST(M) HPPST(M) UBST(M/S) VBST(M/S) WBST(M/S)
11	T(SEC) VVEST(M/S) MACHST QDST(N/M/M) ALFST(D) HPPLPST(M) HPPHP(M/S) GAST(D) SIBBMST(D) GAGA(D*S) GAAL(D*S) GAMA(S)
12	T(SEC) MD(KG/S) IXD(KMM/S) IYD(KMM/S) IZD(KMM/S) IXYD(KMM/S) IXZD(KMM/S) IYZD(KMM/S) XCMD(M/S) YCMD(M/S) ZCMD(M/S)
13	T(SEC) RBLPST(D/S) ETLPST(D) SIBMLPS(D) SIBLPST(D) MST(KG) AXST(M/S/S) AYST(M/S/S) AZST(M/S/S) VVELP(M/S) PHILT(D) GARB(D*S)
14	T(SEC) IAPLON IAPLAT RBBC(D/S) PHIBC(D) THETBC(D) SIBC(D) XIIC(M) YIIC(M) HAGLC(M) VVT(M/S)

TABLE 4 FLISIMV1 AND FLISIMV2 FORMATTED OUTPUT FILE TABLE HEADERS (cont'd)

OUTPUT TABLE	HEADER
15	T(SEC) VVEC(M/S) MACHC QDDC(N/M/M) ALFC(D) HPHPC(M/S) GAC(D) ETAC(D) NZC PBS(D/S) QBS(D/S) RBS(D/S)
16	T(SEC) NXSEN NYSEN NZSEN PAAS(PA) PDDS(PA) HHXS(MGA) HHYS(MGA) HHZS(MGA) HRAD(M) ETASEN(D)
17	T(SEC) GPHI(D*D) VED(M/S/S) TLPVG(D) PLPVG(D) TDVG(D/S) PDVG(D/S) THLTST(D) THEVG(D) PHIVG(D) GMPH(D*S) GMTH(D*S)
18	T(SEC) GMSI(D*S) GETA(D*S) PLP(D/S) QLP(D/S) PDST(D/S2) QDST(D/S2) RDST(D/S2)
19	T(SEC) NXCGST NYCGST NZCGST NXCGST NXCG NYCG NZCG GASTLT(D) GAST(D)
20	T(SEC) NXSLP NYSLP NZSLP PASLP(PA) PDSLP(PA) HDPLP(M/S) GGALT(D*S) GHPR(M*S) GHRA(M*S)

Remarks:

a) Suppression of output tables 1 to 20 is controlled through the master data file parameter IOUTS (see Sections 3.3.1 and 3.4.1).

b) Table 0 is always generated and may not be suppressed.

TABLE 5 FLISIMV1 AND FLISIMV2 FORMATTED OUTPUT FILE TABLE VARIABLE INDICES

OUTPUT TABLE	# OF VARIABLES	INDICES
1	9	1, 2, 4, 6, 9, 11, 13, 62, 64
2	10	1, 27, 25, 26, 50, 51, 52, 53, 54, 55
3	12	1, 30, 31, 32, 33, 34, 68, 69, 70, 71, 72, 73
4	8	1, 13, 38, 39, 40, 35, 36, 37
5	10	1, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23
6	7	1, 41, 42, 43, 44, 45, 46
7	9	1, 13, 56, 57, 58, 59, 60, 61, 67
8	9	1, 3, 24, 5, 7, 8, 10, 12, 63
9	8	1, 28, 29, 47, 49, 48, 65, 66
10	10	1, 90, 91, 92, 93, 94, 95, 84, 85, 86
11	12	1, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106
12	11	1, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83
13	12	1, 107, 108, 109, 110, 111, 112, 113, 114, 145, 146, 147
14	11	1, 143, 144, 115, 116, 117, 118, 119, 120, 121, 159
15	12	1, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132
16	11	1, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142
17	12	1, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158
18	8	1, 160, 161, 162, 163, 164, 165, 166
19	9	1, 167, 168, 169, 170, 171, 172, 173, 102
20	10	1, 174, 175, 176, 177, 178, 179, 180, 181, 182

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION (cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
21	x_{cm}	m	XCM
22	y_{cm}	m	YCM
23	z_{cm}	m	ZCM
24	β	deg	BETA
25	V	m/s	VV or V
26	M_a		MACHN
27	α_f	deg	ALPHAF
28	γ_E	deg	GAEE
29	ψ_E	deg	SIEE
30	δ_T		DELTTT
31	δ_E	deg	DLEE
32	i_H	deg	RIHH
33	δ_A	deg	DAA
34	δ_R	deg	DRR
35	$\dot{W}_1$	m/s^2	WW1D
36	$\dot{W}_2$	m/s^2	WW2D
37	$\dot{W}_3$	m/s^2	WW3D
38	W_1	m/s	WW1
39	W_2	m/s	WW2
40	W_3	m/s	WW3

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION (cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
41	T_x	N	TTXXRR
42	T_y	N	TTYYRR
43	T_z	N	TTZZRR
44	L_{BR}	Nm	LLBBRR
45	M_{BR}	Nm	MMBBRR
46	N_{BR}	Nm	NNBBRR
47	R	m	RR
48	ξ_{AI}	deg	ZAAII
49	ξ_{EI}	deg	ZEEII
50	C_{L_1}		CCLL
51	C_{D_1}		CCDD
52	C_{m_1}		CCM
53	C_y		CCYBB
54	C_ℓ		CCLBB
55	C_n		CCNBB
56	T_A	K	TEMP
57	ρ	kg/m^3	RHO
58	R_H	m	RRHOR
59	a	m/s	SOFS
60	P_A	Pa	PRES

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION

(cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
61	q_D	N/m^2	QDYN
62	a_x/g	g's	AXBB/G
63	a_y/g	g's	AYBB/G
64	a_z/g	g's	AZBB/G
65	ξ_{ET}	deg	ZEETT
66	ξ_{AT}	deg	ZAATT
67	g	m/s^2	G
68	IDFLAP		IDFLAP
69	δ_T^c		DELTTC
70	δ_E^c	deg	DLEC
71	i_H^c	deg	RIHC
72	δ_A^c	deg	DAAC
73	δ_R^c	deg	DRRC
74	$\dot{m}$	kg/s	MD
75	$\dot{I}_{xx}$	kgm^2/s	IXD

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION (cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
76	$\dot{I}_{yy}$	kgm²/s	IYD
77	$\dot{I}_{zz}$	kgm²/s	IZD
78	$\dot{I}_{xy}$	kgm²/s	IXYD
79	$\dot{I}_{xz}$	kgm²/s	IXZD
80	$\dot{I}_{yz}$	kgm²/s	IYZD
81	$\dot{x}_{cm}$	m/s	XCMD
82	$\dot{y}_{cm}$	m/s	YCMD
83	$\dot{z}_{cm}$	m/s	ZCMD
84	U*	m/s	UBBST
85	V*	m/s	VBBST
86	W*	m/s	WBBST
87	P*	deg/s	PBS
88	Q*	deg/s	QBS
89	R*	deg/s	RBS
90	Φ*	deg	PHIBST

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION

(cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
91	Θ^*	deg	THETBST
92	Ψ^*	deg	SIBBST
93	x_I^*	m	XIIST
94	y_I^*	m	YIIST
95	h_{AGL}^*	m	HPPST
96	V_e^*	m/s	VVEST
97	M_a^*		MACHST
98	q_D^*	P_a	QDST
99	α_f^*	deg	ALFST
100	$h_{p_{\ell_p}}^*$	m	HPPLPST
101	$\dot{h}_p^*$	m/s	HPPHP
102	γ^*	deg	GAST
103	ψ_{BM}	deg	SIBBMST
104	γ_{γ_ϵ}	deg•s	GAGA
105	$\gamma_{\alpha_{f_\epsilon}}$	deg•s	GAAL

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION

(cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
106	$\gamma_{n_{z_\epsilon}}$	s	GAMA
107	$R_{s_{\ell p}}$	deg/s	RBLPST
108	η_{ℓ_p}	deg	ETLPST
109	$\psi^*_{m_{\ell p}}$	deg	SIBMLPS
110	$\psi^*_{B_{\ell p}}$	deg	SIBLPST
111	m^*	kg	MST
112	a^*_x	m/s^2	AXST
113	a^*_y	m/s^2	AYST
114	a^*_z	m/s^2	AZST
115	R^c_B	deg/s	RBBC
116	Φ^c_B	deg	PHIBC
117	Θ^c_B	deg	THETBC
118	Ψ^c_B	deg	SIBC
119	x^c_I	m	XIIC
120	y^c_I	m	YIIC

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION (cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
121	h^c_{AGL}	m	HAGLC
122	V^c_e	m/s	VVEC
123	M^c_a		MACHC
124	q^c_D	P_a	QDDC
125	α^c_f	deg	ALFC
126	$\dot{h}^c_p$	m/s	HPPHPC
127	γ^c	deg	GAC
128	$\eta^c_{\ell p}$	deg	ETAC
129	n^c_z	g's	NZC
130	P_{B_s}	deg/s	PBS
131	Q_{B_s}	deg/s	QBS
132	R_{B_s}	deg/s	RBS
133	n_{x_s}	g's	NXSEN
134	n_{y_s}	g's	NYSEN
135	n_{z_s}	g's	NZSEN

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION (cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
136	P_{A_s}	Pa	PAAS
137	P_{D_s}	Pa	PDDS
138	H_{x_s}	mGauss	HHXS
139	H_{y_s}	mGauss	HHYS
140	H_{z_s}	mGauss	HHZS
141	h_{R_s}	m	HRAD
142	η_s	deg	ETASEN
143	IAPLON		IAPLON
144	IAPLAT		IAPLAT
145	$V^*_{e_{\ell p}}$	m/s	VVELP
146	$\Phi^*_{\ell t}$	deg	PHILT
147	$\gamma_{R_{B_\epsilon}}$	deg	GARB
148	$\gamma_{\Phi_{B_\epsilon}}$	deg•s	GPHI
149	$\dot{V}^*_e$	m/s	VED
150	$\Theta^*_{B_{\ell p}}$	deg	TLPVG

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION (cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
151	$\Phi^*_{B_{\ell p}}$	deg	PLPVG
152	$\dot{\Theta}^*_{VG}$	deg/s	TDVG
153	$\dot{\Phi}^*_{VG}$	deg/s	PDVG
154	$\Theta^*_{\ell t}$	deg	THLTST
155	Θ_{VG_s}	deg	THEVG
156	Φ_{VG_s}	deg	PHIVG
157	$\gamma_{\Phi_{cf}}$	deg•s	GMPH
158	$\gamma_{\theta_{cf}}$	deg•s	GMTH
159	V*	m/s	VVT
160	$\gamma_{\psi_{cf}}$	deg•s	GMSI

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION (cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
161	γ_{η_ϵ} or γ_{ψ_ϵ}	deg•s	GETA
162	$P^*_{B_{\ell p}}$	deg/s	PLP
163	$Q^*_{B_{\ell p}}$	deg/s	QLP
164	$\dot{P}^*_B$	deg/s²	PDST
165	$\dot{Q}^*_B$	deg/s²	QDST
166	$\dot{R}^*_B$	deg/s²	RDST
167	$n^*_{x_{cg}}$	g's	NXCGST
168	$n^*_{y_{cg}}$	g's	NYCGST
169	$n^*_{z_{cg}}$	g's	NZCGST
170	$n_{x_{cg}}$	g's	NXCG

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION

(cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
171	$n_{y_{cg}}$	g's	NYCG
172	$n_{z_{cg}}$	g's	NZCG
173	γ^*_{LT}	deg	GASTLT
174	$n_{x_{s_{\ell p}}}$	g's	NXSLP
175	$n_{y_{s_{\ell p}}}$	g's	NYSLP
176	$n_{z_{s_{\ell p}}}$	g's	NZSLP
177	$P_{A_{s_{\ell p}}}$	P_a	PASLP
178	$P_{D_{s_{\ell p}}}$	P_a	PDSLP
179	$\dot{h}^*_{p_{\ell p}}$	m/s	HDPLP
180	$\gamma_{\gamma_{LT_{\epsilon}}}$	deg•s	GGALT

TABLE 6 FLISIMV1 AND FLISIMV2 OUTPUT PLOTTING FILE DESCRIPTION (cont'd)

I	SX(I)	UNITS	ASSOCIATED OUTPUT TABLE NAME
181	$\gamma_{h_{P_\epsilon}}$	ms	GHPR
182	$\gamma_{h_{R_\epsilon}}$	ms	GHRA

NOTE: Units are given as used in output tables or as stored on the plotting file.

TABLE 7 FLISIMV1 AND FLISIMV2 ARRAY PAR ELEMENT DEFINITIONS

ELEMENT	VARIABLE	UNITS	FORTRAN EQUIVALENT NAME	UTILIZING SUBROUTINE	REMARKS
1	T_{δ_T}	sec	TTDTT	SERVO	Servo model time
2	T_{δ_E}	sec	TTDEE	SERVO	constants.
3	T_{i_H}	sec	TTIHH	SERVO	
4	T_{δ_A}	sec	TTDAA	SERVO	
5	T_{δ_R}	sec	TTDRR	SERVO	
6	Δx_I	m	DINCX	WINDI, TWINDR	Wind total derivative
7	Δy_I	m	DINCY	WINDI, TWINDR	finite difference
8	Δh	m	DINCH	WINDI, TWINDR	time increments.
9	Δt	sec	DINCT	WINDI, TWINDR	
10	--	---	ITWIND	WINDI	Wind model flag. ITWIND=0 wind and turbulence are zero. =1 shear profiles + turbulence. =2 constant mean wind + turbulence. =3 user defined wind profiles as a function of h_{ASL}.

TABLE 7 FLISIMV1 AND FLISIMV2 ARRAY PAR ELEMENT DEFINITIONS (cont'd)

ELEMENT	VARIABLE	UNITS	FORTRAN EQUIVALENT NAME	UTILIZING SUBROUTINE	REMARKS
11	σ_1	m/s	SIGMA1	WINDI	Turbulence files standard derivations.
12	σ_2	m/s	SIGMA2	WINDI	
13	σ_3	m/s	SIGMA3	WINDI	
14	W_{1L_0}	m/s	WW1LLØ	WINDI	Constant mean wind characteristics.
15	W_{2L_0}	m/s	WW2LLØ	WINDI	
16	W_{3L_0}	m/s	WW3LLØ	WINDI	
17	ψ_L	deg	AZLAD	WINDI	
18	W_G	m/s	BLWIND	WINDI	Exponential boundary layer wind characteristics.
19	h_G	m	HBL	WINDI	
20	n		EXPONT	WINDI	
21	h_{i_1}	m	HI1	WINDI	Wind speed jet characteristics.
22	h_{i_2}	m	HI2	WINDI	
23	Δs_{H_1}	m	SH1	WINDI	
24	Δs_{H_2}	m	SH2	WINDI	
25	W_{i_1}	m/s	WINDI1	WINDI	
26	W_{i_2}	m/s	WINDI2	WINDI	
27	$x_{E_{11}}$	m	XERI1(1)	WINDI	Downdraft cell characteristics.
28	$x_{E_{12}}$	m	XERI1(2)	WINDI	
29	$x_{E_{21}}$	m	XERI2(1)	WINDI	
30	$x_{E_{22}}$	m	XERI2(2)	WINDI	

TABLE 7 FLISIMV1 AND FLISIMV2 ARRAY PAR ELEMENT DEFINITIONS (cont'd)

ELEMENT	VARIABLE	UNITS	FORTRAN EQUIVALENT NAME	UTILIZING SUBROUTINE	REMARKS
31	$y_{E_{11}}$	m	YERI1(1)	WINDI	
32	$y_{E_{12}}$	m	YERI1(2)	WINDI	
33	$y_{E_{21}}$	m	YERI2(1)	WINDI	
34	$y_{E_{22}}$	m	YERI2(2)	WINDI	
35	$\Delta s_{V_{x_1}}$	m	SVX(1)	WINDI	
36	$\Delta s_{V_{x_2}}$	m	SVX(2)	WINDI	
37	$\Delta s_{V_{y_1}}$	m	SVY(1)	WINDI	
38	$\Delta s_{V_{y_2}}$	m	SVY(2)	WINDI	
39	$W_{V_{10}}$	m/s	VWIND(1)	WINDI	
40	$W_{V_{20}}$	m/s	VWIND(2)	WINDI	
41	N_V		NDRAFT	WINDI	Number of downdraft cells, $N_V \leq 2$.
42	ζ_o	deg	ZETNOT	WINDI	Direction shear characteristics.
43	κ_ζ	deg/m	KAPPA	WINDI	
44	$h_{\zeta_{i_1}}$	m	HZETI1	WINDI	
45	$h_{\zeta_{i_2}}$	m	HZETI2	WINDI	
46	Δs_{ζ_1}	m	SZETA1	WINDI	
47	Δs_{ζ_2}	m	SZETA2	WINDI	
48	ζ_{i_1}	deg	ZETAI1	WINDI	
49	ζ_{i_2}	deg	ZETAI2	WINDI	

TABLE 7 FLISIMV1 AND FLISIMV2 ARRAY PAR ELEMENT DEFINITIONS (cont'd)

ELEMENT	VARIABLE	UNITS	FORTRAN EQUIVALENT NAME	UTILIZING SUBROUTINE	REMARKS
50	--	--	--	--	Not used.
51	--	--	--	--	Not used.
52	N_{wind}		NWIN	WINDI	Number of user-defined wind data altitude levels.
53+(I-1)*4	h_{ASLi}	m	WALTA(I)	WINDI	Altitude ASL for i-th user defined wind velocity data altitude level, I=1, NWIN
54+(I-1)*4	ζ	deg	WDIRA(I)	WINDI	As above for wind direction measured relative to True North.
55+(I-1)*4	W_H	m/s	WSPEA(I)	WINDI	As above for horizontal wind speed.
56	W_V	m/s	WS3A(I)	WINDI	As above for vertical wind vector component, positive downwards.

TABLE 8 ESTIMATED STATE VECTOR XEST: ROBOT-X FLISIMAP MODULE

ELEMENT	SYMBOL	DESCRIPTION
1	U*	Estimated body-axes x airspeed vector component.
2	V*	Estimated body-axes y airspeed vector component.
3	W*	Estimated body-axes z airspeed vector component.
4	P*	Estimated body-axes roll rate.
5	Q*	Estimated body-axes pitch rate.
6	R*	Estimated body axes yaw rate.
7	Φ^*	Estimated Euler roll angle.
8	Θ^*	Estimated Euler pitch angle.
9	Ψ^*	Estimated Euler heading angle.
10	x_I^*	Estimated inertial x-position.
11	y_I^*	Estimated inertial y-position.
12	h_P^*	Estimated pressure altitude above launch site elevation.

TABLE 9 AUXILIARY STATE VECTOR XAUXES:
ROBOT-X FLISIMAP MODULE

ELEMENT	SYMBOL	DESCRIPTION
1	V_e^*	Estimated equivalent airspeed.
2	M_a^*	Estimated Mach number.
3	q_D^*	Estimated dynamic pressure.
4	α_f^*	Estimated fuselage reference line angle of attack.
5	$h^*_{p_{\ell p}}$	Estimated low pass filtered altitude above launch site elevation.
6	$\dot{h}_p^*$	Estimated pressure rate of climb.
7	γ^*	Estimated flight path angle.
8	ψ_m^*	Estimated magnetic heading corrected for magnetic variation, i.e. relative to True North.
9	γ_{γ_ϵ}	Integral of flight path angle deviation.
10	$\gamma_{\alpha_{f_\epsilon}}$	Integral of angle of attack deviation.
11	$\gamma_{n_{z_\epsilon}}$	Integral of maneuvering g (n_z) deviation.
12	$R_{s_{\ell p}}$	Low pass filtered yaw rate.
13	$\eta_{\ell p}$	Low pass filtered homing angle.
14	$\psi^*_{m_{\ell p}}$	Low pass filtered magnetic heading.
15	$\psi^*_{B_{\ell p}}$	Low pass filtered Euler heading angle.

TABLE 9 AUXILIARY STATE VECTOR XAUXES: ROBOT-X FLISIMAP MODULE (cont'd)

ELEMENT	SYMBOL	DESCRIPTION
16	m^*	Estimated ROBOT-X mass.
17	a_x^*	Estimated x-direction inertial acceleration expressed as body-axes components.
18	a_y^*	Estimated y-direction inertial acceleration expressed as body-axes components.
19	a_z^*	Estimated z-direction inertial acceleration expressed as body-axes components.
20	$\gamma_{R_{B_\epsilon}}$	Integral of commanded yaw rate deviation.
21	$V^*_{e_{\ell p}}$	Low pass filtered equivalent airspeed.
22	$\Phi^*_{\ell t}$	Long-term estimated bank angle.
23	$\gamma_{\Phi_{B_\epsilon}}$	Integral of commanded bank angle deviation.
24	$\dot{V}^*_e$	Estimated equivalent airspeed time derivative.
25	$\Theta^*_{B_{\ell p}}$	Low pass filtered pitch angle.
26	$\Phi^*_{B_{\ell p}}$	Low pass filtered roll angle.
27	$\dot{\Theta}^*_{VG}$	Pitch rate derived from vertical gyro.
28	$\dot{\Phi}^*_{VG}$	Roll rate derived from vertical gyro.
29	$\Theta^*_{\ell t}$	Long-term estimate of pitch angle.

TABLE 9 AUXILIARY STATE VECTOR XAUXES:
ROBOT-X FLISIMAP MODULE (cont'd)

ELEMENT	SYMBOL	DESCRIPTION
30	$\gamma_{\Phi_{cf}}$	Estimation difference integral for roll Euler angle (complementary filtering algorithm).
31	$\gamma_{\Theta_{cf}}$	Estimation difference integral for pitch Euler angle (complementary filtering algorithm).
32	V^*	True airspeed estimate.
33	$\gamma_{\psi_{cf}}$	Estimation difference integral for heading Euler angle (complementary filtering algorithm).
34	γ_{η_ϵ}, γ_{ψ_ϵ}	Integral of homing angle deviation, or of heading angle deviation, depending on the lateral control mode.
35	$P^*_{B_{\ell p}}$	Low pass filtered roll rate.
36	$Q^*_{B_{\ell p}}$	Low pass filtered pitch rate.
37	$\dot{P}^*_B$	Estimated roll rate time derivative.
38	$\dot{Q}^*_B$	Estimated pitch rate time derivative.
39	$\dot{R}^*_B$	Estimated yaw rate time derivative.
40	$n^*_{x_{cg}}$	Corrected x-accelerometer output.
41	$n^*_{y_{cg}}$	Corrected y-accelerometer output.
42	$n^*_{z_{cg}}$	Corrected z-accelerometer output.
43	γ^*_{LT}	Long term estimate of flight path angle.

TABLE 9 AUXILIARY STATE VECTOR XAUXES: ROBOT-X FLISIMAP MODULE (cont'd)

ELEMENT	SYMBOL	DESCRIPTION
44	$n_{x_{s_{\ell p}}}$	Low pass filtered x-accelerometer.
45	$n_{y_{s_{\ell p}}}$	Low pass filtered y-accelerometer.
46	$n_{z_{s_{\ell p}}}$	Low pass filtered z-accelerometer.
47	$P_{A_{s_{\ell p}}}$	Low pass filtered absolute pressure.
48	$P_{D_{s_{\ell p}}}$	Low pass filtered differential pressure.
49	$\dot{h}^*_{P_{\ell p}}$	Low pass filtered estimated pressure altitude rate of change.
50	$\gamma_{\gamma_{LT_\epsilon}}$	Integral of long-term flight path angle deviation.
51	$\gamma_{h_{P_\epsilon}}$	Integral of pressure altitude deviation.
52	$\gamma_{h_{R_\epsilon}}$	Integral of radar altitude deviation.

TABLE 10 SENSOR OUTPUT VECTOR XSEN:
ROBOT-X FLISIMAP MODULE

ELEMENT	SYMBOL	DESCRIPTION
1	P_s	Sensed roll rate.
2	Q_s	Sensed pitch rate.
3	R_s	Sensed yaw rate.
4	n_{x_s}	Sensed x-accelerometer output.
5	n_{y_s}	Sensed y-accelerometer output.
6	n_{z_s}	Sensed z-accelerometer output.
7	P_{A_s}	Primary absolute pressure transducer.
8	P_{D_s}	Primary differential pressure transducer.
9	H_{x_s}	Sensed x-axis magnetometer output.
10	H_{y_s}	Sensed y-axis magnetometer output.
11	H_{z_s}	Sensed z-axis magnetometer output.
12	h_{R_s}	Sensed radar altitude.
13	η_s	Sensed ADF homing angle.
14	Θ_{VG_s}	Sensed vertical gyro pitch attitude.
15	Φ_{VG_s}	Sensed vertical gyro roll attitude.

TABLE 11 FLISIMV2 PROGRAM MODULE DESCRIPTIONS*

PROGRAM SEGMENT	MODULE NAME	ADDITIONAL ENTRY POINTS	TYPE	DESCRIPTION
FLISIMV2	AEROD2	None	Subroutine	Used to define the aerodynamic forcing functions.
	FLISIMV2	None	Main Program	Controls program execution.
	XD	None	Subroutine	Defines the right hand side of the first order differential equations modeling the vehicle's six degree of freedom rigid body dynamics. It is emphasized that the version of XD utilized in FLISIMV2 is different from that in FLISIMV1.
WTDATA	WTDTCON	AEROWT	Subroutine	Provides linearly interpolated wind tunnel data.
MATHPACKP	LNINUE	None	Subroutine	Performs bilinear interpolation.
	INTPLS	None	Subroutine	Single precision version of INTPL, see Table 1.

* Only modules not documented in Table 1 are shown.

TABLE 12 PLTSIMV1 PROGRAM MODULE DESCRIPTIONS*

PROGRAM SEGMENT	MODULE NAME	ADDITIONAL ENTRY POINTS	TYPE	DESCRIPTION
PLTSIMV1	PLOT	None	Subroutine	Generates plots as per user specified options.
	PLTSIMV1	None	Main Program	Controls program execution.
PLOTPACK	ABSRANGE	None	Subroutine	Determines abscissa range.
	ASKLINE	None	Subroutine	Queries line type.
	ASKSYMBOL	None	Subroutine	Queries symbol type.
	CLARIFY	None	Subroutine	Increases character space for major tick mark labels.
	INTERVAL	None	Subroutine	Adjusts the number of major tick mark intervals.
	LIMITSORT	None	Subroutine	Determines which abscissa values are within the user specified range.
	LINENULL	None	Subroutine	Sets all line types to 0.
	LINETYPE	None	Subroutine	Presents line type options.
	NULLSYMB	None	Subroutine	Sets all symbol types to 0.
	PRINT	None	Subroutine	Prints plotting variable options.
	RESPONSE	None	Subroutine	Inappropriate response routine.
	SORT	None	Subroutine	Sorts data in ascending order of values.
	SYMBOL	None	Subroutine	Presents symbol types.
	TITLE	None	Subroutine	Queries plot title.

* Only PLOT 10 functions and subroutines called directly in PLTSIMV1 and PLOTPACK routines are listed.

TABLE 12 PLTSIMV1 PROGRAM MODULE DESCRIPTIONS* (cont'd)

PROGRAM SEGMENT	MODULE NAME	ADDITIONAL ENTRY POINTS	TYPE	DESCRIPTION
PLOT 10*	BINITT	None	Subroutine	See References 2 and 3 for a description of PLOT 10 routines.
	CHECK	None	Subroutine	
	COMGET	None	Function	
	CPLOT	None	Subroutine	
	DINITY	None	Subroutine	
	DLIMX	None	Subroutine	
	DLIMY	None	Subroutine	
	DSHREL	None	Subroutine	
	DSPLAY	None	Subroutine	
	HLABEL	None	Subroutine	
	IBASEX	None	Subroutine	
	IBASEY	None	Subroutine	
	INITT	None	Subroutine	
	KAM2AS	None	Subroutine	
	LINE	None	Subroutine	
	MOVABS	None	Subroutine	
	MOVREL	None	Subroutine	
	NEWPAG	None	Subroutine	
	NOTATE	None	Subroutine	

* Only PLOT 10 functions and subroutines called directly in PLTSIMV1 and PLOTPACK routines are listed.

TABLE 12 PLTSIMV1 PROGRAM MODULE DESCRIPTIONS* (cont'd)

PROGRAM SEGMENT	MODULE NAME	ADDITIONAL ENTRY POINTS	TYPE	DESCRIPTION
PLOT 10* (cont'd)	NPTS	None	Subroutine	See References 2 and 3 for a description of PLOT 10 routines.
	SLIMX	None	Subroutine	
	SLIMY	None	Subroutine	
	STEPL	None	Subroutine	
	STEPS	None	Subroutine	
	SYMBL	None	Subroutine	
	SYMOUT	None	Subroutine	
	XFRM	None	Subroutine	
	XLAB	None	Subroutine	
	XTICS	None	Subroutine	
	XWDTH	None	Subroutine	
	YFRM	None	Subroutine	
	YLOC	None	Subroutine	
	YLOCRT	None	Subroutine	
	YTICS	None	Subroutine	
	YWDTH	None	Subroutine	

* Only PLOT 10 functions and subroutines called directly in PLTSIMV1 and PLOTPACK routines are listed.

TABLE 15 PLTSIMV1 ARRAY VARTYP UNIT CODES

Variable Unit Code	Units
1	(M)
2	(M/S)
3	(M/S/S)
4	(M*M/S)
5	(N)
6	(N*M)
7	(KG)
8	(KG/M**3)
9	(KG*M*M)
10	(C)
11	(KCAL)
12	(PA)
13	(DEG)
14	(DEG/S)
15	(D*S)
16	(SEC)
17	(M*S)
18	NIL
19	(MGAUS)
20	(KG/S)
21	(K*M*M/S)
22	(DEG/S/S)
23	(G*S)
24	(G)

TABLE 16 PLTSIMV1 ARRAY ELEMENT STORAGE

F - File number, V - Variable Number

No. of Files (NFILES)	1		2		3		4-5		6-10	
Max. No. of Variables (MXVARS)	10		5		4		2		1	
Curve #	F	V	F	V	F	V	F	V	F	V
1	1	1	1	1	1	1	1	1	1	1
2	1	2	1	2	1	2	1	2	2	1
3	1	3	1	3	1	3	2	1	3	1
4	1	4	1	4	2	1	2	2	4	1
5	1	5	1	5	2	2	3	1	5	1
6	1	6	2	1	2	3	3	2	6	1
7	1	7	2	2	3	1	4	1	7	1
8	1	8	2	3	3	2	4	2	8	1
9	1	9	2	4	3	3	5	1	9	1
10	1	10	2	5	-	-	5	2	10	1

A. Arrays Organized by Curve Number
AERDAT
IAXIS
NORDNT

B. Arrays Organized by File Number
ABSMIN, ABSMAX
KASE
IOUTC
ILUN
INC
LRANGE
NUMBER

UNCLASSIFIED
This Sheet Security Classification

DOCUMENT CONTROL DATA — R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1 ORIGINATING ACTIVITY
Defence Research Establishment Suffield
Box 4000, Medicine Hat, AB T1A 8K6

2a. DOCUMENT SECURITY CLASSIFICATION
unclassified

2b. GROUP

3 DOCUMENT TITLE
A NONLINEAR SIX DEGREE-OF-FREEDOM FLIGHT SIMULATION MODEL
VOLUME 2 SOFTWARE DOCUMENTATION (U)

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)
Suffield Report No. 446

5. AUTHOR(S) (Last name, first name, middle initial)

Markov, A.B.

6. DOCUMENT DATE May 1990

7a. TOTAL NO. OF PAGES

7b. NO. OF REFS
Volume 2 - 8

8a. PROJECT OR GRANT NO.
031SE

9a. ORIGINATOR'S DOCUMENT NUMBER(S)
SR 446

8b CONTRACT NO.

9b. OTHER DOCUMENT NO.(S) (Any other numbers that may be assigned this document)

10. DISTRIBUTION STATEMENT
Volumes 1 & 2 Unlimited Distribution
Volume 3 Limited Distribution

11. SUPPLEMENTARY NOTES

12. SPONSORING ACTIVITY
Defence Research Establishment Suffield
Box 4000, Medicine Hat, AB T1A 8K6

13. ABSTRACT

Six degree-of-freedom, rigid body equations of motion are described suitable for modeling the dynamic characteristics of multistaged rocket-boosted maneuvering aerial targets such as the DRES developed ROBOT-X. These equations of motion form the core of a Fortran simulation package called FLISIM. FLISIM is currently installed on a VAX 11/780 computer and allows for modeling of vehicle thrust and structural asymmetries, time-varying mass and inertia characteristics, autopilot control laws, autopilot update rates, autopilot sensor non-idealities, nonlinear aerodynamic characteristics, variable wind conditions, turbulence, nonstandard atmospheric conditions, stage and individual motor failures, different rocket motor types, and parachute deceleration dynamics. The FLISIM software package has been developed in two versions (FLISIMV1 and FLISIMV2) using two different aerodynamic models. Both are written in VAX 11 Fortran and run under the VMS Operating System. FLISIM is fully supported with a plotting software package (PLTSIM) developed around Tektronix PLOT 10 core software.

Volume 1 describes the development of the equations of motion. Volume 2 is the FLISIM software userbook. Volume 3 contains FLISIM source code listings.

DSIS
70-070

UNCLASSIFIED
This Sheet Security Classification

KEY WORDS

aerial targets
aerodynamics
control dynamics
CRV-7
drones
equations of motion
flight dynamics
flight mechanics
ROBOT-X
rocket-boosted target
simulation
six degree-of-freedom
unmanned aerial vehicles

90-03580
#65829

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the organization issuing the document.

2a. DOCUMENT SECURITY CLASSIFICATION: Enter the overall security classification of the document including special warning terms whenever applicable.

2b. GROUP: Enter security reclassification group number. The three groups are defined in Appendix 'M' of the DRB Security Regulations.

3. DOCUMENT TITLE: Enter the complete document title in all capital letters. Titles in all cases should be unclassified. If a sufficiently descriptive title cannot be selected without classification, show title classification with the usual one-capital-letter abbreviation in parentheses immediately following the title.

4. DESCRIPTIVE NOTES: Enter the category of document, e.g. technical report, technical note or technical letter. If appropriate, enter the type of document, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the document. Enter last name, first name, middle initial. If military, show rank. The name of the principal author is an absolute minimum requirement.

6. DOCUMENT DATE: Enter the date (month, year) of Establishment approval for publication of the document.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the document.

8a. PROJECT OR GRANT NUMBER: If appropriate, enter the applicable research and development project or grant number under which the document was written.

8b. CONTRACT NUMBER: If appropriate, enter the applicable number under which the document was written.

9a. ORIGINATOR'S DOCUMENT NUMBER(S): Enter the official document number by which the document will be identified and controlled by the originating activity. This number must be unique to this document.

9b. OTHER DOCUMENT NUMBER(S): If the document has been assigned any other document numbers (either by the originator or by the sponsor), also enter this number(s).

10. DISTRIBUTION STATEMENT: Enter any limitations on further dissemination of the document, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this document from their defence documentation center."

(2) "Announcement and dissemination of this document is not authorized without prior approval from originating activity."

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document, even though it may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall end with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (TS), (S), (C), (R), or (U).

The length of the abstract should be limited to 20 single-spaced standard typewritten lines; 7½ inches long.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a document and could be helpful in cataloging the document. Key words should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context.