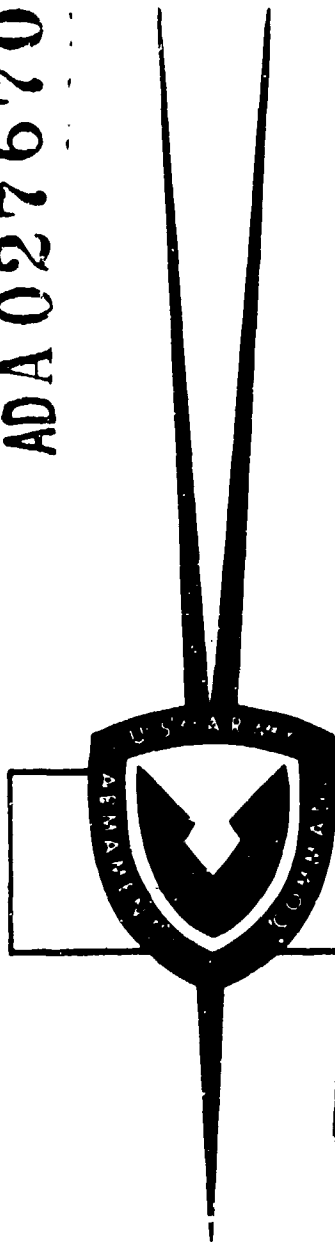


ADA 027670



[Handwritten signature]

AD

[Empty rectangular box]

FG

R-TR-76-019

**HYPERVELOCITY INFLIGHT TRAJECTORY
SCATTER (HIYS) CN CODE**

USER'S MANUAL

**ROBERT G. BELLAIRE
THEODORE O. GUSTAFSON**

APRIL 1976

FINAL REPORT

PREPARED BY

**AVCO CORPORATION
201 LOWELL STREET
WILMINGTON, MA 01887**

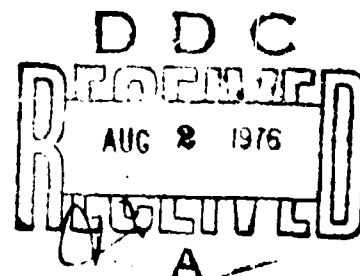
**CONTRACT NO.
DAAA09-74-C-
2026-P00004**

DISTRIBUTION STATEMENT A

**Approved for public release
Distribution Unlimited**

PREPARED FOR

**RESEARCH DIRECTORATE
GENERAL THOMAS J. RODMAN LABORATORY
ROCK ISLAND ARSENAL
ROCK ISLAND, ILLINOIS 61201**



HYPERVELOCITY INFLIGHT TRAJECTORY SCATTER (HITS) CODE

USER'S MANUAL

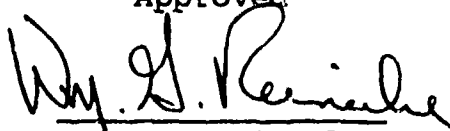
Prepared by

AVCO SYSTEMS DIVISION
201 Lowell Street
Wilmington, Massachusetts 01887

AVSD-0137-76-CR

Contract DAAA09-74-C-2026-P00004

Approved



Wm. G. Reinecke
Program Manager

Prepared for

GENERAL THOMAS J. RODMAN LABORATORY
Research Directorate
Rock Island Arsenal
Rock Island, Illinois 61201

April 1976

The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER R-ER-76-019	2. SOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER 9
4. TITLE (and Subtitle) Hypervelocity Inflight Trajectory Scatter (HITS) Code User's Manual.		5. DATE OF REPORT & PERIOD COVERED Final Report. Dec 1975 - Apr 1976
7. AUTHOR 10 Robert G. Bellaire Theodore O. Gustafson		6. PERFORMING ORG. REPORT NUMBER 14 AVSD-0137-76-CR 15 DAAA09-74-C-2026-P00004
8. PERFORMING ORGANIZATION NAME AND ADDRESS Avco Systems Division 201 Lowell Street Wilmington, Massachusetts 01887		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62603-1W662603AH78- 04-04
11. CONTROLLING OFFICE NAME AND ADDRESS Research Directorate, General Thomas J. Rodman Laboratory, Rock Island Arsenal, Rock Island, Illinois 61201		12. SECURITY DATE 11 Apr 1976
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 12 368 p.		13. NUMBER OF PAGES 378
16. DISTRIBUTION STATEMENT (of this Report) Unlimited 16 DA-1-W-662603-AH-78		15. SECURITY CLASS. (of this report) None
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) 17 1-W-662603-AH-7804		
18. SUPPLEMENTARY NOTES 18 RIA-R 19 TR-76-019		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) dispersion computer code user's manual increased velocity projectile exterior ballistics		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report is a User's Manual for the HITS computer code. The HITS code statistically evaluates projectile dispersion, which is the dispersion associated with the inflight behavior of a gun launched projectile. Projectile dispersion is a fundamental limit on overall weapon system effectiveness. The code is an automated computer-based analytic procedure for computing crossrange and downrange dispersion error budgets and sensitivity → next page		

DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

45

20. Abstract (Concl'd)

coefficients to the sources of projectile dispersion. The error budgets statistically define projectile dispersion at a level conducive to interpretation and comprehensive understanding.

HITS evaluates projectile dispersion by either analytic or Monte Carlo methods employing trajectory equations. The trajectory equations are closed form approximations to the six degrees of freedom equations of motion. The closed form equations presume the projectile does not experience transonic flow conditions during any phase of flight. This limits the present HITS code to this type of projectile (referred to here as "hypervelocity", with no other connotation intended). The trajectory equations have been subjected to extensive testing. In all cases they were accurate to within engineering tolerances. These equations have potential application to operational fire control computers, the computation of firing tables, and any other situation requiring rapid, accurate, low-cost trajectory determination.

Since this report is a User's Manual, the emphasis is placed on providing the information necessary to operate the code and interpret the results. Detailed input/output information is summarized in tables featuring step-by-step cook-book instructions. Example problems are discussed. Appendices present the theoretical and programming fine points, as well as the complete program listing.

FOREWORD

This User's Manual was prepared by Avco Systems Division, 201 Lowell Street, Wilmington, Massachusetts 01887, for the U. S. Army Armament Command, Gen. T. J. Rodman Laboratory, Rock Island Arsenal, Rock Island Illinois 61201, with Mr. William P. Wohlford of the Research Directorate as contract monitor.

The principal authors appreciate the contributions and suggestions offered by their colleagues: P. Crimi, R. Flaherty, P. Fote, R. Kingsley, W. Reinecke, and R. Torrani.

The authors wish to thank the many Rock Island personnel, and particularly Mr. William P. Wohlford and Mr. Charles S. Hicks, who provided Avco with much timely and relevant data and many sound suggestions.

The authors also wish to thank Mrs. L. Kiley, Mrs. J. Di Pietro, Mrs. P. Ford and Miss M. Smith for their careful and conscientious typing, Mrs. H. Cassell for the many equations, Miss A. Gaudet and Mr. R. Gaumond for their excellent art work, and Mr. E. Schurmann and Mr. L. Sacks for their editorial assistance.

ACCESSION NO.	
NTIS	White Section ☒
BDC	Buff Section ☐
UNANNOUNCED	☐
JUSTIFICATION	
BY	
DISTRIBUTION/AVAILABILITY CODES	
Dist.	AVAIL. AND/OR SPECIAL
A	

ABSTRACT

This report is a User's Manual for the HITS computer code. The HITS code statistically evaluates "projectile dispersion," which is the dispersion associated with the inflight behavior of a gun launched projectile. Projectile dispersion is a fundamental limit on overall weapon system effectiveness. The code is an automated computer-based analytic procedure for computing crossrange and downrange dispersion error budgets and sensitivity coefficients to the sources of projectile dispersion. The error budgets statistically define projectile dispersion at a level conducive to interpretation and comprehensive understanding.

HITS evaluates projectile dispersion by either analytic or Monte Carlo methods employing trajectory equations. The trajectory equations are closed form approximations to the six degrees of freedom equations of motion. The closed form equations presume the projectile does not experience transonic flow conditions during any phase of flight. This limits the present HITS code to this type of projectile (referred to here as "hypervelocity", with no other connotation intended). The trajectory equations have been subjected to extensive testing. In all cases they were accurate to within engineering tolerances. These equations have potential application to operational fire control computers, the computation of firing tables, and any other situation requiring rapid, accurate, low-cost trajectory determination.

Since this report is a User's Manual, the emphasis is placed on providing the information necessary to operate the code and interpret the results. Detailed input/output information is summarized in tables featuring step-by-step cook-book instructions. Example problems are discussed. Appendices present the theoretical and programming fine points, as well as the complete program listing.

TABLE OF CONTENTS

	<u>Page</u>
1.0 ENGINEERING PERSPECTIVE.....	1
1.1 Projectile Dispersion Sources.....	1
1.2 Weapon System Effectiveness Implications.....	3
1.3 Dispersion Analysis Methodology.....	5
1.4 Code Limitations.....	8
1.5 Report Overview.....	11
2.0 OVERVIEW.....	13
2.1 Code Structure.....	13
2.1.1 Input Processor.....	15
2.1.2 Statistical Processor.....	15
2.1.3 Projectile Trajectory Module.....	15
2.2 Analysis Options.....	16
2.2.1 Single Trajectory.....	16
2.2.2 Range Check.....	16
2.2.3 Analytical Statistical.....	17
2.2.4 Monte Carlo.....	17
2.3 Trajectory Equations.....	17
2.3.1 Closed Form Rationale.....	17
2.3.2 Summary Description.....	18
2.4 Related Topics.....	21
2.4.1 Crossrange Dispersion Indices.....	21
2.4.2 Correlation Analysis - An Example.....	22
2.4.3 Target-Fixed Crossrange Dispersion.....	25
3.0 INPUT DESCRIPTION.....	29
3.1 Statistical Processor Control Card (Card Group 1)	29
3.2 Input Processor Control Cards (Card Group 2)....	34

TABLE OF CONTENTS (Cont'd)

	<u>Page</u>
3.2.1 Card Group 2 Assigned Qualities.....	34
3.2.2 Trajectory Variable Code Numbers.....	39
3.3 Optional Input Processor Control Cards (Card Group 3).....	39
3.4 Error Source Model Encoding.....	54
4.0 OUTPUT INTERPRETATION.....	61
4.1 Single Trajectory Mode.....	61
4.2 Range Check Mode.....	71
4.3 Analytical Statistical Mode.....	84
4.4 Monte Carlo Mode.....	106
5.0 SUMMARY.....	139
APPENDIX A - ELEMENTS OF PROBABILITY THEORY.....	141
A.1 Random Phenomenon and Probabilities.....	141
A.1.1 Fundamental Concepts.....	141
A.1.2 Rules of Combination.....	142
A.1.3 Conditional Probabilities.....	144
A.2 Random Variables.....	145
A.2.1 Distribution Functions.....	145
A.2.2 Joint Distribution Functions.....	147
A.2.3 Statistical Independence.....	148
A.3 Averages.....	148
A.3.1 Expectation.....	149
A.3.2 Moments.....	150
A.3.3 Correlation.....	151
A.3.4 Confidence Limits.....	153
A.4 Typical Distributions.....	154
A.4.1 Uniformly Distributed Random Variable (TYPE = 3).....	154

TABLE OF CONTENTS (Cont'd)

	<u>Page</u>
A.4.2 Gaussian Distributed Random Variable (TYPE = 2).....	156
A.4.3 Jointly Gaussian Random Variables.....	159
A.4.4 Rayleigh Distributed Random Variable...	165
A.5 Random Variable Generation.....	166
A.5.1 Uniformly Distributed Variables (TYPE = 3).....	166
A.5.2 Gaussian Distributed Variables (TYPE = 2).....	167
A.5.3 Arbitrarily Distributed Variable (TYPE = 4).....	168
A.6 Monte Carlo Inaccuracies.....	170
A.6.1 Grouping Errors.....	170
A.6.2 Sampling Errors.....	173
APPENDIX B - HITS COMPUTER CODE DESCRIPTION.....	183
B.1 Statistical Processor Computational Procedures	183
B.1.1 Analytical Statistical Calculations....	183
B.1.2 Monte Carlo Calculations.....	190
B.2 Code Architecture.....	194
B.2.1 Data Storage.....	194
B.2.2 Input Processor Functions.....	195
B.2.3 Statistical Processor Functions.....	198
B.2.4 HITS Subroutines.....	203
B.2.5 Histogram Computer Plots.....	203
APPENDIX C - CLOSED FORM TRAJECTORY EQUATIONS.....	213
C.1 Introduction.....	213
C.2 Particle Trajectory.....	214

TABLE OF CONTENTS (Cont'd)

	<u>Page</u>
C.2.1 Analytical Development.....	214
C.2.2 Verification.....	224
C.3 Crossrange Perturbations.....	224
C.3.1 Analytical Development.....	224
C.3.2 Verification.....	238
C.4 Downrange Perturbations.....	238
C.4.1 Analytical Development.....	238
C.4.2 Verification.....	246
APPENDIX D - HITS PROGRAM LISTING.....	249
D.1 MAIN Program.....	250
D.2 Subroutine ARTLU.....	254
D.3 Function A2.....	255
D.4 Function A2F.....	256
D.5 Block Data.....	257
D.6 Subroutine CHCKLA.....	258
D.7 Subroutine CHCKIN.....	259
D.8 Subroutine CNVRT.....	261
D.9 Subroutine CROSS.....	262
D.10 Subroutine CROSSF.....	265
D.11 Subroutine CRVFT.....	268
D.12 Subroutine DOABC.....	269
D.13 Subroutine DO234.....	271
D.14 Subroutine EXTRA.....	282
D.15 Subroutine FC2987.....	283
D.16 Function FEXP.....	285
D.17 Subroutine FILLA.....	286
D.18 Subroutine FILLIN.....	287
D.19 Subroutine GQCOR.....	288
D.20 Subroutine GQUC.....	289
D.21 Subroutine G1795.....	290
D.22 Subroutine G2440.....	291
D.23 Subroutine G2987.....	292
D.24 Subroutine HSTG1.....	295
D.25 Subroutine HSTG2.....	296
D.26 Subroutine HYSPLT.....	297

TABLE OF CONTENTS (Concl'd)

	<u>Page</u>
D.27 Subroutine INCARD.....	298
D.28 Subroutine INCON	301
D.29 Subroutine INCONF.....	305
D.30 Subroutine INITIL.....	309
D.31 Subroutine INLHST.....	311
D.32 Subroutine LOADER.....	312
D.33 Subroutine MCRL.....	315
D.34 Subroutine MOVEUP.....	317
D.35 Function NCOV.....	318
D.36 Function NMEAN.....	319
D.37 Function NVAR.....	320
D.38 Function NVARX.....	321
D.39 Function N1D.....	322
D.40 Function N2D.....	323
D.41 Subroutine PICK1.....	324
D.42 Subroutine PRNTMC.....	325
D.43 Subroutine QCOR.....	327
D.44 Function RANDOM.....	328
D.45 Subroutine RANDU.....	329
D.46 Subroutine RW2987.....	330
D.47 Subroutine SAMPLE.....	332
D.48 Subroutine SPRNT.....	334
D.49 Subroutine STDDEV.....	336
D.50 Subroutine STOREC.....	337
D.51 Subroutine TG1795.....	338
D.52 Subroutine TG2440.....	339
D.53 Subroutine TG2987.....	340
D.54 Subroutine TRAJT.....	343
D.55 Subroutine TRAJX.....	346
D.56 Subroutine TRAJXF.....	348
D.57 Subroutine TVX.....	350
D.58 Subroutine T4NTV.....	351
D.59 Subroutine VXT.....	352
D.60 Subroutine WIND.....	353
D.61 Subroutine XXC.....	354
D.62 Function ZATAN2.....	355
D.63 Function ZZ.....	356
DISTRIBUTION.....	357

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1-1	Weapon System Effectiveness.....	3
1-2	HITS Overview.....	7
2-1	HITS Flow Diagram	14
2-2	Trajectory Equations Computational Procedure.....	19
2-3	Gaussian Scale Factors.....	23
2-4	Intercept Geometry.....	26
2-5	Crossrange Sensitivity to Downrange Dispersion.....	26
3-1	IBM-029 Keypunch Control Card for Card Group 1.....	33
3-2	IBM-029 Keypunch Control Cards for Card Group 2.....	40
3-3	IBM-029 Keypunch Control Cards for Card Group 3.....	56
3-4	Encoded Error Source Model.....	59
4-1	Single Trajectory Mode Check Problem Input.....	62
4-2	Single Trajectory Mode Check Problem Output.....	63
4-3	Range Check Mode Check Problem Input.....	72
4-4	Range Check Mode Check Problem Output.....	73
4-5	Analytical Statistical Mode Check Problem Input.....	86
4-6	Analytical Statistical Mode Check Problem Output.....	87
4-7	Monte Carlo Mode Check Problem Input.....	108
4-8	Monte Carlo Mode Check Problem Output.....	109

LIST OF FIGURES (CONT'D)

<u>Figure</u>		<u>Page</u>
A-1	The Venn Diagram.....	143
A-2	Chebyshev's Lower Bound	155
A-3	Gaussian Density Function.....	157
A-4	A Comparison of Two Gaussian Density Functions With Different Standard Deviations.....	158
A-5	Two Probability Density Functions With Identical Means and Variances.....	160
A-6	Gaussian Scale Factors.....	163
A-7	Von Newmann Rejection Sampling Illustration.....	169
A-8	Expected Error in Sample Mean in Percentage of Standard Deviation.....	180
A-9	Expected Error in Sample Variance Value.....	181
B-1	HITS Flow Diagram.....	184
B-2	Coordinates for Derivative Evaluations.....	191
B-3	Internal Data Transfers.....	196
B-4	HITS Subroutine Linking Diagram.....	208
B-5	Projectile Trajectory Module Subroutine Linking Diagram.....	209
B-6	Histogram Title Array.....	211
B-7	Histogram Storage Array ZDATAD (I,J,*).....	212
C-1	Trajectory Coordinate System.....	215
C-2	Typical Ballistic Coefficient Variation With Mach Number.....	223

LIST OF FIGURES (CONCL'D)

<u>Figure</u>		<u>Page</u>
C-3	Trajectory Comparison.....	225
C-4	Projectile Body Fixed Coordinate System.....	228
C-5	Lift Force Components in the Trajectory Frame.....	234
C-6	Lift Induced Dispersion as a Function of Roll Rate...	240
C-7	Downrange Forces Due to Angle of Attack.....	242

LIST OF TABLES

<u>Table</u>		<u>Page</u>
1-1	Sources of Projectile Dispersion	2
1-2	Error Source Model	6
1-3	Crossrange Dispersion at Nominal Time (Nominal Range = 10 Kft)	9
2-1	Probabilities of Occurrence	23
3-1	Statistical Processor Control Card (Card Group 1).....	30
3-2	Input Processor Control Cards (Card Group 2).....	35
3-3	HITS Trajectory Parameter List.....	41
3-4	Optional Input Processor Control Cards (Card Group 3)..	55
3-5	Error Source Model Code Numbers.....	57
4-1	Monte Carlo Mode Output Interpretation.....	134
A-1	Probabilities of Occurrence.....	164
B-1	Time and Storage Requirements.....	199
B-2	HITS Subroutine List.....	204
C-1	List of Symbols.....	216
C-2	Test Case Projectile Properties.....	226
C-3	Trajectory Model Verification.....	239

1.0 ENGINEERING PERSPECTIVE

The Hypervelocity¹ Inflight Trajectory Scatter (HITS) computer code is an automated procedure for evaluating "projectile dispersion", which is the dispersion associated with the in-flight behavior of a gun launched projectile. This report documents the code in the format of a User's Manual. Primary emphasis is placed on the information necessary to operate the code and interpret the results. Theoretical aspects and computational details are treated in the appendices.

This chapter addresses the following. Section 1.1 identifies the sources of projectile dispersion and, thereby, more clearly defines the term. Section 1.2 discusses the relationship of projectile dispersion to overall weapon system effectiveness. The dispersion analysis methodology suggested by the code is illustrated by example in Section 1.3. Section 1.4 summarizes the limitations of the code. An overview of the report is presented in Section 1.5. The overall objective of this chapter is to enable the engineer to quickly determine the utility of HITS with respect to his particular problem.

1.1 Projectile Dispersion Sources

The HITS computer code quantifies projectile dispersion. Projectile dispersion is defined here to be the dispersion associated with inflight behavior which cannot be compensated by fire control. (HITS does not compute the dispersion attributable to fire control. However, it does include in the projectile dispersion calculation the effects of fire control corrections for inflight phenomena.) Thus, the sources of projectile dispersion enumerated in this section are those factors which influence inflight behavior and cause the fire control predicted trajectory to differ from the true flight path.

Table 1-1 lists the sources of projectile dispersion. The left hand column contains the physical mechanisms through which the contributing factors of the right hand column act. There are four general categories: (1) initial conditions at the muzzle, (2) projectile mass properties, (3) projectile aerodynamic characteristics, and (4) atmospheric effects. The sources of Table 1-1 cause dispersion only insofar as the fire control computer cannot anticipate their effects and apply corrective action. For instance, winds are a source of projectile dispersion if the weapon system does not measure them. If it

¹ "Hypervelocity" is used throughout this report to indicate the limitation of the present HITS code to projectiles whose velocity does not become transonic at any point along the trajectory. No other connotation is intended.

Table 1-1 Sources of Projectile Dispersion

ERROR SOURCE	TYPICAL CONTRIBUTING FACTORS
Initial Conditions: Velocity Vector Angle of Attack Angular Rates	Sabot Separation In-Barrel Dynamics Blast Effects Barrel Vibrations Tipoff Rates Gun Slewing Rates
Inertial Characteristics: Weight Moments of Inertia Physical Dimensions C. G. Asymmetries	Manufacturing Tolerances
Aerodynamic Characteristics: Static Coefficients Dynamic Coefficients Spin Rate Trim Angle of Attack Static Margin	Manufacturing Tolerances Ablations Effect on the Ballistic Coefficient Coefficient Measurement Errors
Atmospheric Effects: Winds Density	Temporal and Spatial Atmospheric Variations

does, only the wind measurement error causes projectile dispersion. Although not mentioned in Table 1-1, simplifying assumptions incorporated in the fire control trajectory to minimize computational requirements are also considered sources of projectile dispersion. The HITS code accounts for each of these factors.

1.2 Weapon System Effectiveness Implications

This section discusses the role projectile dispersion plays in determining weapon system effectiveness with implications to projectile design procedures. The discussion opens with a brief overview of the factors which affect weapon system effectiveness.

Figure 1-1 is a block diagram of a modern gun weapon system. It indicates the various factors that influence weapon system effectiveness, as defined by the probability of kill. Referring to Figure 1-1, the engagement scenario shown on the left hand side establishes the geometry of the encounter. Target motion dynamics limit target motion to physically possible rates and accelerations. The acquisition sensor continually measures the target coordinates and passes the information to the fire control computer. Fire control's basic jobs are to solve the intercept geometry, align the gun, and signal the gunner at the appropriate time. Its calculations take into account not only the target coordinates but also information concerning the inflight behavior of the projectile, winds, atmospheric density, gun orientation, and various system models describing target dynamic constraints, gun mount rate and orientation limits, etc. Fire control issues commands to the gun mount servo system to point the gun. The servo holds the gun on the target while awaiting the command to fire. After the round has been fired, the projectile trajectory relative to the true target position determines the point of closest approach, that is the miss distance. If a hit is scored, the location of the hit and the projectile terminal ballistics determine the probability of kill.

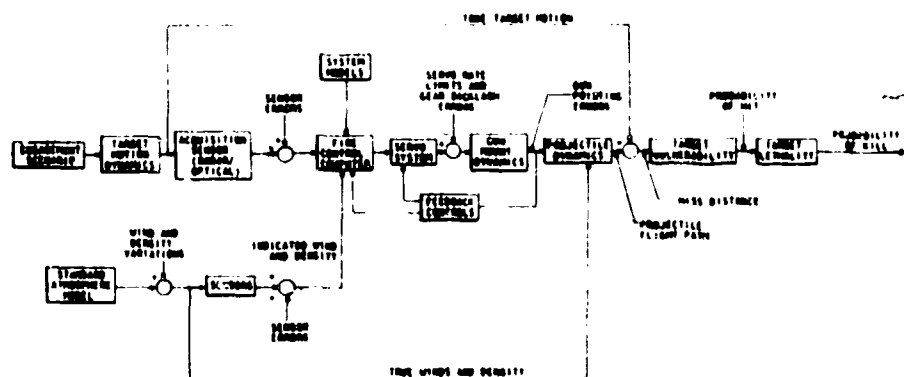


Figure 1-1 Weapon System Effectiveness

Projectile dispersion affects overall weapon system effectiveness. The accuracy with which the fire control computer can predict the true flight path of the projectile poses a fundamental limitation on weapon system effectiveness. Even if the exact position of the target relative to the gun were known and the target could be tracked with infinite precision, the inability of fire control to predict the flight path exactly would cause the projectile to miss the target. Thus, projectile dispersion is a fundamental limit on weapon system effectiveness because it defines a level of accuracy which cannot be improved upon by refinements of other elements of the weapon system. The HITS code quantitatively defines this limit.

HITS may also be used as a projectile design tool. Since all projectiles exhibit dispersion to a greater or lesser extent and this can limit overall weapon system effectiveness, it would be prudent to include the evaluation of projectile dispersion in all projectile design studies. The HITS code is an analytic tool for performing this evaluation inexpensively. Furthermore, HITS can compute a single trajectory under a given set of conditions or under a cyclic permutation of parameters. The calculation is quick, accurate, and inexpensive. Thus, HITS brings to design activities, additional capabilities that go beyond the assessment of projectile dispersion.

Although projectile dispersion poses a fundamental limit on weapon system effectiveness, it is not always clear what impact projectile dispersion will have on overall weapon system effectiveness. Returning to Figure 1-1, it is clear that overall weapon system effectiveness ultimately depends on how well the gun system operates as a whole. There is a synergistic effect which makes the system better than the sum of its parts, since one subsystem can be designed to compensate for the errors of another. For instance, fire control could be designed to monitor and correct for gun pointing errors, as indicated by the feedback loop shown in Figure 1-1. Thus, fire control can point the gun better than an analysis of the gun-mount servo system would indicate was possible. For this reason, it is potentially misleading to forecast overall weapon system effectiveness from an analysis of any single subsystem (e.g., a projectile dispersion analysis). Thus, judgment must be used in interpreting projectile dispersion assessments produced by the HITS code in terms of overall weapon system effectiveness.

1.3 Dispersion Analysis Methodology

The HITS computer code suggests a methodology for determining the susceptibility of projectile designs to projectile dispersion. This section demonstrates the two step method with an illustrative example. The following paragraphs illustrate the information required, the role of the HITS code, and the format of the results.

Table 1-1 lists the sources of projectile dispersion. By analyzing the contributing factors, statistics may be assigned to each source which quantify the uncertainty as illustrated in Table 1-2: the error source model. Displayed in the left hand column of Table 1-2 are the sources of projectile dispersion. The third column contains the nominal values for the projectile properties. The second column defines the distribution of variations about the nominal. The uncertainties are given in the fourth column: the standard deviations or one-sigma ($1-\sigma$) values.¹ The standard deviations quantify the level of uncertainty in the information supplied to fire control. The error source model may be even more detailed. The basic error source model of Table 1-2 could be expanded to include correlations between the elements of the error source model, such as projectile weight and muzzle velocity. Whatever the level of detail, construction of the error source model concludes the first step in evaluating projectile dispersion via the HITS code.

The second and last step is to use the HITS code to evaluate the projectile dispersion corresponding to the error source model as suggested by the schematic block diagram of Figure 1-2. The code reads the error source model as data and proceeds to quantify projectile dispersion in the form of an error budget. The code contains two sets of closed form trajectory equations: one represents the true trajectory and the other represents fire control. A Statistical Processor manipulates the trajectory equations in either an analytical or Monte Carlo fashion to determine the dispersion statistics. The results of the calculations are most conveniently summarized in the format of a projectile dispersion error budget as indicated in Figure 1-2. Any simplifying assumptions incorporated in an actual fire control trajectory are input to the code and taken into account.

The HITS code can construct error budgets in both cross-range and downrange directions as well as evaluate three dimensional dispersion indices such as the Spherical Error Probable (SEP). Error budgets may be constructed at either nominal time or nominal range. In this respect HITS is very flexible and can compute most every statistic customarily used to describe dispersion.

1 Rayleigh/Uniform uncertainties are specified by the mean magnitude given in the third column.

Table 1-2 Error Source Model

ERROR SOURCE	STATISTICAL DISTRIBUTION	MEAN VALUE	STANDARD DEVIATION
● INITIAL CONDITIONS ● VELOCITY VECTOR - MAGNITUDE - ORIENTATION ● INERTIAL ORIENTATION - ATTITUDE - ATTITUDE RATE	- Gaussian - Rayleigh/Uniform* - Rayleigh/Uniform* - Rayleigh/Uniform*	11,000 ft/sec 0.0005 deg 0.1 deg 55 rad/sec	1/3% -- -- --
● PHYSICAL CHARACTERISTICS ● WEIGHT ● MOMENTS OF INERTIA - AXIAL - PITCH ● PHYSICAL DIMENSIONS - REFERENCE AREA - LENGTH - BASE DIAMETER	- Gaussian - Gaussian - Gaussian - Uniform - Uniform - Uniform	0.11 lbs 1.234×10^{-6} slug-ft² 2.110×10^{-5} slug-ft² 3.068×10^{-3} ft² 0.31 ft 0.0625 ft	1.0% 1 2/3% 1 2/3% 2/3% 1/3% 1/3%
● AERODYNAMIC CHARACTERISTICS ● STATIC COEFFICIENTS - DRAG VARIATION EFFECTS - VELOCITY - ANGLE OF ATTACK - NORMAL FORCE ● DYNAMIC COEFFICIENTS - PITCH DAMPING - MAGNUS MOMENT ● SPIN RATE ● TRIM ANGLE OF ATTACK ● STATIC MARGIN	- Gaussian - Gaussian - Gaussian - Gaussian - Gaussian - Gaussian - Rayleigh/Uniform* - Gaussian	** 1.0 1/deg² 1.9767 1/rad -7.5 (none) 0.0 (none) 400 rad/sec 0.1 deg 6.2% length	1.0% 1.0% 2.0% 20.0% 0.0 1.0% -- 1.0% of length
● ATMOSPHERIC EFFECTS ● CONSTANT WINDS ● DENSITY VARIATIONS	- Rayleigh/Uniform* - Gaussian	11.0 ft/sec 2.378×10^{-3} slug-ft³	-- 1.0%

*Denotes a Rayleigh distribution of magnitude with a Uniform 360° distribution in orientation.

**Drag variation with velocity closely approximated by $C_{X_0} = C_{X_\infty} + \frac{K_D}{V^2}$ which is fit to two data points $(C_{X_1}, V_1) = (0.03585, 16,740)$ and $(C_{X_2}, V_2) = (0.11951, 3906)$. Uncertainty in C_{X_1} and C_{X_2} .

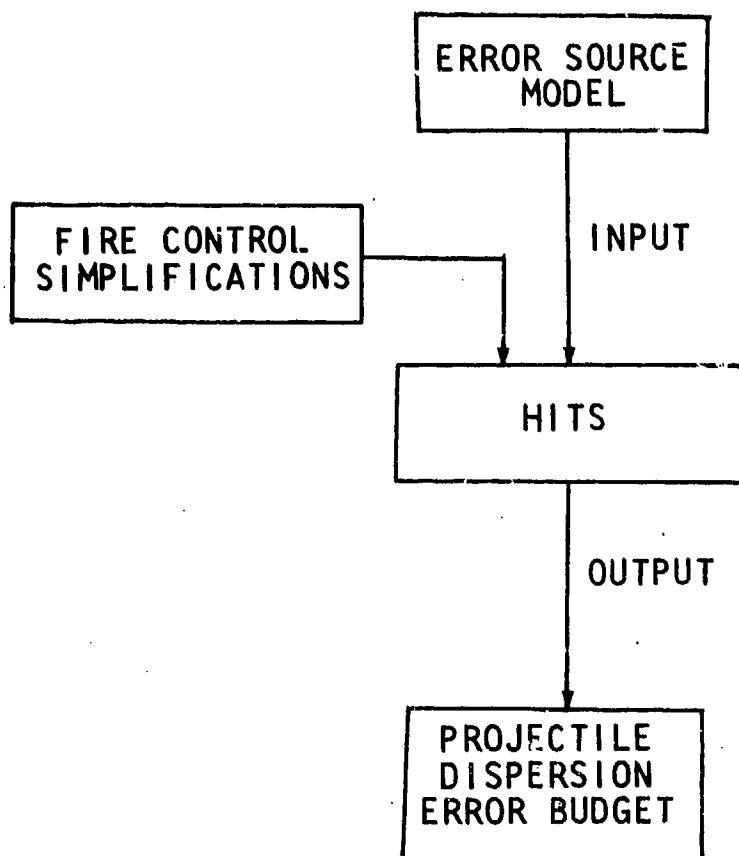


Figure 1-2 HITS Overview

Table 1-3 presents the error budget generated by HITS for crossrange dispersion at nominal time corresponding to the error source model of Table 1-2. The error budget presents both the dispersion and the sensitivity coefficient associated with each error source. This identifies the larger contributors and those with significant potential. The total dispersion is given in the lower right hand corner. Table 1-3 illustrates the level of detail and types of information available from HITS. This is precisely the level of detail the engineer needs to assess the situation and make decisions. Supplemental information concerning the dispersion probability distribution can be developed if desired.

In summary, the methodology suggested by the HITS code consists of (1) constructing a comprehensive error source model and (2) processing it to obtain a detailed error budget. The error budget contains the information necessary to evaluate a projectile with respect to projectile dispersion program objectives and/or trade-off rival candidate designs. The method could also be employed in conjunction with ballistic range tests devoted to dispersion assessment with the objective of deriving additional insight. Whatever the motive, HITS provides a scientific, systematic method of evaluating projectile dispersion.

Presentation of the error source model and error budget, Tables 1-2 and 1-3, requires comment concerning their generality. The error source model contains implicit assumptions concerning (1) manufacturing tolerances on the projectile and barrel, (2) the amount of testing employed to determine the aerodynamic characteristics, and (3) the presumed accuracy of wind and density measurements. The error budget reflects these assumptions and further assumes the fire control trajectory algorithm contains no simplifying approximations. Thus, the error source model and error budget presented here are only for the purposes of illustration. They are not necessarily representative of any weapon system in the inventory, under procurement, or in development.

1.4 Code Limitations

This section states the limitations of the HITS code. Most of the items listed below are not theoretical limitations. Some are assumptions made for modeling convenience, while others are merely computer requirements. In any case, they are listed

Table 1-3 Crossrange Dispersion at Nominal Time
(Nominal Range = 10 Kft)

ERROR SOURCE	UNCERTAINTY	SENSITIVITY (MRAD/.)	1-σ ERROR (MRAD)
● INITIAL CONDITIONS			
● VELOCITY VECTOR			
- MAGNITUDE	1/3%	0.001913	0.0006377
- ORIENTATION	0.0005 deg	13.91	0.006957
● INITIAL ORIENTATION			
- ATTITUDE	0.1 deg	0.01551	0.001551
- ATTITUDE RATE	55.0 rad/sec	0.02346	1.290
● PHYSICAL CHARACTERISTICS			
● WEIGHT	1.0%	0.009808	0.009808
● MOMENTS OF INERTIA			
- AXIAL	1 2/3%	-0	-0
- PITCH	1 2/3%	-0	-0
● PHYSICAL DIMENSIONS			
- REFERENCE AREA	2/3%	0.003439	0.002292
- LENGTH	1/3%	0.01274	0.004248
- BASE DIAMETER	1/3%	0.00002225	0.00007418
● AERODYNAMIC CHARACTERISTICS			
● STATIC COEFFICIENTS			
- DRAG VARIATION EFFECTS			
VELOCITY	1.0%	-0	-0
ANGLE OF ATTACK	1.0%	0.000007418	0.000007418
- NORMAL FORCE	2.0%	0.0008093	0.001619
● DYNAMIC COEFFICIENTS			
- PITCH DAMPING	20.0%	0.00001131	0.0002261
- MAGNUS MOMENT	0.0%	--	--
● SPIN RATE	1.0%	0.0007241	0.0007241
● TRIM ANGLE OF ATTACK	0.1 deg	0.8103	0.08103
● STATIC MARGIN	1% L	0.2183	0.2183
● ATMOSPHERIC EFFECTS			
● CONSTANT WINDS	11 ft/sec	0.01538	0.1692
● DENSITY VARIATIONS	1.0%	0.003429	0.003429
RSS TOTAL DISPERSION			1.322

below to aid the engineer in determining whether HITS is applicable to a particular problem and, if so, facilitate installation. The limitations are:

- The projectile drag model assumes the projectile does not experience transonic flow conditions at any time from launch to impact. A transonic condition would cause appreciable error only if it persisted over a significant portion of the flight time. This assumption can be overcome by additional code development. However, the code as presently configured is limited to "hypervelocity" projectiles.
- Muzzle blast, sabot separation and in-barrel dynamic effects are combined and represented by equivalent velocity and attitude perturbations at the muzzle. Ballistic range data reduction methods typically combine these effects in this manner.
- Wind gusts and density variations, that is fluctuations along the trajectory, are assumed to be zero. Covariance analysis using available wind gust spectra suggest wind gusts are a negligible source of dispersion for typical projectiles.¹ On the other hand, steady winds are known to be a significant source and are accounted for by the HITS code.
- When operated in the Monte Carlo mode, HITS cannot assess the effects of error source model correlations. A program modification would be required. Correlation effects may, however, be evaluated via the Analytical Statistical mode.
- The HITS code is written in FORTRAN IV and requires approximately 175K bytes of computer memory. The card deck is punched in the EBCDIC format.

¹ Gustafson, T. G., Crimi, P., Bellaire, R. G., "Dispersion of Increased Velocity Projectiles - Feasibility Phase," Avco Corporation, AVSD-0200-75-CR, July 1975.

- The random number generator (Subroutine RANDU) will work properly only on a 32 bit word computer, such as an IBM-360. The subroutine can be easily modified. However, some care must be exercised in performing the modification to insure the generated random sequences will reproduce those of the original code on a 32 bit word computer. Otherwise, the full instructional benefits of the example problems presented here will not be achieved, since it will not be possible to reproduce the Monte Carlo simulations.
- The code contains an interface for automatic computer plots. Since plotting software is hardware specific, this capability is not fully developed. The interface is thoroughly documented.
- The code was designed to facilitate real-time interactive mode operation from remote keyboard terminals. This capability is latent in the code, but not an operational reality.

1.5 Report Overview

Since this report is a User's Manual for the HITS code, the emphasis is on providing the information necessary to operate the code and interpret the results. The text presents the day-to-day necessities required to perform dispersion analyses. The appendices discuss the theoretical and programming aspects.

Chapter 2 presents an overview of the HITS code directed to the user. The objective is to acquaint the analyst quickly with the code. The basic code structure and options are described. The trajectory equations, which are the heart of the code, are discussed from a functional point of view. Chapter 2 concludes with a collection of topics peripheral to HITS, but pertinent to dispersion analysis.

Detailed input information is supplied in Chapter 3. A quick-reference step-by-step format is employed. Tables are used to define all variables and present all relevant information; comments direct the user to sections of this report containing related indepth discussions. Chapter 3 closes with an illustrative example demonstrating the encoding of the error source model of Table 1-2.

Chapter 4 presents four input-output example problems. Since the computer printed output is self-explanatory, the initial impression might be that these examples are superfluous. They are not. They illustrate the output format and provide numerical check problems. The accompanying text stresses interpretation. Thus, users are encouraged to reproduce these results to familiarize themselves with HITS and to verify the code at their facility.

A brief summary is presented in Chapter 5. The antecedents of the HITS code and related analyses are discussed.

The appendices relate the theoretical and programming aspects of the HITS code. Appendix A is a summary discussion of the relevant aspects of probability theory and statistics. Appendix B discusses programming fine points. Appendix C mathematically develops the trajectory equations. Appendix D contains the complete program listing.

2.0 OVERVIEW

The purpose of this chapter is to discuss the salient aspects of the HITS code. The objective is to present the user with a clear picture of the essence and function of the code as a whole. To this end, analytical and programming fine points are glossed over in an effort to place maximum emphasis on the big picture. Detailed "how to do it" information is presented in later chapters. Theoretical analyses and programming information are presented in the appendices.

Since projectile dispersion is most meaningfully stated in a statistical sense, the HITS code quantifies it in statistical terms. Thus, familiarity with the basic concepts of probability and statistics is essential to (1) the preparation of the engineering formulation of the HITS input, (2) a thorough understanding of the algorithms employed by HITS, and (3) the interpretation of the HITS output. Appendix A reviews the most pertinent elements of probability and statistics with the objective of supplying this understanding. HITS can be operated by personnel unfamiliar with probability and statistics. Given the information presented in Chapter 3, they can encode the formulated problem and obtain the desired numerical results.

Section 2.1 discusses the structure of the code. Analysis options are enumerated in Section 2.2. The close-form trajectory equations, which are the heart of the HITS code, are described in Section 2.3. Section 2.4 discusses three topics which are peripheral to HITS, but pertinent to dispersion analysis.

2.1 Code Structure

A flow diagram illustrating the basic structure of the HITS code is shown in Figure 2-1. As suggested by the figure, HITS basic function is to process an error source model to obtain a projectile dispersion error budget. This section describes the three computational modules that perform this task: (1) the Input Processor, (2) the Statistical Processor, and (3) the Projectile Trajectory Module. The code contains facilities for accessing a Time Phased Data Base for updating projectile characteristics under direction from a remote interactive computer terminal; however, this capability is not an operational reality at the present time. Sections 2.1.1 and 2.1.2 discuss the function of the Input and Statistical Processors which were taken from previously developed software. Section 2.1.3 functionally describes the Projectile Trajectory Module, which was developed specifically for dispersion assessment.

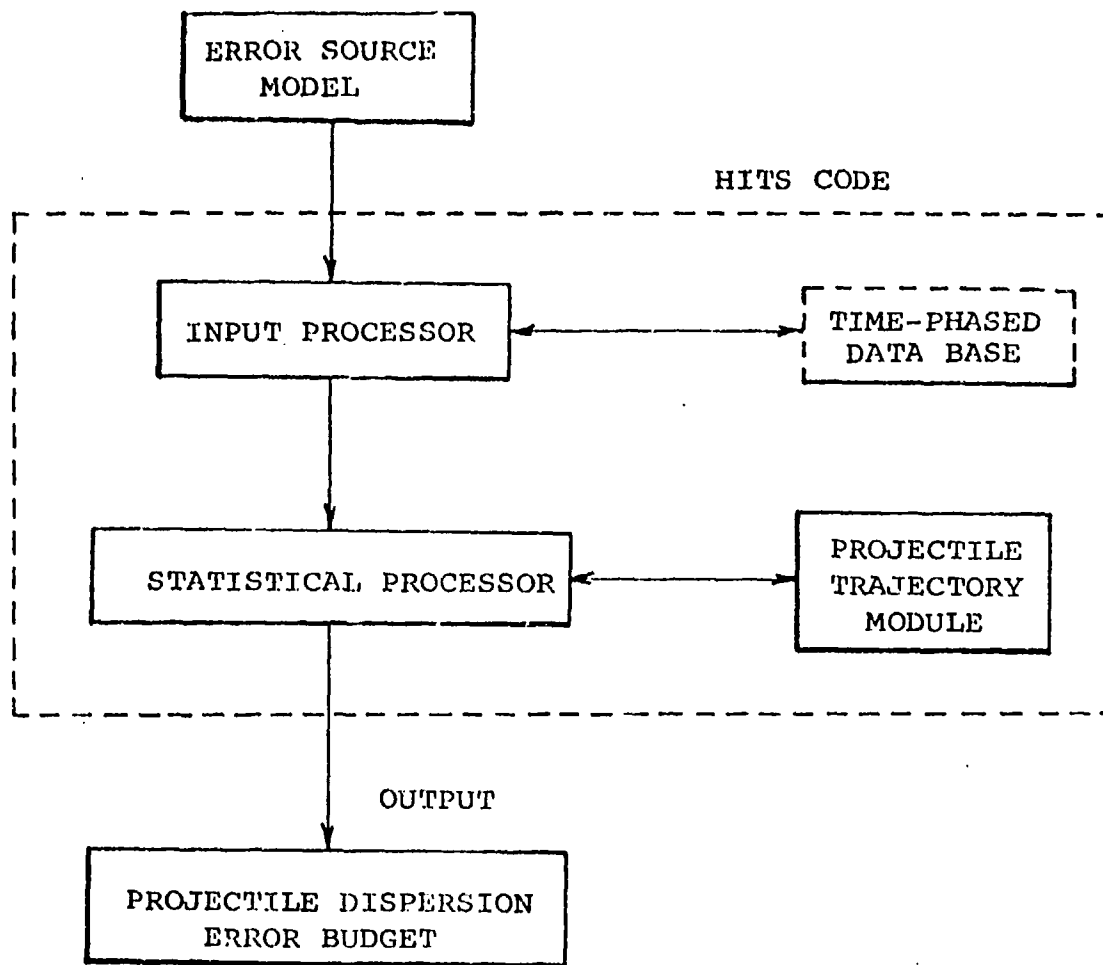


Figure 2-1 HITS Flow Diagram

2.1.1 Input Processor

The Input Processor reads error source model information from the input data stream. Two operations are performed once all the data has been read: (1) the input data is checked against a master list to determine whether crucial information is missing, and (2) the experimenter is informed as to the completeness of the stated error model. Missing data is automatically filled in with preset values.

The input data must define each element of the error source model (be it aerodynamic coefficient, physical dimension, or whatever) either as a constant or as a random variable. Random variables must be defined as to distribution type, mean or nominal value, and standard deviation (i.e., the $1-\sigma$ uncertainty). Gaussian, Rayleigh and uniform random variables are particularly easy to input, but any distribution is accepted. The input information is passed to the Statistical Processor.

2.1.2 Statistical Processor

The Statistical Processor derives dispersion statistics by referring to the Projectile Trajectory Module, as suggested by Figure 2-1. Each reference is the analytic equivalent of a ballistic range shot fired under the conditions specified by the Trajectory Module inputs. The Statistical Processor manipulates the Trajectory Module inputs in accord with the error source model uncertainties. The shot-by-shot dispersion returned by the Trajectory Module is compiled by the Statistical Processor to determine the dispersion error budget statistics.

2.1.3 Projectile Trajectory Module

The Projectile Trajectory Module consists of two sets of closed form parametric trajectory equations. The first computes the Fire Control predicted trajectory and the second simulates the true flight path or "Real World" Trajectory. The Trajectory Module returns the difference to the Statistical Processor.

The Fire Control equations are solved for the nominal time required to reach nominal range and the coordinates and velocity of the projectile at that time. This fixes the aim point in time as well as space. Typically, the parameters of the Fire Control equations are set equal to the error source model nominal values, so that the aim point is based on nominal muzzle velocity, weight, drag, etc. Simplifying assumptions found in operational

Fire Control computers can be simulated by judiciously selecting these parameters. Dispersion is computed relative to the aim point.

The Real World trajectory equations are solved for the actual projectile coordinates and velocity at nominal time and at nominal range. The distance between the projectile position at nominal time and the aim point defines dispersion for engagements with rapidly moving targets, where time of arrival is important. The displacement of the projectile position at nominal range from the aim point defines dispersion for engagements with slowly moving targets. The parameters of the Real World equations change from shot to shot and contain perturbations in accord with the error source model uncertainties. Thus, the dispersion returned to the Statistical Processor is consistent with the error source model and incorporates the effects of Fire Control simplifying assumptions.

2.2 Analysis Options

The Statistical Processor of Figure 2-1 was extracted from previously developed software. As a result, it has several desirable features unrelated to statistical dispersion assessment. The Statistical Processor has four modes of operation: (1) Single Trajectory, (2) Range Check, (3) Analytical Statistical, and (4) Monte Carlo. The first two modes are of primary interest when designing projectiles or analyzing various launch phenomena. The latter two modes are statistical methods for quantifying dispersion. The following subsections define the modes and suggest usages.

2.2.1 Single Trajectory

This mode computes the dispersion associated with a single shot at a given range. The Single Trajectory mode would be particularly useful in selecting nominal design parameters such as spin rate. A nominal trim lift would be input and the spin rate varied to minimize dispersion at maximum range.

2.2.2 Range Check

This mode evaluates the dispersion associated with a systematic variation of input parameters, thereby providing valuable design data. For instance, a Range Check on nominal range (5000, 6000, ---, 10,000 ft) would generate a complete trajectory. Alternatively, a Range Check on static margin (4, 5, and 6%) and spin rate (300, 400, and 500 rad/sec) would produce simulated shots with all possible combinations of static margin and spin rate. Comparison of the dispersions might suggest the optimal combination.

2.2.3 Analytical Statistical

The Analytical Statistical mode evaluates first and second order dispersion sensitivity coefficients by varying the trajectory inputs about the error source nominal. This mode computes, for example, the jump angle sensitivity to cross wind uncertainty; that is, the milliradians per ft/sec. The Analytical Statistical mode then uses the sensitivity coefficients to evaluate the average dispersion as well as the $1-\sigma$ uncertainty. The dispersion probability distribution is not determined. The analytic techniques employed permit rapid, low-cost dispersion assessment.

2.2.4 Monte Carlo

The Monte Carlo mode simulates ballistic range tests. It evaluates not only dispersion statistics (i.e., mean values and σ 's) but also the dispersion distribution. Numerous trajectories are computed using computer generated random numbers to simulate error source model uncertainties. Each simulated trajectory is comparable to a ballistic range shot, at a fraction of the cost. A test sequence is summarized by statistical indices and bar charts showing the number of times the projectiles had dispersions of say 0 to 0.25 milliradians, 0.25 to 0.50 milliradians, etc. The Monte Carlo mode gives the most complete dispersion picture.

2.3 Trajectory Equations

The Projectile Trajectory Module contains both Fire Control and Real World closed form trajectory equations. Both mechanize the theory presented in Appendix C. Section 2.3.1 discusses the rationale for the selection of closed form equations. An overview of the theory is presented in Section 2.3.2. Although the trajectory equations are used here to define dispersion, there are other potential applications, which include incorporation in operational Fire Control computers, the computation of firing tables, as well as any other situation requiring rapid, accurate, low-cost trajectory determination.

2.3.1 Closed Form Rationale

One approach to determining the Projectile Trajectory would be to perform full numerical simulations of the six degrees of freedom (6 DOF) differential equations of motion. However, since a Monte Carlo approach requires at least hundreds and possibly thousands of simulations in order to determine the dispersion

accurately, the comparatively long running times of 6 DOF simulations makes this approach impractical. Another approach is to develop approximate trajectory equations which can be rapidly evaluated, while at the same time include the effects of the major error sources. HITS uses the latter approach. The projectile equations of motion were simplified and the trajectory was determined in closed form. Perturbation equations are solved to refine the trajectory model. Six degrees of freedom simulations were conducted to authenticate the trajectory model approximations and are presented in Appendix C. In all cases the trajectory model checked out to within engineering tolerances.

2.3.2 Summary Description

The trajectory equations are composed of three parts, as illustrated in Figure 2-2. The first part is a particle trajectory which accounts for a large number of drag effects and the effect of constant velocity winds. The second part is a perturbation model which evaluates the effect of lift on cross-range dispersion. The final part is a perturbation model to account for the downrange effect of lift induced drag. The particle trajectory is evaluated first. Corrections for lifting effects are then computed via the perturbation equations and applied to the particle trajectory. The following paragraphs discuss the three components of the Trajectory Module.

Particle Trajectory

The particle trajectory equations shown in Figure 2-2 account for the downrange effects of muzzle velocity, projectile weight, drag, etc., as well as crossrange and downrange winds. The equations assume a zero angle of attack, a uniform density atmosphere, and the absence of gravity, and a variable drag coefficient. Classically, particle trajectories have approximated the drag coefficient as being independent of velocity. Although adequate for conventional muzzle velocities and short flight times, the drag coefficient model for hypervelocity projectiles must include the variation with velocity in order to avoid substantial errors, since the drag coefficient of typical hypervelocity projectiles (i.e., slender bodies) can vary by a factor of 3 to 4 over the velocity range of interest. The HITS particle trajectory models the drag coefficient as

$$C_D = C_{D_\infty} + \frac{K_D}{v^2} \quad (2.3-1)$$

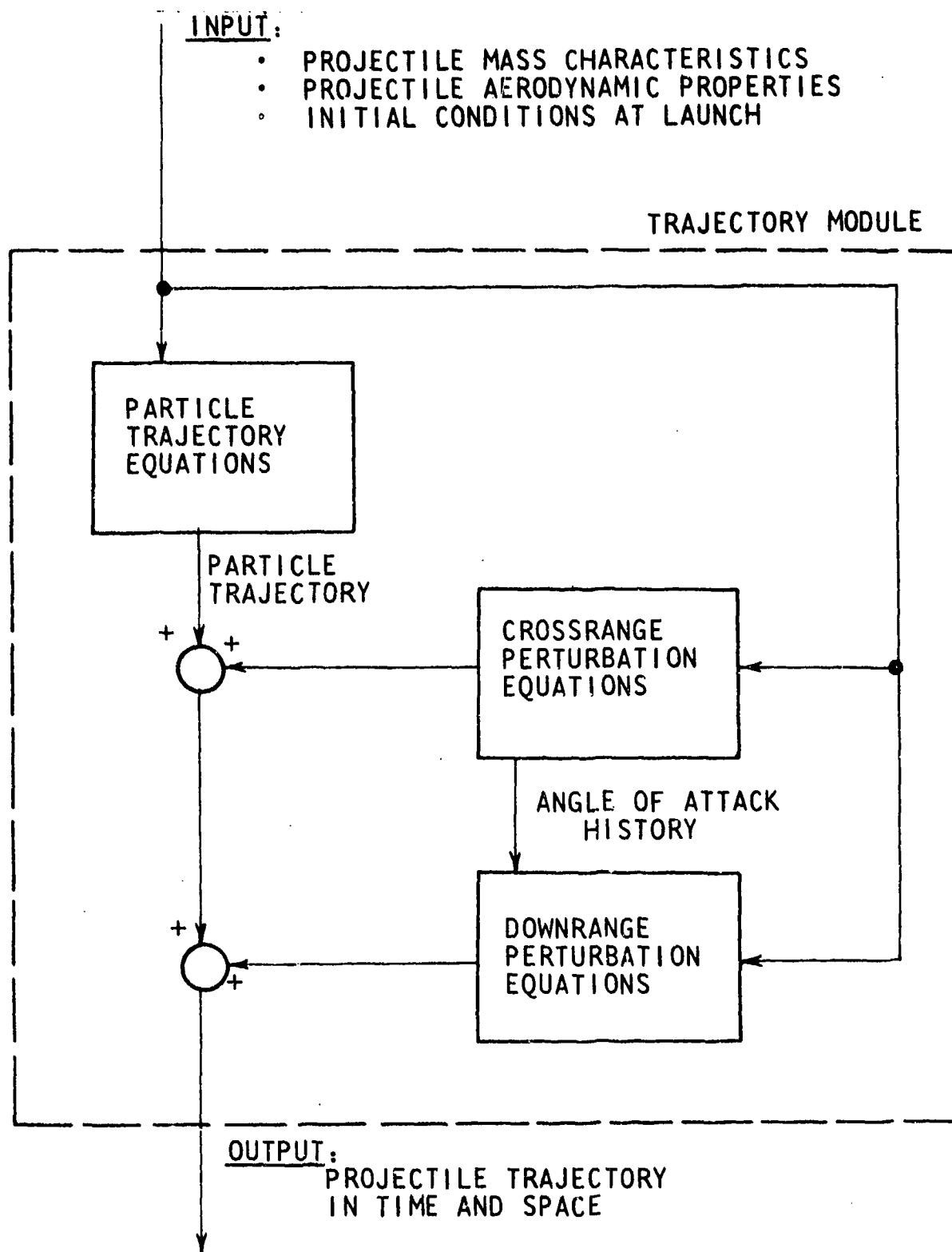


Figure 2-2 Trajectory Equations Computational Procedure

which can also be expressed in terms of the ballistic coefficient
($\beta = W/C_D A$):

$$\frac{1}{\beta} = \frac{1}{\beta_{\infty}} + \frac{K}{v^2} \quad (2.3-2)$$

where $C_{D\infty}$ and K_D or β_{∞} and K are empirically determined constants. This is an excellent model in the subsonic and supersonic velocity regimes. Ablative effects (i.e., tip recession and mass loss) can be approximated by making appropriate changes to the drag model parameters. Thus, including the drag variation with velocity opens the door to considering the effects of ablation.

Crossrange Perturbation Equations

This component of the Trajectory Model computes perturbations to the particle trajectory due to spurious lift forces caused by angle of attack oscillations and static trim angles. The angle of attack history is determined. The effect of a constant projectile spin rate is included. Velocity is assumed to be constant, so the crossrange error arising from transient roll resonance is not determined. Aerodynamic crossrange perturbation forces are significant only during the first few pitch oscillations. Since these occur near the muzzle while velocity is essentially constant, the constant velocity assumption is realistic. The crossrange perturbations are applied to the particle trajectory as a function of range. The particle trajectory includes the variation in velocity. The solution of these equations are applied as corrections to the particle trajectory as indicated in Figure 2-2.

Downrange Perturbation Equations

The downrange perturbation equations incorporate the effects of angle of attack variations on the drag coefficient. They include the effect of a constant spin rate. During the period when the projectile is oscillating, the perturbations to the velocity and flight time are determined by computing a mean drag coefficient with and without the angle of attack history as determined by the crossrange perturbation equations. The perturbations are applied as corrections to the velocity and flight time as determined by the (zero angle of attack) particle trajectory. The trajectory is carried beyond the point of angle of attack convergence by restarting the particle trajectory solution with the corrected velocity and time at convergence as initial conditions.

2.4 Related Topics

This section discusses three subjects which are peripheral to HITS but pertinent to projectile dispersion analyses. Section 2.4.1 presents the various statistics customarily used to describe crossrange dispersion. Correlation analysis as related to error source models, is discussed in Section 2.4.2. Section 2.4.3 considers the effects of engagement geometry on target-fixed crossrange dispersion. Each of these topics has a bearing on the calculations of the HITS code.

2.4.1 Crossrange Dispersion Indices

Crossrange dispersion is an important measure of accuracy, particularly for engagements involving slowly moving targets and high rate-of-fire weapons. Prior to this, crossrange dispersion was not included in the HITS code. This section defines the various dispersion indices and presents conversion factors. Equations are given for the jump angle in greater detail.

Two statistical indices are commonly used to summarize crossrange dispersion.

- The radial standard deviation is denoted by σ_{RSD} .
- The radius of the 80% circle (R80) is the radius of the circle which encloses 80% of the impact points.

$$P \left[\sqrt{x^2 + y^2} \leq R80 \right] = 0.80 \quad (2.4-1)$$

- The radius of the 80% circle is denoted R80.
- The Radial Standard Deviation (RSD) is the rms value of the impact point from the aim point, (x, y) ,

$$RSD = \sqrt{E[x^2 + y^2]} \quad (2.4-2)$$

where E denotes the ensemble average.

- The "jump angle", J , is the average displacement of the impact point from the aim point, (x, y) ,

$$J = E \left[\sqrt{x^2 + y^2} \right] \quad (2.4-3)$$

when the impact coordinates x and y are expressed as angular deflections.

- There is a 50% probability the impact point coordinates relative to the aim point, (x, y) , will lie between two parallel lines which are equidistant from the origin and are separated by twice the Linear Error Probable (LEP).

The aim point is usually defined as the centroid of the impact points.

Since the components of crossrange dispersion can usually be assumed to have equal magnitude and be independent Gaussian random phenomenon, the statistical dispersion indices are proportional. The scale factors are presented in Figure 2-3. The HITS code evaluates the crossrange standard deviation, σ_{CR} . The probability of occurrence associated with each index is given in Table 2-1.

2.4.2 Correlation Analysis - An Example

The error source model of Table 1-2 treats each source as an independent random phenomenon. Whereas this is a perfectly valid assumption in most instances, the potentiality exists that there are some subtle correlations (i.e., interrelationships) which could affect the dispersion statistics. The HITS code can evaluate the effects of correlation provided the correlation coefficient can be determined. This section illustrates analytic procedures for evaluating correlation coefficients from design information. The most direct approach would be to use ballistic range test data to evaluate the correlation. However, for the purposes of illustration, it is assumed this avenue is not open and the correlation must be evaluated from design information. The example is incomplete in the sense that no concrete conclusions are drawn. The objective is to illustrate that correlation coefficients are amenable to analyses based on physical considerations. The correlation coefficient is theoretically discussed in Appendix A.3.3.

Intuitively, muzzle velocity variations, δV and projectile weight variations, δW_p , would be expected to be negatively correlated to some extent, since a heavier projectile should result in a lower muzzle velocity. These two sources are typically large contributors to downrange dispersion if they are considered uncorrelated. However, because a heavier projectile would be expected to slow down less, it might be suspected that when correlation is taken into account, weight and muzzle velocity effects compensate, and result in significantly less net downrange dispersion. Since they are likely to be correlated and their

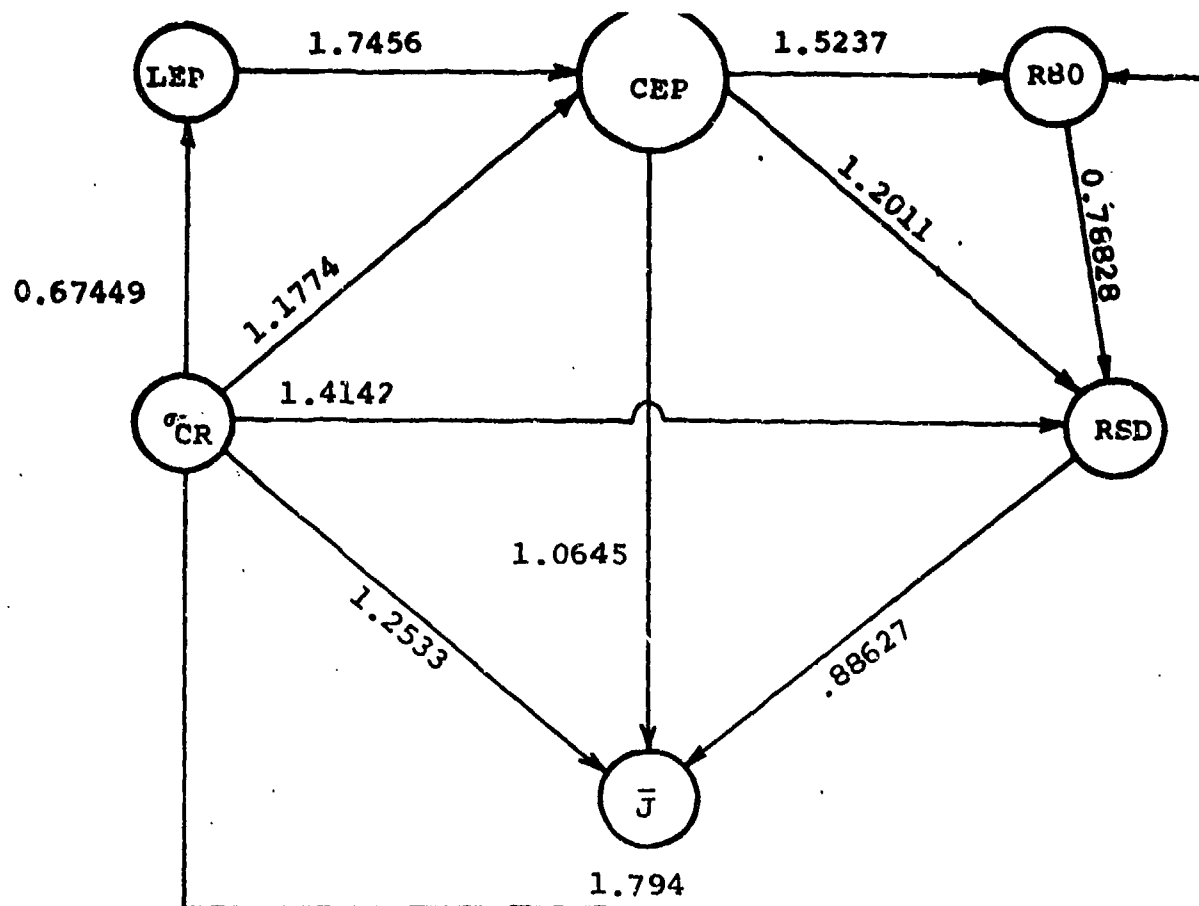


Figure 2-3 Gaussian Scale Factors

Table 2-1 Probabilities of Occurrence

Circle Radius	Probability of Occurrence (%)
LEP	20.3
σ_{CR}	39.3
CEP	50.0
$\bar{J}$	54.4
RSD	63.2
R80	80.0

correlation potentially could have a substantial effect on down-range dispersion, muzzle velocity and projectile weight variations are selected for the purpose of discussing analyses procedures.

Presuming the gun is a relatively uniform accelerator suggests an analysis of the kinetic energy imparted to the launch package might be the most direct approach to determining the correlation between projectile weight and muzzle velocity variations. (An analysis of the momentum imparted by launch might be a more fruitful approach in a launcher with an impulsive launch cycle.) The kinetic energy of the sabot/projectile is

$$K_e = \frac{1}{2} \frac{W_p + W_s}{g} v^2 \quad (2.4-4)$$

where W_s is the sabot weight and g is the acceleration due to gravity. Viewing weight as the dependent variable, Eq. (2.4-4) can be linearized about the nominal launch conditions and solved for the variation in velocity,

$$\frac{\delta v}{v} = \frac{1}{2} \left\{ \frac{\delta K_e}{K_e} - \left[\eta \frac{\delta W_p}{W_p} + (1 - \eta) \frac{\delta W_s}{W_s} \right] \right\} \quad (2.4-5)$$

where η is the design ratio of projectile weight to the total package weight. Equation (2.4-5) determines the muzzle velocity-projectile weight correlation coefficient. It is

$$\rho_{vW_p} = \frac{1}{2} \left\{ \frac{\sigma_{K_e}/K_e}{\sigma_v/v} \rho_{K_e W_p} - \left[\eta \frac{\sigma_{W_p}/W_p}{\sigma_v/v} + (1 - \eta) \frac{\sigma_{W_s}/W_s}{\sigma_v/v} \rho_{W_s W_p} \right] \right\} \quad (2.4-6)$$

The correlation coefficient $\rho_{W_s W_p}$ is the correlation coefficient of the projectile and sabot weights. Since these are the result of different manufacturing processes, $\rho_{W_s W_p}$ is most probably zero. σ_{W_p} and σ_v are the standard deviations of the projectile weight and the muzzle velocity and are given by the basic error source model. σ_{K_e} is the standard deviation of the kinetic energy imparted to the launch package. Assuming the launch cycle is efficient, σ_{K_e} should be dominated by the variations in the chemical energy released by the propellant. (Other factors influencing σ_{K_e} would include heat absorbed by the gun, the energy lost by in-barrel balloting and friction, and the residual energy in the gun gases.) The launch package kinetic energy-projectile

weight correlation coefficient, ρ_{KeWp} , is an interesting one. At first these two quantities would appear to be unrelated ($\rho_{KeWp} = 0$). However, the projectile weight has an influence on the diameter of the projectile which in turn affects the tightness of the seal between the launch package and the barrel. By this argument an increased weight should cause more kinetic energy to be imparted to the launch package. Analysis of the projectile geometry and consideration of typical variations in material mass densities would determine the appropriate value of ρ_{KeWp} . All values would be substituted into Eq. (2.4-5) to determine the projectile weight-muzzle velocity correlation coefficient.

This example has illustrated the determination of correlation coefficients. They may be evaluated from ballistic range test data or design analyses. The analytical procedures are straightforward, systematic, and scientific and are based on the first principles of the physical sciences.

2.4.3 Target-Fixed Crossrange Dispersion

The HITS code defines projectile dispersion in a coordinate frame fixed with respect to the gun emplacement. The dispersion must be transformed into a frame moving with the target in order to assess weapon system effectiveness. The conversion to target-fixed coordinates is usually performed during an engagement analysis and can be quite elaborate. This section presents a simplified procedure which allows the analyst to quickly determine the approximate target-fixed crossrange dispersion.

For the intercept geometry of Figure 2-4, the target-fixed crossrange dispersion in milliradians, σ_{TCD} , is given by

$$\sigma_{TCD} = \sqrt{\sigma_{CR}^2 + (S_{C/D} \sigma_{DR})^2} \quad (2.4-7)$$

where σ_{CR} and σ_{DR} are the crossrange and downrange projectile dispersion standard deviations in milliradians and percent of range, respectively, and

$$S_{C/D} = \left| \frac{10 \sin \psi}{\frac{V_p}{V_t} + \cos \psi} \right| \quad (2.4-8)$$

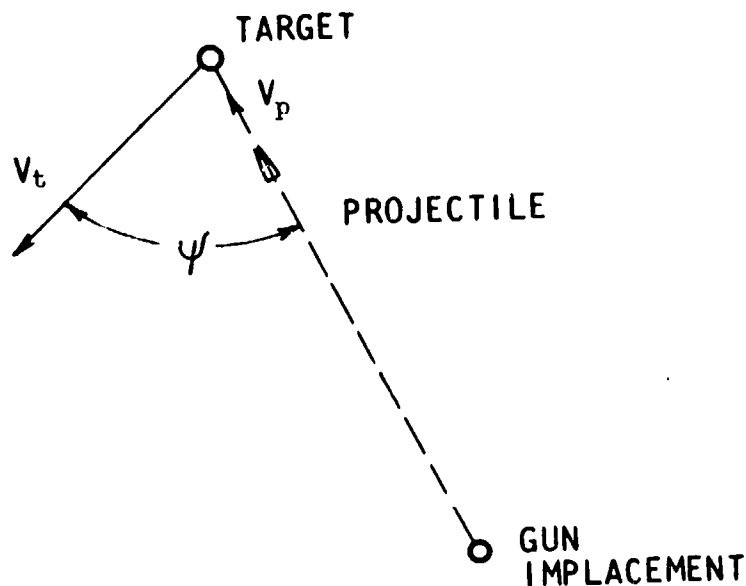


Figure 2-4 Intercept Geometry

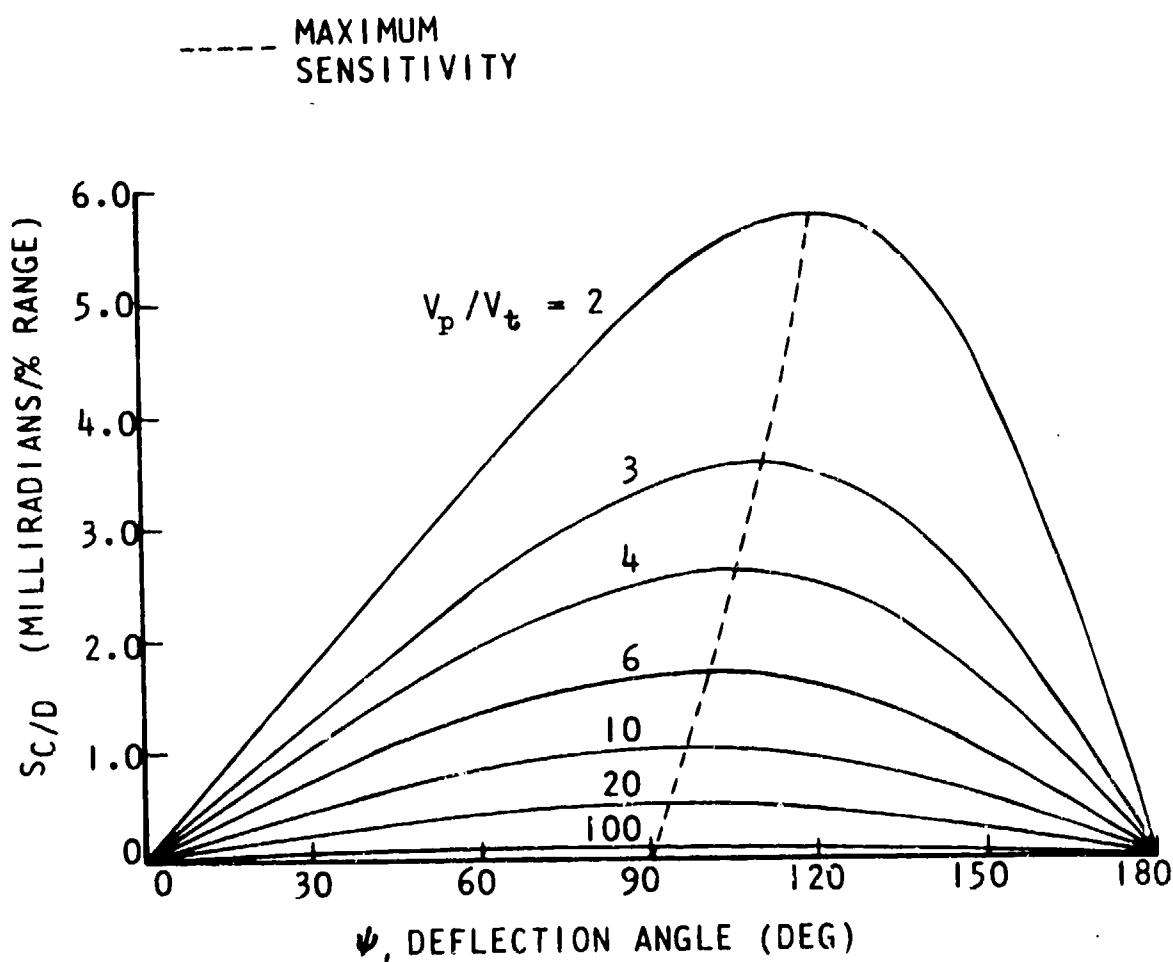


Figure 2-5 Crossrange Sensitivity to Downrange Dispersion

where V_p/V_t is the projectile to target velocity ratio in the vicinity of the target, and ψ is the deflection angle, i.e., the angle between the target velocity vector and the bore sight.

The coefficient $S_{C/D}$ is the equivalent crossrange dispersion sensitivity to downrange dispersion. The sensitivity coefficient is evaluated in Figure 2-5 for representative velocity ratios. Maximum sensitivity occurs at a deflection angle of

$$\psi_{\max} = 90^\circ + \sin^{-1} \left(\frac{V_t}{V_p} \right) \quad (2.4-9)$$

The maximum sensitivity is

$$(S_{C/D})_{\max} = \frac{10}{\sqrt{(V_p/V_t)^2 - 1}} \quad (2.4-10)$$

and can be used to perform a worst-case analysis.

This section has presented a simple approximate technique for converting projectile dispersion into a target-fixed coordinate frame. The results may be used in conjunction with target vulnerability, projectile terminal ballistics, and target lethality data to estimate the effect of projectile dispersion on weapon system effectiveness.

3.0 INPUT DESCRIPTION

This chapter describes the order and content of the input cards. The input for a single case consists of three card groups. Section 3.1 defines the Statistical Processor Control Card (Card Group 1). Input Processor Control Cards (Card Group 2) are discussed in Section 3.2. The Optional Input Processor Control Cards (Card Group 3) are described in Section 3.3. The card groups are defined by cook-book tables that give the card format and options. Comments in the tables alert the user to related, indepth discussions presented in other portions of the text. Section 3.4 illustrates encoding of an error source model.

The card groups for a single case must appear in the order suggested by the card group numbers. The card formats utilize fixed numeric fields to facilitate data base operations. IBM-029 keypunch control cards are presented to simplify keypunching. All integer variables must be right-hand justified. Multiple cases may be stacked. Each case in the stack must be complete in itself and not simply changes from the prior case.

3.1 Statistical Processor Control Card (Card Group 1)

The Statistical Processor Control Card is the first card appearing in the deck describing a single case. This card directs the Statistical Processor to enter a specific mode of operation. Table 3-1 defines the format of the card and the various options. Figure 3-1 illustrates the associated IBM-029 keypunch control card.

The Statistical Processor Control Card (Card Group 1) described in Table 3-1, defines 13 variables. Only IØPRNT and MCØPT play significant roles and are discussed in depth. IØPRNT and MCØPT control the mode of the Statistical Processor as described in the following two paragraphs.

Referring to Table 3-1, the IØPRNT variable serves a dual purpose. When IØPRNT = 0, the Statistical Processor identifies the case as either a Single Trajectory or Range Check simulation, and relinquishes control to the Input Processor. The Input Processor determines the proper mode from the contents of Card Group 2. Whenever IØPRNT $\geq$ 0, the Statistical Processor enters the Analytical Statistical mode and requests the Input Processor to supply the contents of Card Group 2.

Table 3-1 Statistical Processor Control Card (Card Group 1)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
1	IV	Dispersion analysis selector IV = 1 procedure described in this manual.	HITS has facilities to access (as yet undeveloped) alternate dispersion analysis procedures.
4	I0PRNT	Analytical Statistical mode calculation control. I0PRNT = 0 for Single Trajectory or Range Check modes = 1 calculation stops after limiting cases. = 2 calculation stops after unmixed first and second derivatives are evaluated. = 3 calculation stops after second order mean and first order variance are calculated. = 4 calculation stops after second order mixed partial derivatives are evaluated = 5 calculation stops after second order mean and first order variance are calculated for correlated independent variables. = 6 calculation stops after second order variance calculation, assumes all independent variables are Gaussian. = 7 calculation stops after second order variance for correlated independent variables, assumes all independent variables are Gaussian.	- I0PRNT ≥ 3 for Monte Carlo mode - The number of TYPE = 2, 3, and 4 variables (K234) and the number of dependent variables (KD) must be limited so that $K234 \leq 5000$ where: $K234 = KD(1+2K234), \quad I0PRNT \leq 2$ $K234 = KD(3+2K234), \quad I0PRNT \leq 3$ $K234 = KD \left[3 + K234 \left(2 + \frac{K234-1}{2} \right) \right], \quad I0PRNT \geq 4$ Otherwise, the C array will be exceeded and computation will terminate abnormally.

*Card Group 3 required

Table 3-1 Statistical Processor Control Card (Card Group 1) (cont'd)

Column(s)	Variable Name	Description	Comments
7	ISPRNT	Trajectory Module output suppression code ISPRNT = 0 trajectory module output suppressed = 1 trajectory module output printed at each reference to trajectory module	- ISPRNT = 1 provides extensive information. Usually used only in Single Trajectory or Range Check mode. - ISPRNT = 0 recommended for all Monte Carlo mode and most Analytical Statistical mode operations.
12	MCQOPT	Monte Carlo mode control variable MCQOPT = 0 no Monte Carlo experiments = 1 Monte Carlo mode to be exercised	
17	MCALC	Monte Carlo control variable to bypass Trajectory module MCALC = 0 trajectory module specified by IV variable is used to evaluate all trajectories. = 1, 2, 3 a Taylor series approximation with (1) first order, (2) second order unmixed, and (3) full second order partial derivatives is used to approximate the full trajectory equations.	- MCALC = 0 recommended for all calculations. - MCALC = 1, 2, or 3 provides a more rapid evaluation at the expense of accuracy.
21-22	NCELL	Number of histogram cells for each independent and dependent variable. NCELL ≤ 20	See Appendix A.6.1 for guidelines.
24-27	NTRIAL	Number of Monte Carlo experiments to be conducted	See Appendix A.6.2 for guidelines.

Table 3-1 Statistical Processor Control Card (Card Group 1) (concl'd)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
29-32	IRJ1	Maximum number of Monte Carlo experiments to be rejected as outside internal ranges (predicted by Analytical Statistical mode) before computation is suspended.	IRJ1 should be large enough to allow for 10% NCILL rejections but small enough to abort simulations with gross input errors.
34-37	IRJ2	Maximum number of Monte Carlo experiments to be rejected as outside external ranges (TYPE = 8 variable) limits before computation is suspended.	See IRJ1 comment.
42	MCPRNT	Monte Carlo mode trajectory Module summary print control MCPRNT = 0 suppress summary trajectory information for all Monte Carlo experiments. = 1 print summary information for each trajectory calculation.	Summary print should be suppressed if large numbers of Monte Carlo experiments are performed.
44-47	IRRAND	Random number generator seed for Monte Carlo mode.	IRRAND should be a large odd positive integer such as 1201.
52	ISPLT	Control variable for automatic histogram bar chart plots. ISPLT = 0 no plots = 1 plot histograms	Plot facility is presently incomplete. Plot arrays are listed on the printed output. See Appendix B.2.5 for format.
57	ISPLT	Automatic histogram plot format control. ISPLT = 0 generate individual plots. = 1 overlay internal and external range histograms.	Presently has no effect on execution.

Drum Card

[illegible]

Figure 3-1 IBM-029 Keypunch Control Card for Card Group 1

The MCØPT variable of Table 3-1 determines whether or not the Statistical Processor will enter the Monte Carlo mode after completing Analytical Statistical mode calculations. If MCØPT = 1, the Statistical Processor automatically shifts into the Monte Carlo mode after the Analytical Statistical mode results have been used to set up the histograms.

3.2 Input Processor Control Cards (Card Group 2)

The Input Processor Control Cards appear second in the deck describing a single case. Each card directs the Input Processor to assign qualities to a trajectory variable. These qualities affect the calculations performed as described in Section 3.2.1. The trajectory variables are defined in Section 3.2.2.

3.2.1 Card Group 2 Assigned Qualities

Each Group 2 card defines two parameters: CØDE# and TYPE. CØDE# (i.e., code number) identifies the trajectory variable to which the qualities determined by the value of TYPE are to be assigned. This is one of the more interesting aspects of the HITS code. There are two basically different TYPE specifications that can be assigned. They are described in the succeeding paragraph. It is assumed the user wishes to determine the effect of projectile weight variations on downrange dispersion at nominal time, for the purposes of discussion.

Whenever $1 \leq \text{TYPE} \leq 5$, the variable is assigned the distinction of being an independent variable, that is, an independent variable whose value is determined by input data rather than preset (or default) values. In the example, projectile weight is the independent variable and would be assigned a TYPE between 1 and 5 depending on whether it is to be treated as a deterministic constant (different than the preset), a range check variable, or a random variable. Whenever $7 \leq \text{TYPE} \leq 8$, the variable is defined to be a dependent variable, that is, a dependent variable for which printed output is to be produced. In the example, downrange dispersion at nominal time would be declared a TYPE = 7 or TYPE = 8 variable, since the user desires to have the dispersion displayed. Thus, the CØDE# identifies the variable and TYPE determines what is to be done with it. All independent variables not mentioned in Card Group 2 are given their preset values. All dependent variables not mentioned are computed but not printed.

A Group 2 card may require more information than just the CØDE# and TYPE to completely define the variable. Table 3-2

Table 3-2 Input Processor Control Cards (Card Group 2)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
3-5	CODE#	Independent/dependent variable identification number.	• See Table 3-3 for definitions. • Group 2 Cards may appear in any order. • Integer variable ★ Negative value (e.g. "-1") required to indicate end of Card Group 2. • CODE# is an address in the GE array.
7	TYPE	Statistical characterization variable for independent and dependent variables. <u>Independent Variables:</u> TYPE = 1 Range Check variable to be parametrically varied over the NT values on the immediately succeeding card(s).	• Every Group 2 Card must define TYPE. • Integer variable • Maximum of three TYPE = 1 variables per Card Group 2 data set. • Minimum of one and a maximum of ten dependent variables (TYPE = 7 and TYPE = 8) per Card Group 2 data set. • TYPE = 1 variables are incompatible with TYPE = 2, 3, or 4 variables. These incompatible types should never appear in the same Card Group 2 data set. ★ Additional data is required for TYPE = 1 variables. • Card format: 5 floating point fields of length 13 (beginning in column 1) per card. • Use multiple cards if required. • NT is defined later in this table.

Table 3-2 Input Processor Control Cards (Card Group 2) (cont'd)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
7	TYPE (cont'd)	TYPE = 2 A Gaussian random variable TYPE = 3 A uniformly distributed random variable. TYPE = 4 Arbitrarily distributed random variable. Immediately succeeding cards must define the probability density function at NT points.	• Gaussian random variables (TYPE = 2) are discussed in Appendices A.4.2 and A.4.3. • Uniformly distributed random variables (TYPE = 3) are discussed in Appendix A.4.1. ★ Additional data required for arbitrarily distributed (TYPE = 4) random variables. • Card format: 5 floating point fields of length 13 each (beginning in column 1) per card. • Multiple cards will be required. • Probability Density Function (PDF) represented by two paired tables. Ordinate values (i.e., PDF values) must appear first. Corresponding abscissa values must appear second with abscissa table beginning on a new card. • PDF assumed to be zero outside range of abscissa table, so use enough points. • Abscissa table must contain equally spaced values. • NT is defined later in this table.

Table 3-2 Input Processor Control Cards (Card Group 2) (cont'd)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
7	TYPE (cont'd)	TYPE = 5 Deterministic variable TYPE = 6 Reserved for future use <u>Dependent Variables:</u> TYPE = 7 Variable whose histogram range is determined by Analytical Statistical mode. TYPE = 8 Variable for which histogram is to be constructed with a range supplied by the user.	• TYPE = 5 variables simply override preset (default) values. • TYPE = 6 should not be used. • Histogram INTERNAL RANGE and USER RANGE are identical • Histogram INTERNAL RANGE determined by Analytical Statistical mode. • Histogram USER RANGE determined by user supplied values.
9-20	VALUE	Value for variable. Subject to varying definitions TYPE = 1, 4, 7 VALUE not required. May be left blank. TYPE = 2, 3 VALUE is the expected or mean value of the random variable. TYPE = 5 VALUE is the desired deterministic value. TYPE = 8 VALUE is the histogram mid-point.	• Floating point variable • Interpretation given in Appendix A.3 for TYPE = 2 and TYPE = 3 variables.
22-33	TOL	Tolerance for variable. Subject to varying definitions. TYPE = 1, 4, 5, 7 TOL not required. May be left blank.	• Floating point variable.

Table 3-2 Input Processor Control Cards (Card Group 2 (concl'd))

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
22-33	TØL (cont'd)	TYPE = 2 TØL is the incremental length used in computing the partial derivatives. TYPE = 3 TØL defines the variable as being uniformly distributed on the interval (VALUE-TØL, VALUE + TØL) TYPE = 8 TØL defines the USERS RANGES as (VALUE-TØL, VALUE + TØL)	For TYPE = 2, TØL may be set equal to $TØL = 3 * STDEV$
35-46	STDEV	Standard deviation (σ) for some statistically defined variables. TYPE = 1,4,5,7,8 STDEV not required. May be left blank. TYPE = 2 STDEV is the standard deviation of the Gaussian random variable. TYPE = 3 STDEV is the standard deviation of the uniformly distributed random variable.	- Floating point variable - Interpretation given in Appendix A.3 for TYPE = 2 and TYPE = 3 variables. ★ For a TYPE = 3 variable uniformly distributed on the interval (VALUE-TØL, VALUE + TØL), STDEV must be set equal to $STDEV = TØL / \sqrt{3}$ $= 0.577350 * TØL$
48-50	NT	Table length for additional data on immediately succeeding cards. TYPE = 2,3,5,7,8 NT not required. May be left blank. TYPE = 1 NT is the number of values over which systematic variation is to occur. TYPE = 4 NT is the number of pairs of points used to represent the probability density function.	- Integer variable. - Succeeding cards required only for TYPE = 1 and TYPE = 4 variables. - For TYPE = 4, NT is the number of ordinate-abscissa pairs.

explains the details. In Table 3-2 hash marks are used to delimit the extent of the comments made in the right hand column. Comments identified by stars command action on the part of the user. Group 2 cards may be arranged in any order. IBM-029 key-punch control cards for Group 2 are presented in Figure 3-2.

3.2.2 Trajectory Variable Code Numbers

HITS uses "code numbers" to identify variables in input data cards and the printed output. CODE# in Card Group 2 is just the first occurrence of code numbers in the discussion. This section discusses Table 3-3 which catalogs the code numbers.

The code number of a variable is its address in the HITS' active storage array (i.e., the OE array). Table 3-3 lists the code numbers and their associated FORTRAN names, preset values, units, and analysis names. The code numbers also link the HITS variables to the closed form trajectory equations of Appendix C via Table C-1. An asterisk attached to a FORTRAN name denotes a potential independent variable. All others may be defined as dependent variables. The presets are the nominal values of the Table 1-2 error source model.

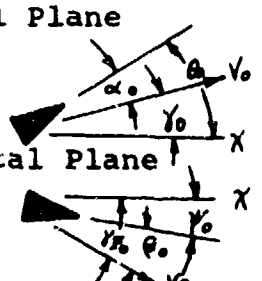
Referring to Table 3-3, the code numbers from 1 to 100 define Fire Control trajectory variables. Code numbers from 101 to 200 are the Real World trajectory variables. Variables with code numbers from 201 to 300 are system controls. These are particularly important since they can greatly reduce the amount of input data and the number of computer runs. Note that variable 203, (i.e., IFC) distinguishes between Real World and Fire Control trajectory solutions. Variables 301 to 400 are measures of dispersion. Variables 401 to 500 are computed trajectory quantities. Code numbers from 501 to 600 denote variables defining the terminal conditions of the Real World and Fire Control trajectories.

3.3 Optional Input Processor Control Cards (Card Group 3)

Optional Input Processor Control Cards (Card Group 3) appear third (last) in the deck describing a single case. Card Group 3 may not appear when the Statistical Processor is operated in either the Single Trajectory or Range Check modes. It has no effect on the Monte Carlo mode. Only the Analytical Statistical mode is affected by Group 3 data.

Each Group 3 card directs the Analytical Statistical Processor to treat two of the statistically defined independent

Table 3-3 HITS Trajectory Parameter List

Code Number	FORTTRAN Name	Preset	Units	Remarks
1	VOF*	11,000.	ft/sec	F.C. ⁽¹⁾ Nozzle Velocity
2	WXF*	0.	ft/sec	F.C. Steady Winds: Down Range Horiz. Cross Wind
3	WZF*	0.	ft/sec	
4	RHOF*	2.378×10^{-3}	slugs-ft ³	F.C. Atmospheric Density
5	CX1F*	3.585×10^{-2}	--	F.C. Projectile Drag Model $C_X = C_{X\infty} + K/V^2$ Fit to (CX1F, V1F) & CX2F, V2F) CX2F < CX1F VX2F > VX1F
6	V1F*	1.674×10^4	ft/sec	
7	CX2F*	1.195×10^{-1}	--	
8	V2F*	3.906×10^3	ft/sec	
9	XNF*	10,000	ft	F.C. Nominal Range
10	CNAF*	1.9767	1/rad	F.C. Aerodynamic Effects: • C_{N_a} • Static Margin • C_{m_q} • Magnus Moment $C_{m_p a}$ • Static Trim ($C_m=0$) -Angle of Attack -Angle of Sideslip
11	SMF*	0.062	--	
12	CMQF*	-7.5	--	
13	CMPAF*	0.	--	
14	ATRMSF*	0.	deg	
15	BTRMSF*	0.	deg	
16	THETOF*	0.	deg	F.C. Initial Attitude: θ_0 Vertical Plane ψ_0 $\dot{\theta}_0$ Horizontal Plane $\dot{\psi}_0$ 
17	PSIOF*	0.	deg	
18	TDOTOF*	0.	rad/sec	
19	PDOTOF*	0.	rad/sec	

*Denotes Potential Independent Variable

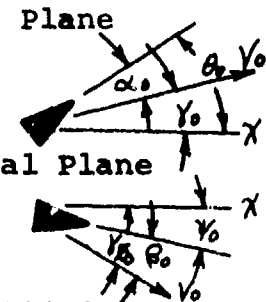
(1) Indicates Fire Control Parameter

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
20	GAMOF*	0.	deg	F.C. Initial Attitude (Cont'd) γ_0 γ_{z_0}
21	AZOF*	0.	deg	
22	AF*	3.068×10^{-3}	ft ²	F.C. Projectile Physical Characteristics: A - Base Area ($\pi D^2/4$) L - Length D - Base Diameter W - Weight I_x - Roll Moment of Inertia I_y - Pitch Moment of Inertia
23	ELLF*	3.1×10^{-1}	ft	
24	DF*	6.25×10^{-2}	ft	
25	WF*	1.1×10^{-1}	lbs	
26	AIXF*	1.235×10^{-6}	slugs-ft ²	
27	AIYF*	2.110×10^{-5}	slugs-ft ²	
28	PF*	400.	rad/sec	F.C. Spin Rate
29	ALPCØF*	0.5	deg	F.C. Angle of Attack Oscillation Convergence Criterion
30	CAAF*	0.	1/deg ²	F.C. Second Order Drag Effect $\Delta C_x = \text{CAAF } \alpha^2$

*Potential Independent Variable

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
101	V0*	11,000	ft/sec	R.W. ⁽¹⁾ Muzzle Velocity
102	WX*	0.	ft/sec	R.W. Steady Winds Down Range Horiz. Cross Wind
103	WZ*	0.	ft/sec	
104	RHØ*	2.378x10 ⁻³	slugs-ft ³	R.W. Atmospheric Density
105	CX1*	3.585x10 ⁻²	--	R.W. Projectile Drag Model $C_X = C_{X_{00}} + K/V^2$ Fit to (CX1,V1) & (CX2,V2) CX2 < CX1 VX2 > VX1
106	V1*	1.674x10 ⁴	ft/sec	
107	CX2*	1.195x10 ⁻¹	--	
108	V2*	3.906x10 ³	ft/sec	
109	XN*	10,000	ft	R.W. Nominal Range
110	CNA*	1.9767	1/rad	R.W. Aerodynamic Effects: • C_{N_a} • Static Margin • C_{m_q} • Magnus Moment $C_{m_{pa}}$ • Static Trim ($C_m=0$) -Angle of Attack -Angle of Sideslip
111	SM*	0.062	--	
112	CMQ*	-7.5	--	
113	CMFA*	0.	--	
114	ATRMS*	0.	deg	
115	BTRMS*	0.	deg	
116	THETO*	0.	deg	R.W. Initial Attitude θ_0 Vertical Plane ψ_0 Horizontal Plane 
117	PSIO*	0.	deg	
118	TDØTO*	0.	rad/sec	
119	PDØTO*	0.	rad/sec	

*Potential Independent Variable

(1) Denotes Real World Parameter

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
				R.W. Initial Attitude (Cont'd)
120	GAM0*	0.	deg	γ_0
121	AZ0*	0.	deg	γ_{z_0}
				R.W. Projectile Physical Characteristics
122	A*	3.068×10^{-3}	ft ²	A - Base Area ($\pi D^2/4$)
123	ELL*	3.1×10^{-1}	ft	L - Length
124	D*	6.25×10^{-2}	ft	D - Base Diameter
125	W*	1.1×10^{-1}	lbs	W - Weight
126	AIX*	1.235×10^{-6}	slugs-ft ²	I _x - Roll Moment of Inertia
127	AIY*	2.110×10^{-5}	slugs-ft ²	I _y - Pitch Moment of Inertia
128	P*	400.	rad/sec	R.W. Spin Rate
129	ALPCØN*	0.5	deg	R.W. Angle of Attack Oscillation Convergence Criterion
				R.W. Second Order Drag Effect
130	CAA*	0.	1/deg ²	$\Delta C_X = CAA \ a^2$
				Externally Supplied SEP Center
131	XBART*	0.	ft	$\delta \bar{x}_T$ (Displacements from Fire Control Nominal Aim Point)
132	YBART*	0.	ft	$\delta \bar{y}$
133	ZBART*	0.	ft	$\delta \bar{z}_T$
				Externally Supplied CEP Center
134	YBARX*	0.	ft	$\delta \bar{y}_x$ (Displacements from Fire Control Nominal Aim Point)
135	ZBARX*	0.	ft	$\delta \bar{z}_x$

*Potential Independent Variable

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
201	IA	0.	--	System Control IFCRW=1 F.C. Parameters are set equal to R.W. nominal values IFCRW=0 F.C. Parameters determined solely by inputs
202	IFCRC*	0.	--	System Control IFCRC=1 F.C. Parameters are set equal to R.W. values for each Range Check Combination IFCRC=0 F.C. Parameters determined solely by inputs.
203	IFC	1.	--	Internal Sequencing Variable IFC = 1 F.C. Solution IFC = 0 R.W. Solution
204	IDUMP*	0.	--	System Print Control IDUMP = 1 Print lots of intermediate data IDUMP = 0 Print summary data only
205	ICONV*	-10.	--	Trajectory Module Iteration Control TOL = 10.ICONV
206	ICNCL*	1.	--	Trajectory Aerodynamic Control Parameter ICNCL = 1 $C_{L\alpha}$ used in crossrange perturbation. Recommended for most applications. ICNCL = 0 $C_{N\alpha}$ used in crossrange perturbation.

* Potential Independent Variable

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
301	XBARTI	--	ft	Internally Computed SEP Center δx_T (Displacements from Fire Control Nominal Aim Point) δy_T δz_T
302	YBARTI	--	ft	
303	ZBARTI	--	ft	
304	YBARXI	--	ft	Internally Computed CEP Center δy_x (Displacements from Fire Control Nominal Aim Point) δz_x
305	ZBARXI	--	ft	
306	DXT	--	ft	Projectile Position at Nominal Time δx_t (Displacement from Fire Control Nominal Aim Point) δy_t δz_t
307	DYT	--	ft	
308	DZT	--	ft	
309	DYX	--	ft	Projectile Position at Nominal Range δy_x (Displacement from Fire Control Nominal Aim Point) δz_x
310	DZX	--	ft	
311	DVXT	--	ft/sec	Velocity Difference at Nominal Time δv_{xt} (Difference from F.C. Value at Nominal Time) δv_{yt} δv_{zt}
312	DVYT	--	ft/sec	
313	DVZT	--	ft/sec	
314	DVXX	--	ft/sec	Velocity Difference at Nominal Range δv_{xx} Difference from F.C. Value at Nominal Time) δv_{yx} δv_{zx}
315	DVYX	--	ft/sec	
316	DVZX	--	ft/sec	
317	DT	--	sec	δt Time to Nominal Range less F.C. Nominal Time to Nominal Range

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
318	RADT ⁺	--	ft	Radial Displacement from Externally Supplied SEP Center
319	RADX ⁺	--	ft	Radial Displacement from Externally Supplied CEP Center
320	RHØT ⁽¹⁾	--	ft	Radial Displacement from Internally Computed SEP Center
321	RHØX ⁽²⁾	--	ft	Radial Displacement from Internally Computed CEP Center
322	DXTI	--	ft	Projectile Position at Nominal Time δx_{ti} (Displacement from Internally Computed SEP Center) δy_{ti} δz_{ti}
323	DYTI	--	ft	
324	DZTI	--	ft	
325	DXTE	--	ft	Projectile Position at Nominal Time δx_{te} (Displacement from Externally Supplied SEP Center) δy_{te} δz_{te}
326	DYTE	--	ft	
327	DZTE	--	ft	
328	DYXI	--	ft	Projectile Position at Nominal Range δy_{xi} (Displacement from Internally Computed CEP Center) δz_{xi}
329	DZXI	--	ft	
330	DYXE	--	ft	Projectile Position at Nominal Range δy_{xe} (Displacement from Externally Supplied CEP Center) δz_{xe}
331	DZXE	--	ft	
332	DXDYT	--	ft ²	Correlations of Projectile Position at Nominal Time $\delta x_t \delta y_t$ (Displacements from Fire Control Nominal Aim Point) $\delta x_t \delta z_t$ $\delta y_t \delta z_t$
333	DXDZT	--	ft ²	
334	DYDZT	--	ft ²	

⁺TYPE = 8 only

(1) Requires Code Numbers 306, 307, and 308 be TYPE = 7 or TYPE = 8.

(2) Requires Code Numbers 309 and 310 be TYPE = 7 or TYPE = 8.

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
335	DYDZX	--	ft ²	Correlation at Nominal Range $\delta y_x \delta z_x$ (Displacements from F.C. Nominal Aim Point)
336	DXDYTI	--	ft ²	Correlations about Internally Computed SEP Center at Nominal Time $\delta x_{ti} \delta y_{ti}$ (Displacements from
337	DXDZTI	--	ft ²	$\delta x_{ti} \delta z_{ti}$ Internally Computed SEP Center)
338	DYDZTI	--	ft ²	$\delta y_{ti} \delta z_{ti}$
339	DYDZXI	--	ft ²	Correlations about Internally Computed CEP Center at Nominal Range $\delta y_{xi} \delta z_{xi}$ (Displacements from Internally Computed CEP Center)
340	DXDYTE	--	ft ²	Correlations about Externally Supplied SEP Center at Nominal Time $\delta x_{te} \delta y_{te}$ (Displacements from
341	DXDZTE	--	ft ²	$\delta x_{te} \delta z_{te}$ Externally Supplied SEP Center)
342	DYDZTE	--	ft ²	$\delta y_{te} \delta z_{te}$
343	DYDZXE	--	ft ²	Correlations about Externally Supplied CEP Center at Nominal Range $\delta y_{xe} \delta z_{xe}$ (Displacements from Externally Supplied CEP Center)

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
400	ALPHA	--	rad	$\alpha(t_N)$ or $\alpha(x_N)$ (RW/PC)
401	ATOT	--	rad	$\bar{\alpha}(t_N)$ or $\bar{\alpha}(x_N)$ (RW/PC)
402	ALPTRM	--	rad	Rolling Trim α
403	ADPHDO	--	rad/sec	$\dot{\alpha}_0$ β
404	B	--	--	
405	BETTRM	--	rad	Rolling Trim β
406	BETA0	--	rad	β_0
407	BETAD0	--	rad/sec	$\dot{\beta}_0$
408	BETA	--	rad	$\beta(t_N)$ or $\beta(x_N)$ (RW/PC)
409	ALPMAX	--	rad	$\bar{\alpha}_{UPPER}$
410	ALPMIN	--	rad	$\bar{\alpha}_{LOWER}$
411	CAYD	--	(ft/sec) ²	K_D
412	CD8	--	--	C_{D00}
413	CMA	--	1/rad	$C_{M\alpha}$
414	CMHTD	--	sec	$C_{M\dot{\theta}}$
415	CMHTA	--	sec	$C_{M\dot{\phi}\alpha}$
416	CAY1	--	rad	K_1
417	CAY2	--	rad	K_2
418	CDA0B	--	--	$\bar{C}_D (\bar{\alpha} = 0)$
419	CDAB	--	--	$\bar{C}_D$

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
420	DELTT	--	sec	δt
421	DELV	--	ft/sec	δv
422	DELT	--	sec	Δt
423	DELX	--	ft	Δx
424	DYDX0	--	rad	γ_0
425	DZDX0	--	rad	γ_{z0}
426	DYDT0	--	ft/sec	$v_y(0)$
427	DZDT0	--	ft/sec	$v_z(0)$
428	DELW	--	1/sec	$\Delta \omega$
429	DELLAM	--	1/sec	$\Delta \lambda$
430	EYEP	--	sec ²	l'
431	EMP	--	sec	m'
432	EOLT	--	--	$e^{\lambda_0 t'}$
433	EDLT	--	--	$e^{\Delta \lambda_0 t'}$
434	F	--	1/ft	f
435	H	--	ft/sec ²	h
436	PSID0	--	rad/sec	$\dot{\psi}_0$
437	PHI0	--	rad	ϕ_0
438	PHI1	--	rad	ϕ_1
439	PHI2	--	rad	ϕ_2

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
440	PSIO	--	rad	ψ_0
441	R1	--	rad	R_1
442	R2	--	rad	R_2
443	R3	--	rad	R_3
444	R4	--	rad	R_4
445	RTRIM	--	--	R_{TRIM}
446	TC	--	sec	t_c
447	TC0	--	sec	t_{c0}
448	TS	--	sec	t^*
449	TG0	--	sec	t_{G0}
450	TP	--	sec	t'
451	THETA0	--	rad	θ_0
452	THETD0	--	rad/sec	$\dot{\theta}_0$
453	TOL	--	--	$TOL = 10 \cdot ICONV$
454	VC	--	ft/sec	V_c
455	VC0	--	ft/sec	V_{c0}
456	VYA	--	ft/sec	V_{ya}
457	VZA	--	ft/sec	V_{za}
458	WX	--	ft/sec	W_x
459	WZ	--	ft/sec	W_z

Table 3-3 HITS Trajectory Parameter List (Cont'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
460	W0	--	sec ⁻¹	ω_0
461	W1	--	sec ⁻¹	ω_1
462	W2	--	sec ⁻¹	ω_2
463	WL2	--	sec ⁻²	$2(\omega_0^2 + \Delta\lambda^2)$
464	XLAM0	--	sec ⁻¹	λ_0
465	XLAM1	--	sec ⁻¹	λ_1
466	XLAM2	--	sec ⁻¹	λ_2
467	XNU1	--	rad	ν_1
468	XNU2	--	rad	ν_2
469	XG	--	ft	x_G
470	XMU	--	--	μ
471	XJAY	--	rad	$J\Lambda_Y$
472	XJAZ	--	rad	$J\Lambda_Z$
473	XJA	--	rad	$J\Lambda$
474	YA	--	ft	ψ_a
475	ZA	--	ft	Z_a
476	ALPH0	--	rad	α_0

Table 3-3 HITS Trajectory Parameter List (Concl'd)

Code Number	FORTTRAN Name	Preset	Units	Remarks
500	TN	--	sec	F.C. Established Nominal Time at Nominal Range
501	XTFC	--	ft	F.C. Computed Position at Nominal Time $x_{fc}(t_N)$ (Displacements from Muzzle) $y_{fc}(t_N)$ $z_{fc}(t_N)$
502	YTFC	--	ft	
503	ZTFC	--	ft	
504	VXTFC	--	ft/sec	F.C. Computed Velocity at Nominal Time $V_{x_{fc}}(t_N)$ $V_{y_{fc}}(t_N)$ $V_{z_{fc}}(t_N)$
505	VYTFC	--	ft/sec	
506	VZTFC	--	ft/sec	
507	XT	--	ft	Projectile Position at Nominal Time $x(t_N)$ (Displacements from Muzzle) $y(t_N)$ $z(t_N)$
508	YT	--	ft	
509	ZT	--	ft	
510	VXT	--	ft/sec	Projectile Velocity at Nominal Time $V_x(t_N)$ $V_y(t_N)$ $V_z(t_N)$
511	VYT	--	ft/sec	
512	VZT	--	ft/sec	
513	YR	--	ft	Projectile Position at Nominal Range $y(x_N)$ (Displacements from Muzzle) $z(x_N)$
514	ZR	--	ft	
515	VXR	--	ft	Projectile Velocity at Nominal Range $V_x(x_N)$ $V_y(x_N)$ $V_z(x_N)$
516	VYR	--	ft	
517	VZR	--	ft	
518	TX	--	sec	Projectile Time to Reach Nominal Range

variables as being correlated. Thus each variable appearing in Card Group 3 must also appear in Card Group 2 with a TYPE specification of 2, 3, or 4. Table 3-4 defines the format of Group 3 cards and provides pertinent details. The comments denoted by stars command action on the part of the user. The IBM-029 key-punch control cards are presented in Figure 3-3. Group 3 cards may appear in any order.

3.4 Error Source Model Encoding

This section illustrates the encoding of the error source model of Table 1-2. Primary attention is focused on Card Group 2. Maximum instructional benefit will be achieved by the user who works the problem independently and refers to the text simply for verification.

The first task is to identify the code numbers (i.e., trajectory variable names), corresponding to line items in the error source model. Table 3-5 is a complete listing arrived at by cross referencing the error source model, Table 1-2, and the Parameter List, Table 3-3. These are Real World trajectory variables. Each of the Rayleigh/Uniform distributed sources are two-dimensional and, as a result, are described by two trajectory variables.

With the code numbers in hand, Card Group 2 may be constructed by reference to Table 3-2. For instance:

- Muzzle velocity (CODE# = 101) is a Gaussian distributed uncertainty (TYPE = 2). The nominal or mean value is

$$\text{VALUE} = 1.1 \times 10^4 \text{ ft/sec}$$

The standard deviation is

$$\text{STDEV} = (1/3\%) (1.1 \times 10^4) = 3.666 \times 10^1 \text{ ft/sec}$$

and

$$\text{TOL} = 3 * \text{STDEV} = 1.1 \times 10^2 \text{ ft/sec}$$

- Projectile reference area (CODE# = 122) is a uniformly distributed uncertainty (TYPE = 3) with a nominal or mean value of

$$\text{VALUE} = 3.068 \times 10^{-3} \text{ ft}^2$$

Table 3-4 Optional Input Processor Control Cards (Card Group 3)

COLUMN(S)	VARIABLE NAME	DESCRIPTION	COMMENTS
3-5	CØDE#1	Code number for first variable of correlated pair.	• See Table 3-3 for definitions. • Group 3 cards may appear in any order. • Integer variable ★ Negative value (e.g. "-1") required to indicate end of Card Group 3. • CØDE#1 is an address in the ØE array.
8-10	CØDE#2	Code number for second variable of correlated pair. Of necessity CØDE#2 ≠ CØDE#1	• See Table 3-3 for definitions. • Integer variable • CØDE#2 is an address in the ØE array.
12-23	RHØ	The correlation coefficient between the variables defined by CØDE#1 and CØDE#2. Of necessity $-1 \leq RHØ \leq 1$	• Floating point variable. • See Appendix A.3.3 for discussion. • The covariance matrix, Eq. (A.4-15), formed from the correlation coefficients is positive definite as theoretically required.

First Card

[illegible]

Figure 3-3 IBM-029 Keypunch Control Cards for Card Group 3

Table 3-5 Error Source Model Code Numbers

ERROR SOURCE	STATISTICAL DISTRIBUTIONS	CODE NUMBER(S)
• INITIAL CONDITIONS • VELOCITY VECTOR - MAGNITUDE - ORIENTATION • INERTIAL ORIENTATION - ATTITUDE - ATTITUDE RATE	GAUSSIAN RAYLEIGH/UNIFORM* RAYLEIGH/UNIFORM* RAYLEIGH/UNIFORM*	101 120 & 121 116 & 117 118 & 119
• PHYSICAL CHARACTERISTICS • WEIGHT • MOMENTS OF INERTIA - AXIAL - PITCH • PHYSICAL DIMENSIONS - REFERENCE AREA - LENGTH - BASE DIAMETER	GAUSSIAN GAUSSIAN GAUSSIAN UNIFORM UNIFORM UNIFORM	125 126 127 122 123 124
• AERODYNAMIC CHARACTERISTICS • STATIC COEFFICIENTS - DRAG VARIATION EFFECTS - VELOCITY** - ANGLE OF ATTACK - NORMAL FORCE • DYNAMIC COEFFICIENTS - PITCH DAMPING - MAGNUS MOMENT • SPIN RATE • TRIM ANGLE OF ATTACK • STATIC MARGIN	GAUSSIAN GAUSSIAN GAUSSIAN GAUSSIAN GAUSSIAN GAUSSIAN RAYLEIGH/UNIFORM* GAUSSIAN	105 & 107 130 110 112 113 128 114 & 115 111
• ATMOSPHERIC EFFECTS • CONSTANT WINDS • DENSITY VARIATIONS	RAYLEIGH/UNIFORM* GAUSSIAN	102 & 103 104

*Denotes a Rayleigh distribution of magnitude with a uniform 360° distribution in orientation.

**Drag variation with velocity closely approximated by $C_{X_0} = C_{X_{\infty}} + K_D/V^2$ which is fit to two data points $(C_{X_1}, V_1) = (0.03585, 16.740)$ and $(C_{X_2}, V_2) = (0.11951, 3906.)$. Uncertainty in C_{X_1} and C_{X_2} .

The standard deviation is

$$STDEV = (2/3\%)(3.068 \times 10^{-3}) = 2.045 \times 10^{-5} \text{ ft}^2$$

and

$$TOL = \sqrt{3} * STDEV = 3.543 \times 10^{-5} \text{ ft}^2$$

- Constant winds (Code Numbers 102 and 103) are Rayleigh distributed in magnitude and iniformly distributed in direction. This type of distribution is modeled as two independent Gaussian random variables (TYPE = 2). There is no average wind so

$$VALUE = 0.0$$

As discussed in Appendix A.4.4, the equivalent Gaussian standard deviation is given by

$$STDEV = \frac{\bar{J}}{1.2533}$$

where $\bar{J}$ is the mean magnitude given in the mean value column of the error source model, so

$$STDEV = \frac{11}{1.2533} = 8.777 \text{ ft/sec}$$

and

$$TOL = 3 * STDEV = 2.633 \times 10 \text{ ft/sec}$$

These and the other error source Group 2 cards are shown in Figure 3-4.

The error source model of Table 1-2 presumed Fire Control made no simplifying assumptions in the prediction of the projectile trajectory. Thus, since Fire Control would always make use of the known nominal values, the first Group 2 card in Figure 3-4 sets a system control (CØDE# = 201) to initialize the Fire Control trajectory parameters at the nominal values of the Real World trajectory by declaring it a TYPE = 5 equal to 1. This considerably simplifies the input. Even though CØDE# = 201 denotes an integer variable, VALUE is input as a floating point number. Nominal range (CØDE# = 109) is entered as a TYPE = 5 constant value, as shown in Figure 3-4. The second order drag effect variable (CØDE# = 130) appears on two Group 2 cards. The latter prevails. The dependent variables (TYPE = 7), force

1	7	0	0	0	0	11	15	15	15	112001 ← Card Group 1
201	5	1.0000000000	0	0	0	0	0	0	0	
109	5	1.0000000000	0	0	0	0	0	0	0	
130	5	1.0000000000	0	0	0	0	0	0	0	
101	2	1.1000000000	0	0	0	0	0	0	0	
120	2	0.0000000000	0	0	0	0	0	0	0	
121	2	0.0000000000	0	0	0	0	0	0	0	
116	2	0.0000000000	0	0	0	0	0	0	0	
117	2	0.0000000000	0	0	0	0	0	0	0	
118	2	0.0000000000	0	0	0	0	0	0	0	
119	2	0.0000000000	0	0	0	0	0	0	0	
125	2	1.1000000000	0	0	0	0	0	0	0	
126	2	1.2340000000	0	0	0	0	0	0	0	
127	2	2.1100000000	0	0	0	0	0	0	0	
122	3	3.0000000000	0	0	0	0	0	0	0	
123	3	3.1000000000	0	0	0	0	0	0	0	
124	3	0.2000000000	0	0	0	0	0	0	0	
125	2	3.5000000000	0	0	0	0	0	0	0	
127	2	1.1000000000	0	0	0	0	0	0	0	
130	2	1.0000000000	0	0	0	0	0	0	0	
110	2	1.9767000000	0	0	0	0	0	0	0	
112	2	-7.5000000000	0	0	0	0	0	0	0	
128	2	4.0000000000	0	0	0	0	0	0	0	
114	2	0.0000000000	0	0	0	0	0	0	0	
115	2	0.0000000000	0	0	0	0	0	0	0	
111	2	0.2000000000	0	0	0	0	0	0	0	
102	2	0.0000000000	0	0	0	0	0	0	0	
103	2	0.0000000000	0	0	0	0	0	0	0	
104	2	2.3780000000	0	0	0	0	0	0	0	
306	7									
307	7									
308	7									
-1										
101	125	-0.2000000000								
-1										

Card Group 2

} Card Group 3

Figure 3-4 Encoded Error Source Model

the statistics of dispersion at nominal time to be calculated and printed. Card Group 2 is closed by the required card with a negative code number.

Card Group 2 is now complete. The basic error source model has been encoded and attention turns to selecting various HITS options. The Group 1 card is constructed by referring to Table 3-1. An Analytical Statistical mode calculation is requested by the Group 1 card in Figure 3-4. Card Group 3 calls for muzzle velocity (CODE#1 = 101) and projectile weight (CODE#2 = 125) variations to be treated as negatively correlated. The Group 3 card was constructed in accord with Table 3-4. Card Group 3 concludes with the mandatory CODE#1 = -1 card.

Exercising HITS with the case input of Figure 3-4 would evaluate the total dispersion at nominal time due to all twenty error sources acting in consort. If it were desired to evaluate the dispersion due to a single error source, the case input would be generated by (1) duplicating the cards shown in Figure 3-4, and (2) discarding the Group 2 cards pertaining to other error sources. Muzzle velocity and projectile weight effects would be treated as a single source, since they are declared correlated. An error budget similar to Table 1-3 could be constructed by repeating this process for each error source. The resulting case inputs could be stacked and submitted simultaneously.

4.0 OUTPUT INTERPRETATION

This chapter presents four example problems which demonstrate the basic modes of operation of the HITS code. The user is encouraged to reproduce these results to verify that the code has been properly installed and to gain familiarity.

4.1 Single Trajectory Mode

This problem demonstrates the evaluation of single-shot dispersion using the HITS code. This mode is of interest to projectile designers and exterior ballisticians.

Figure 4-1 lists the input cards. The first card is the Group 1 card defined by Table 3-1. All other cards belong to Card Group 2 discussed in Table 3-2. Only TYPE = 5 and TYPE = 7 variables may appear in Single Trajectory mode simulations.

The results of the HITS simulation are shown in Figure 4-2. Frame 'a' is the cover sheet which identifies the code and separates the results of stacked cases. Frames 'b' through 'f' verify the inputs. Frame 'b' is the transcription of the Group 1 card. Frame 'c' lists the Card Group 2 data. Frames 'd', 'e', and 'f' are cross reference indices linking variable names and code numbers. The objective is to minimize the need to refer to Table 3-3 for variable definitions. Frame 'd' gives the FORTRAN names and code numbers of the dependent variables (TYPE = 7 and TYPE = 8). Frame 'e' gives the same information for the deterministic constants whose values have been changed from the presets (TYPE = 5 variables). All variables not changed by the input sequence are listed in Frame 'f'.

The results of calculations presented in Figure 4-2g and h. This is the information passed across the Trajectory Module interface. This level of information may be requested at any time by setting ISPRNT = 1 in Card Group 1 (see Table 3-1), although it can result in excessive output. The formats of Frames 'g' and 'h' are identical. At the top are the Fire Control parameters. These are followed by the Real World parameters, System Controls, and Computed Quantities. Table 3-3 gives the definitions of all quantities. Frame 'g' summarizes the Fire Control calculation to establish the coordinates of the nominal aim point and the nominal time-to-nominal range. The Fire Control summary is distinguished from the Real World summary by the value of the IFC System Control (IFC = 1 → Fire Control, IFC = 0 → Real World). Frame 'h' is the Real World trajectory. Values presented under the Computed Quantities heading quantify dispersion.


```

1  0  1
206 5 0.0000000000
105 5.035853976500
107 5.143415906100
111 5.05
120 5-.0001
126 5.5381000000-6
127 5.7866000000-5
201 51.0
109 52085.2
116 510.0
318 7
319 7
320 7
321 7
306 7
307 7
308 7
309 7
310 7
-1

```

Figure 4-1 Single Trajectory Mode Check Problem Input

*** SIMULATION INPUT SUMMARY ***

SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CONTROLS

IV =	1	IGPRT =	0	ISPRNT =	1
MONTE CARLO CONTROLS					
MCOPT =	0	MCALC =	0	NCELL =	0
IRJ1 =	0	IRJ2 =	0	MCPRT =	0
IOPLT =	0	ISPLT =	0	NTRIAL =	0
				IRANNO =	0

Figure 4-2b Single Trajectory Mode Check Problem Output

*** INPUT VARIABLE SPECIFICATIONS ***

CODE NUMBER	VARIABLE TYPE	NOMINAL VALUE	TOLERANCE	STANDARD DEVIATION	SUBSEQUENT POINTS
206	5	0.0	0.0	0.0	0
105	5	3.58540-02	0.0	0.0	0
107	5	1.43420-01	0.0	0.0	0
111	5	5.00000-02	0.0	0.0	0
120	5	-1.00000-04	0.0	0.0	0
126	5	5.38100-07	0.0	0.0	0
127	5	7.86600-06	0.0	0.0	0
201	5	1.00000 00	0.0	0.0	0
109	5	2.00520 03	0.0	0.0	0
116	5	1.00000 01	0.0	0.0	0
318	7	0.0	0.0	0.0	0
319	7	0.0	0.0	0.0	0
320	7	0.0	0.0	0.0	0
321	7	0.0	0.0	0.0	0
305	7	0.0	0.0	0.0	0
307	7	0.0	0.0	0.0	0
308	7	0.0	0.0	0.0	0
309	7	0.0	0.0	0.0	0
310	7	0.0	0.0	0.0	0

Figure 4-2c Single Trajectory Mode Check Problem Output

*** DEPENDENT VARIABLES ***

RADT STATISTICAL VARIABLE	CODE NUMBER 318
RADX STATISTICAL VARIABLE	CODE NUMBER 319
RROT STATISTICAL VARIABLE	CODE NUMBER 320
RQX STATISTICAL VARIABLE	CODE NUMBER 321
ODT STATISTICAL VARIABLE	CODE NUMBER 306
OYT STATISTICAL VARIABLE	CODE NUMBER 307
ODT STATISTICAL VARIABLE	CODE NUMBER 309
OYX STATISTICAL VARIABLE	CODE NUMBER 309
OZX STATISTICAL VARIABLE	CODE NUMBER 310

Figure 4-2d Single Trajectory Mode Check Problem Output

000 VARIABLES RESET BY INPUT SEQUENCE 000

CX1 REAL WCRLD	CODE NUMBER 105
CX2 REAL WCRLD	CODE NUMBER 107
XN REAL WCRLD	CODE NUMBER 109
SM REAL WCRLD	CODE NUMBER 111
THE TO REAL WCRLD	CODE NUMBER 116
GAMO REAL WORLD	CODE NUMBER 120
AIX REAL WCRLD	CODE NUMBER 126
AIV REAL WCRLD	CODE NUMBER 127
IFCRM SYSTEM CONTROL	CODE NUMBER 201
ICNCL SYSTEM CONTROL	CODE NUMBER 206

Figure 4-2e Single Trajectory Mode Check Problem Output

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

Figure 4-2f Single Trajectory Mode Check Problem Output

[illegible]

Figure 4-2q Single Trajectory Mode Check Problem Output

**COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION**

*** PROJECTILE DISPERSION CASE SUMMARY ***

[illegible]

Figure 4-2h Single Trajectory Mode Check Problem Output

4.2 Range Check Mode

This problem demonstrates the evaluation of dispersion for a systematic permutation of projectile characteristics. This mode could be used to define a trajectory as a function of range. The Range Check mode is of interest to projectile designers and exterior ballisticians.

Figure 4-3 illustrates the input deck for a Range Check mode simulation. The first card is the Group 1 card defined by Table 3-1. All other cards belong to Card Group 2 discussed in Table 3-2. Only TYPE = 1, 5, and 7 variables may appear in a Range Check Card Group 2. There must be at least one TYPE = 1 variable followed by the Range Check values. There must be at least one TYPE = 7 card.

The printed output generated by HITS is illustrated in Figure 4-4. Frame 'a' is the cover sheet. Frames 'b' through 'i' document the input. Frames 'b' and 'c' were discussed in Section 4.1. Frame 'd' lists the Range Check (TYPE = 1) variables. Frames 'e' through 'h' were discussed in Section 4.1. As indicated in Frame 'h', the System Control IFCRC is left at its preset value of zero. Thus, the Fire Control parameters remain at their default values for all Range Check combinations and do not vary with each combination. If a trajectory were being generated as a function of range it would be necessary to set IFCRC = 1 (see Table 3-3 for details). Frame 'i' lists the Range Check combinations. Since there are two TYPE = 1 variables with three values each, HITS will evaluate $3 * 3 = 9$ trajectories.

Frames 'j' and 'k' summarize the results of computation. The summary presents the nine combinations in a sequence of blocks with the variable appearing last in Card Group 2 varying most rapidly. The TYPE = 1 independent variables (IND. VAR.) values appear first, immediately followed by the TYPE = 7 dependent variable values. The code numbers are linked to FORTRAN names by the lists presented in Frames 'd' and 'e'.

1	0	0	0		
206	5	0.00000000D00			
125	1				3
0.8000000D-1	1.1000000D-1	1.5000000D-1			
123	1				3
2.5000000D-1	3.1000000D-1	3.5000000D-1			
105	5	0.0358000D0			
107	5	0.1434000D0			
111	5	0.0500000D0			
318	7				
319	7				
320	7				
321	7				
306	7				
307	7				
308	7				
309	7				
310	7				
-1					

Figure 4-3 Range Check Mode Check Problem Input

*** SIMULATION INPUT SUMMARY ***

SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CONTROLS

IV =	1	IOPHNT =	0	ISPENT =	0
MONTE CARLO CONTROLS					
MCOPT =	0	MCALC =	0	NCELL =	0
IRJ1 =	3	IRJ2 =	0	MCPENT =	0
ISPLT =	0	ISOLUT =	0	NTIAL =	0
				TRANNO =	0

Figure 4-4b Range Check Mode Check Problem Output

*** INPUT VARIABLE SPECIFICATIONS ***

CODE NUMBER	VARIABLE TYPE	NOMINAL VALUE	TOLERANCE	STANDARD DEVIATION	SURSEQUENT POINTS
206	5	0.0	0.0	0.0	0
125	1	0.0	0.0	0.0	3
123	1	0.0	0.0	0.0	3
105	5	3.58000-02	0.0	0.0	0
107	5	1.42400-01	0.0	0.0	0
111	5	5.00000-02	0.0	0.0	0
312	7	0.0	0.0	0.0	0
319	7	0.0	0.0	0.0	0
220	7	0.0	0.0	0.0	0
321	7	0.0	0.0	0.0	0
306	7	0.0	0.0	0.0	0
307	7	0.0	0.0	0.0	0
308	7	0.0	0.0	0.0	0
309	7	0.0	0.0	0.0	0
310	7	0.0	0.0	0.0	0

Figure 4-4c Range Check Mode Check Output

*** RANGE-CHECK INDEPENDENT VARIABLES ***

REAL WORLD	CODE NUMBER 125
ELL REAL WORLD	CODE NUMBER 123

Figure 4-4d Range Check Mode Check Problem Output

*** DEPENDENT VARIABLES ***

R	T	STATISTICAL VARIABLE	CODE NUMBER 319
RAOX		STATISTICAL VARIABLE	CODE NUMBER 319
RHOY		STATISTICAL VARIABLE	CODE NUMBER 320
RHOX		STATISTICAL VARIABLE	CODE NUMBER 321
DXT		STATISTICAL VARIABLE	CODE NUMBER 306
DYT		STATISTICAL VARIABLE	CODE NUMBER 307
DZY		STATISTICAL VARIABLE	CODE NUMBER 308
DYX		STATISTICAL VARIABLE	CODE NUMBER 309
DZX		STATISTICAL VARIABLE	CODE NUMBER 310

Figure 4-4e Range Check Mode Check Problem Output

*** VARIABLES RESET BY INPUT SEQUENCE ***

CX1 REAL WORLD	CODE NUMBER 105
CX2 REAL WORLD	CODE NUMBER 107
SM REAL WORLD	CODE NUMBER 111
ICNCL SYSTEM CONTROL	CODE NUMBER 206

Figure 4-4f Range Check Mode Check Problem Output

*** VARIABLES ASSIGNED PRESET VALUES ***

WZ FIRE CONTROL	CODE NUMBER 3
RHO FIRE CONTROL	CODE NUMBER 4
CX1 FIRE CONTROL	CODE NUMBER 5
V1 FIRE CONTROL	CODE NUMBER 6
CX2 FIRE CONTROL	CODE NUMBER 7
V2 FIRE CONTROL	CODE NUMBER 8
CNA FIRE CONTROL	CODE NUMBER 9
SN FIRE CONTROL	CODE NUMBER 10
CM FIRE CONTROL	CODE NUMBER 11
CMFA FIRE CONTROL	CODE NUMBER 12
ATMS FIRE CONTROL	CODE NUMBER 13
THET45 FIRE CONTROL	CODE NUMBER 14
THETO FIRE CONTROL	CODE NUMBER 15
TS10 FIRE CONTROL	CODE NUMBER 16
PS10 FIRE CONTROL	CODE NUMBER 17
PD010 FIRE CONTROL	CODE NUMBER 18
GA0 FIRE CONTROL	CODE NUMBER 19
A70 FIRE CONTROL	CODE NUMBER 20
ELL FIRE CONTROL	CODE NUMBER 21
A FIRE CONTROL	CODE NUMBER 22
D FIRE CONTROL	CODE NUMBER 23
V FIRE CONTROL	CODE NUMBER 24
W FIRE CONTROL	CODE NUMBER 25
ALY FIRE CONTROL	CODE NUMBER 26
P FIRE CONTROL	CODE NUMBER 27
ALPCON FIRE CONTROL	CODE NUMBER 28
CAA FIRE CONTROL	CODE NUMBER 29
V0 REAL WORLD	CODE NUMBER 30
WZ REAL WORLD	CODE NUMBER 101
NHO REAL WORLD	CODE NUMBER 102
V1 REAL WORLD	CODE NUMBER 103
V2 REAL WORLD	CODE NUMBER 104
XN REAL WORLD	CODE NUMBER 105
CNA REAL WORLD	CODE NUMBER 106
CMQ REAL WORLD	CODE NUMBER 107
CMQA REAL WORLD	CODE NUMBER 108
ATMS REAL WORLD	CODE NUMBER 109
BTMS REAL WORLD	CODE NUMBER 110
THETO REAL WORLD	CODE NUMBER 111
PS10 REAL WORLD	CODE NUMBER 112
PD010 REAL WORLD	CODE NUMBER 113
GA0 REAL WORLD	CODE NUMBER 114
A70 REAL WORLD	CODE NUMBER 115
ELL REAL WORLD	CODE NUMBER 116
A REAL WORLD	CODE NUMBER 117
D REAL WORLD	CODE NUMBER 118
V REAL WORLD	CODE NUMBER 119
W REAL WORLD	CODE NUMBER 120
ALY REAL WORLD	CODE NUMBER 121
P REAL WORLD	CODE NUMBER 122
ALPCON REAL WORLD	CODE NUMBER 123
CAA REAL WORLD	CODE NUMBER 124
V0 REAL WORLD	CODE NUMBER 125
WZ REAL WORLD	CODE NUMBER 126
NHO REAL WORLD	CODE NUMBER 127
V1 REAL WORLD	CODE NUMBER 128
V2 REAL WORLD	CODE NUMBER 129
XN REAL WORLD	CODE NUMBER 130
CNA REAL WORLD	CODE NUMBER 131
CMQ REAL WORLD	CODE NUMBER 132
CMQA REAL WORLD	CODE NUMBER 133
ATMS REAL WORLD	CODE NUMBER 134
BTMS REAL WORLD	CODE NUMBER 135

Figure 4-4g Range Check Mode Check Problem Output

IFCRW SYSTEM CONTROL
IFCRC SYSTEM CONTROL
IDUMP SYSTEM CONTROL
ICONV SYSTEM CONTROL

CODE NUMBER 201
CODE NUMBER 202
CODE NUMBER 204
CODE NUMBER 205

Figure 4-4h Range Check Mode Check Problem Output

*** RANGE CHECK VALUES ***

CODE NUMBER	1 VALUE	2 VALUE	3 VALUE	4 VALUE	5 VALUE
125	8.0000-02	1.1000-01	1.5000-01		
123	2.5000-01	3.1000-01	3.5000-01		

Figure 4-4: Range Check Mode Check Problem Output

*** RANGE CHECK INDEPENDENT VARIABLE COMBINATIONS ***

IND. VAR. OE(125)= 8.00000-07 OE(123)= 2.50000-01

OE(318) OE(319) OE(320) OE(321) OE(306) OE(307) OE(308) OE(309) OE(310)
1.06880 03 0.0 1.06880 03 0.0 -1.06880 03 0.0 0.0 0.0 0.0

IND. VAR. OE(125)= 8.00000-02 OE(123)= 3.10000-01

OE(318) OE(319) OE(320) OE(321) OE(306) OE(307) OE(308) OE(309) OE(310)
1.06880 03 0.0 1.06880 03 0.0 -1.06880 03 0.0 0.0 0.0

IND. VAR. OE(125)= 8.00000-02 OE(123)= 3.50000-01

OE(318) OE(319) OE(320) OE(321) OE(306) OE(307) OE(308) OE(309) OE(310)
1.06880 03 0.0 1.06880 03 0.0 -1.06880 03 0.0 0.0 0.0

IND. VAR. OE(125)= 1.10000-01 OE(123)= 2.50000-01

OE(318) OE(319) OE(320) OE(321) OE(306) OE(307) OE(308) OE(309) OE(310)
1.62250 02 0.0 1.62250 02 0.0 -1.62250 02 0.0 0.0 0.0

IND. VAR. OE(125)= 1.10000-01 OE(123)= 3.10000-01

OE(318) OE(319) OE(320) OE(321) OE(306) OE(307) OE(308) OE(309) OE(310)
1.62250 02 0.0 1.62250 02 0.0 -1.62250 02 0.0 0.0 0.0

IND. VAR. OE(125)= 1.10000-01 OE(123)= 3.50000-01

OE(318) OE(319) OE(320) OE(321) OE(306) OE(307) OE(308) OE(309) OE(310)
1.62250 02 0.0 1.62250 02 0.0 -1.62250 02 0.0 0.0 0.0

IND. VAR. OE(125)= 1.50000-01 OE(123)= 2.50000-01

OE(318) OE(319) OE(320) OE(321) OE(306) OE(307) OE(308) OE(309) OE(310)
1.62250 02 0.0 1.62250 02 0.0 -1.62250 02 0.0 0.0 0.0

Figure 4-4j Range Check Mode Check Problem Output

```

5.5606D 02 0.0      5.5606D 02 0.0      5.5606D 02 0.0      0.0      0.0
IND. VAR. OE( 125)= 1.5000D-01 OE( 123)= 3.1000D-01
OE( 319) OE( 319) OE( 320) OE( 321) OE( 306) OE( 307) OE( 308) OE( 309) OE( 310)
5.5606D 02 0.0      5.5606D 02 0.0      5.5606D 02 0.0      0.0      0.0      0.0
IND. VAR. OE( 125)= 1.5000D-01 OE( 123)= 3.1000D-01
OE( 318) OE( 319) OE( 320) OE( 321) OE( 306) OE( 307) OE( 308) OE( 309) OE( 310)
5.5606D 02 0.0      5.5606D 02 0.0      5.5606D 02 0.0      0.0      0.0      0.0

```

Figure 4-4k Range Check Mode Check Problem Output

4.3 Analytical Statistical Mode

This problem demonstrates the analytical evaluation of dispersion statistics. This mode is of interest to projectile designers and those concerned with the implications of projectile dispersion to overall weapon system effectiveness.

Figure 4-5 illustrates the input deck for an Analytical Statistical mode simulation. The first card is the Group 1 card defined by Table 3-1. All other cards belong to Group 2 discussed in Table 3-2. Only TYPE = 2, 3, 4, 5, and 7 cards may appear in an Analytical Statistical mode simulation. There must be at least one TYPE = 2, 3, or 4 and one TYPE = 7 variable. Figure 4-5 illustrates the TYPE = 4 variable input format. The additional data defining the probability density function must immediately follow the TYPE = 4 Group 2 card.

Figure 4-6 illustrates the Analytical Statistical mode printed output. Frames 'a' through 'h' have been described previously. Frames 'i' and 'j' list the TYPE = 4 probability density functions. Frames 'k' and 'l' give the Fire Control (IFC = 1) and Real World (IFC = 0) solutions for the nominal conditions. Frame 'm' is a self-explanatory warning which appears only when code numbers 320 and 321 are declared TYPE = 7.

Frames 'n' and 'o' of Figure 4-6 present the nominal case independent and dependent variable values, respectively. Whenever the warning doesn't appear, Frame 'o' is followed by the LIMITING CASES output illustrated in Figure 4-6, Frame 'oo', presented at the conclusion of Figure 4-6. As shown in Frame 'oo', each independent variable is incrementally varied by the TOLERANCE amount about the nominal given by VALUE. TOLERANCE is added to get the first set of dependent variable values and subtracted to get the second. Return to the example problem at Frame 'p'. These are the values of the partial derivatives of the dependent variables with respect to the independent. Independent variable code numbers are listed on the left. Dependent variable code numbers are listed above. 1ST and 2ND indicate the first and second unmixed partial derivatives. The second order mixed partial derivatives are printed next as illustrated in Frame 'q'. These are the second order partial derivatives of the dependent variables listed above with respect to the independent variables listed to the left. The partial derivatives define the sensitivity coefficients customarily listed in an error budget.

Frame 'r' of Figure 4-6 presents the dispersion statistics. This is the culmination of the Analytical Statistical mode. The heading summarizes the level of the calculation as determined by the IØPRNT variable in Card Group 1. The mean values, variances, and standard deviations of the dependent variables are printed. Standard deviations are printed a second time with greater numerical precision.

Had any of the independent variables been declared correlated by Card Group 3, Frame 'r' would have been printed assuming no correlation. The following two frames would document the effect of the correlation specified by Card Group 3. The dispersion would be presented in the format of Frame 'r'. Printing the dispersion statistics with and without correlation make it possible to determine the size of the effect from a single simulation.


```

1  4  0
206 5 0.0000000000
116 2 0.000000000 3.000000000 1.000000000
117 2 0.000000000 3.000000000 1.000000000
125 4
0.004430000 0.017520000 0.053980000 0.129500000 0.241900000
0.352030000 0.398900000 0.352030000 0.241900000 0.129500000
0.053980000 0.017520000 0.004430000
15.0000000-2 15.5000000-2 16.0000000-2 16.5000000-2 17.0000000-2
17.5000000-2 18.0000000-2 18.5000000-2 19.0000000-2 19.5000000-2
20.0000000-2 20.5000000-2 21.0000000-2
123 4
0.004430000 0.017520000 0.053980000 0.129500000 0.241900000
0.352030000 0.398900000 0.352030000 0.241900000 0.129500000
0.053980000 0.017520000 0.004430000
33.0000000-2 33.5000000-2 34.0000000-2 34.5000000-2 35.0000000-2
35.5000000-2 36.0000000-2 36.5000000-2 37.0000000-2 37.5000000-2
38.0000000-2 38.5000000-2 39.0000000-2
105 5 0.035800000
107 5 0.143400000
111 5 0.050000000
320 7
321 7
306 7
307 7
308 7
309 7
310 7
-1

```

Figure 4-5 Analytical Statistical Mode Check Problem Input

```

33333 000 00000 33333
 333 0 0 0 333
33333 000 00000 33333

```

ED 437
RESEARCH DIRECTORATE
GENERAL THOMAS J. RODMAN LABORATORY
RDC: 15-AND ARSENAL
ROCK ISLAND, ILLINOIS 61201

EFFECTIVE DATE - JANUARY 1976

Figure 4-6a Analytical Statistical Mode Check Problem Output

*** SIMULATION INPUT SUMMARY ***

SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CONTROLS

IV = 1 IOPRINT = 4 ISPRINT = 0

MONTI CARLO CONTROLS

MCOPT = 0 MCALC = 0 MCELL = 0 MTRIAL = 0
 IRJ1 = 0 IRJ2 = 0 MCPMPT = 0 IRANNO = 0
 IOPLOT = 0 ISPLIT = 0

Figure 4-6b Analytical Statistical Mode Check Problem Output

*** INPUT VARIABLE SPECIFICATIONS ***

CODE NUMBER	VARIABLE TYPE	NOMINAL VALUE	TOLERANCE	STANDARD DEVIATION	SUBSEQUENT POINTS
206	5	0.0	0.0	0.0	0
116	2	0.0	3.00000 00	1.00000 00	0
117	2	0.0	3.00000 00	1.00000 00	0
125	4	0.0	0.0	0.0	13
123	4	0.0	0.0	0.0	13
125	5	3.50000-02	0.0	0.0	0
107	5	1.43400-01	0.0	0.0	0
111	5	5.00000-02	0.0	0.0	0
120	7	0.0	0.0	0.0	0
321	7	0.0	0.0	0.0	0
306	7	0.0	0.0	0.0	0
307	7	0.0	0.0	0.0	0
328	7	0.0	0.0	0.0	0
309	7	0.0	0.0	0.0	0
310	7	0.0	0.0	0.0	0

Figure 4-6c Analytical Statistical Mode Check Problem Output

*** STOCHASTIC INDEPENDENT VARIABLES ***

THEYO REAL WORLD	CODE NUMBER 116
PSIO REAL WORLD	CODE NUMBER 117
ELL REAL WORLD	CODE NUMBER 123
W REAL WORLD	CODE NUMBER 125

Figure 4-6d Analytical Statistical Mode Check Problem Output

*** DEPENDENT VARIABLES ***

RMT STATISTICAL VARIABLE	CODE NUMBER 320
RDX STATISTICAL VARIABLE	CODE NUMBER 321
DAT STATISTICAL VARIABLE	CODE NUMBER 304
DYT STATISTICAL VARIABLE	CODE NUMBER 307
DZT STATISTICAL VARIABLE	CODE NUMBER 308
DYX STATISTICAL VARIABLE	CODE NUMBER 309
DZX STATISTICAL VARIABLE	CODE NUMBER 310

Figure 4-6e Analytical Statistical Mode Check Problem Output

*** VARIABLES RESET BY INPUT SEQUENCE ***

CX1 REAL WORLD	CODE NUMBER 105
CX2 REAL WORLD	CODE NUMBER 107
SM REAL WORLD	CODE NUMBER 111
ICNCL SYSTEM CONTROL	CODE NUMBER 204

Figure 4-6f Analytical Statistical Mode Check Problem Output

*** VARIABLES ASSIGNED PRESET VALUES ***

CX1	FIRE CONTROL	CODE NUMBER	5
V1	FIRE CONTROL	CODE NUMBER	6
CX2	FIRE CONTROL	CODE NUMBER	7
V2	FIRE CONTROL	CODE NUMBER	8
CXN	FIRE CONTROL	CODE NUMBER	9
SM	FIRE CONTROL	CODE NUMBER	10
CMO	FIRE CONTROL	CODE NUMBER	11
CMFA	FIRE CONTROL	CODE NUMBER	12
ATMS	FIRE CONTROL	CODE NUMBER	13
BTMS	FIRE CONTROL	CODE NUMBER	14
THETO	FIRE CONTROL	CODE NUMBER	15
TSIO	FIRE CONTROL	CODE NUMBER	16
PDOTO	FIRE CONTROL	CODE NUMBER	17
GANO	FIRE CONTROL	CODE NUMBER	18
A70	FIRE CONTROL	CODE NUMBER	19
ELL	FIRE CONTROL	CODE NUMBER	20
A	FIRE CONTROL	CODE NUMBER	21
D	FIRE CONTROL	CODE NUMBER	22
W	FIRE CONTROL	CODE NUMBER	23
AIX	FIRE CONTROL	CODE NUMBER	24
AIY	FIRE CONTROL	CODE NUMBER	25
P	FIRE CONTROL	CODE NUMBER	26
ALPCON	FIRE CONTROL	CODE NUMBER	27
CAA	FIRE CONTROL	CODE NUMBER	28
VO	REAL WORLD	CODE NUMBER	29
W	REAL WORLD	CODE NUMBER	30
RHO	REAL WORLD	CODE NUMBER	31
V1	REAL WORLD	CODE NUMBER	32
V2	REAL WORLD	CODE NUMBER	33
YN	REAL WORLD	CODE NUMBER	34
CNA	REAL WORLD	CODE NUMBER	35
CMQ	REAL WORLD	CODE NUMBER	36
CMFA	REAL WORLD	CODE NUMBER	37
ATMS	REAL WORLD	CODE NUMBER	38
BTMS	REAL WORLD	CODE NUMBER	39
V0	FIRE CONTROL	CODE NUMBER	40
V1	FIRE CONTROL	CODE NUMBER	41
PDOTO	REAL WORLD	CODE NUMBER	42
GANO	REAL WORLD	CODE NUMBER	43
A70	REAL WORLD	CODE NUMBER	44
V2	FIRE CONTROL	CODE NUMBER	45
W	REAL WORLD	CODE NUMBER	46
RHO	FIRE CONTROL	CODE NUMBER	47
AIY	REAL WORLD	CODE NUMBER	48
AIY	REAL WORLD	CODE NUMBER	49
D	REAL WORLD	CODE NUMBER	50
ALPCON	REAL WORLD	CODE NUMBER	51
YIART	REAL WORLD	CODE NUMBER	52
ZIART	REAL WORLD	CODE NUMBER	53
YIART	REAL WORLD	CODE NUMBER	54
ZIART	REAL WORLD	CODE NUMBER	55
IFCRM	SYSTEM CONTROL	CODE NUMBER	56
IFCRM	SYSTEM CONTROL	CODE NUMBER	57
IFCRM	SYSTEM CONTROL	CODE NUMBER	58
IFCRM	SYSTEM CONTROL	CODE NUMBER	59
IFCRM	SYSTEM CONTROL	CODE NUMBER	60
IFCRM	SYSTEM CONTROL	CODE NUMBER	61
IFCRM	SYSTEM CONTROL	CODE NUMBER	62
IFCRM	SYSTEM CONTROL	CODE NUMBER	63
IFCRM	SYSTEM CONTROL	CODE NUMBER	64
IFCRM	SYSTEM CONTROL	CODE NUMBER	65
IFCRM	SYSTEM CONTROL	CODE NUMBER	66
IFCRM	SYSTEM CONTROL	CODE NUMBER	67
IFCRM	SYSTEM CONTROL	CODE NUMBER	68
IFCRM	SYSTEM CONTROL	CODE NUMBER	69
IFCRM	SYSTEM CONTROL	CODE NUMBER	70
IFCRM	SYSTEM CONTROL	CODE NUMBER	71
IFCRM	SYSTEM CONTROL	CODE NUMBER	72
IFCRM	SYSTEM CONTROL	CODE NUMBER	73
IFCRM	SYSTEM CONTROL	CODE NUMBER	74
IFCRM	SYSTEM CONTROL	CODE NUMBER	75
IFCRM	SYSTEM CONTROL	CODE NUMBER	76
IFCRM	SYSTEM CONTROL	CODE NUMBER	77
IFCRM	SYSTEM CONTROL	CODE NUMBER	78
IFCRM	SYSTEM CONTROL	CODE NUMBER	79
IFCRM	SYSTEM CONTROL	CODE NUMBER	80
IFCRM	SYSTEM CONTROL	CODE NUMBER	81
IFCRM	SYSTEM CONTROL	CODE NUMBER	82
IFCRM	SYSTEM CONTROL	CODE NUMBER	83
IFCRM	SYSTEM CONTROL	CODE NUMBER	84
IFCRM	SYSTEM CONTROL	CODE NUMBER	85
IFCRM	SYSTEM CONTROL	CODE NUMBER	86
IFCRM	SYSTEM CONTROL	CODE NUMBER	87
IFCRM	SYSTEM CONTROL	CODE NUMBER	88
IFCRM	SYSTEM CONTROL	CODE NUMBER	89
IFCRM	SYSTEM CONTROL	CODE NUMBER	90
IFCRM	SYSTEM CONTROL	CODE NUMBER	91
IFCRM	SYSTEM CONTROL	CODE NUMBER	92
IFCRM	SYSTEM CONTROL	CODE NUMBER	93
IFCRM	SYSTEM CONTROL	CODE NUMBER	94
IFCRM	SYSTEM CONTROL	CODE NUMBER	95
IFCRM	SYSTEM CONTROL	CODE NUMBER	96
IFCRM	SYSTEM CONTROL	CODE NUMBER	97
IFCRM	SYSTEM CONTROL	CODE NUMBER	98
IFCRM	SYSTEM CONTROL	CODE NUMBER	99
IFCRM	SYSTEM CONTROL	CODE NUMBER	100

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

Figure 4-69 Analytical Statistical Mode Check Problem Output

DUMP SYSTEM CONTROL
ICONV SYSTEM CONTROL

CODE NUMBER 204
CODE NUMBER 205

Figure 4-6h Analytical Statistical Mode Check Problem Output

*** CODE NUMBER 123 PROBABILITY DENSITY FUNCTION ***

POINT NUMBER	COORDINATE VALUE	PROBABILITY DENSITY
1	3.30000-01	4.43000-01
2	3.35000-01	1.75200-02
3	3.40000-01	5.39800-01
4	3.45000-01	1.24900-01
5	3.50000-01	2.41000-01
6	3.55000-01	3.52900-01
7	3.60000-01	3.52300-01
8	3.65000-01	2.51000-01
9	3.70000-01	2.20500-01
10	3.75000-01	5.59800-02
11	3.80000-01	1.75200-02
12	3.85000-01	4.43000-03
13	3.90000-01	

COPY AVAILABLE TO DGC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

Figure 4-6i Analytical Statistical Mode Check Problem Output

*** CODE NUMBER 125 PROBABILITY DENSITY FUNCTION ***

POINT NUMBER	COORDINATE VALUE	PROBABILITY DENSITY
1	1.50000-01	4.43000-03
2	1.55000-01	4.75000-02
3	1.60000-01	1.38000-01
4	1.65000-01	2.41000-01
5	1.70000-01	3.52000-01
6	1.75000-01	3.95000-01
7	1.80000-01	3.52000-01
8	1.85000-01	2.41000-01
9	1.90000-01	1.38000-01
10	1.95000-01	4.75000-02
11	2.00000-01	1.75000-02
12	2.05000-01	5.39000-02
13	2.10000-01	4.43000-03

COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION

Figure 4-6j Analytical Statistical Mode Check Problem Output

*** PROJECTILE DISPERSION CASE SUMMARY ***

Figure 4-6k Analytical Statistical Mode Check Problem Output

THE DEPARTMENT OF THE ARMY
WASHINGTON, D. C. 20315

Figure 4-61 Analytical Statistical Mode Check Problem Output

*** NOTICE ***

THE SPECIFICATION OF EITHER RHOY OR RHOX (IE
VARIABLES 320 OR 321) OR BOTH AS OUTPUT VARIABLES
NECESSITATES UP-DATING THE NOMINAL VALUES, DERIVATIVES,
MEANS, AND VARIANCES, THE LIMITING CASES FOR AND PARTIAL
DERIVATIVES OF RHOY AND RHOX ARE MEANINGLESS. THUS THE
FULL PROJECTILE DISPERSION CODE SHOULD BE UTILIZED IN
MORTE CARLO SIMULATIONS (MCALC = 0) TO OBTAIN
MEANINGFUL VALUES OF RHOY AND/OR RHOX. IF A TAYLOR
SERIES APPROXIMATION OF THE PROJECTILE DISPERSION CODE
(MCALC = 1, 2, OR 3) IS USED IN MONTE CARLO
SIMULATIONS THE RHOY AND RHOX RESULTS ARE MEANINGLESS.

**COPY AVAILABLE TO DDC DOES NOT
PERMIT FULLY LEGIBLE PRODUCTION**

Figure 4-6m Analytical Statistical Mode Check Problem Output

NOMINAL CASE, INDEPENDENT VARIABLES ...

OE(116)= 0.0 OE(117)= 0.0 OE(123)= 3.6000D-01 OE(125)= 1.8000D-01

Figure 4-6n Analytical Statistical Mode Check Problem Output

NOMINAL CASE, DEPENDENT VARIABLES ...

UE(320) OE(321) OE(306) OE(307) OE(304) OE(300) OE(312)

5.94770 00 1.66300-16 9.10230 02 0.0 0.0 0.0 0.0

Figure 4-60 Analytical Statistical Mode Check Problem Output

DERIVATIVES IND. VAR.	DEPENDENT VARIABLE									
	320	221	306	307	308	309	310			
115 1ST 0.0	0.0	0.0	0.0	1.0464D-01	-8.4757D-02	9.6250D-02	-7.7717D-02			
2ND 0.0	0.0	0.0	-9.9697D-01	4.0091D-17	0.0	3.7007D-17	0.0			
117 1ST 0.0	0.0	0.0	0.0	8.4757D-02	1.0464D-01	7.7717D-02	9.6250D-02			
2ND 0.0	0.0	0.0	-9.9697D-01	3.0440D-18	3.5925D-15	4.5259D-15	3.2998D-16			
123 1ST 0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0			
2ND 0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0			
125 1ST 0.0	0.0	0.0	1.0299D 04	0.0	0.0	0.0	0.0			
2ND 0.0	0.0	0.0	-1.0024D 05	0.0	0.0	0.0	0.0			

Figure 4-6p Analytical Statistical Mode Check Problem Output

SECOND ORDER MIXED PARTIALS

100% VAR.

DEPENDENT VARIABLES

	320	321	305	307	308	309	310
115 117	0.0	0.0	0.0	1.04400-16	-1.85040-17	1.03700-16	-1.54200-17
115 123	0.0	0.0	0.0	-2.91190-01	2.35440-01	-2.67770-01	2.15990-01
115 125	0.0	0.0	0.0	-4.90320-01	3.93940-01	-5.36310-01	4.34260-01
117 123	0.0	0.0	0.0	-2.35670-01	-2.91190-01	-2.15490-01	-2.67770-01
117 125	0.0	0.0	0.0	-3.93940-01	-4.90320-01	-4.34260-01	-5.36310-01
123 125	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Figure 4-6q Analytical Statistical Mode Check Problem Output

2ND ORDER MEANS AND 1ST ORDER VARIANCES

		DEPENDENT VARIABLES				
		329	321	306	307	309 310
MEANS	5.4589D 01	1.5505D-01	9.0428D 02	2.1588D-17	1.7943D-15	2.0817E-17 1.6499D-16
VARIANCES	9.3073D 02	6.5985D-03	1.0477D 04	1.8133D-02	1.5133D-02	1.5394D-02 1.5304D-02
1ST. DEV.	2.3038D 01	8.1945D-02	1.0236D 02	1.3466D-01	1.3466D-01	1.2371D-01 1.2371D-01

STANDARD DEVIATIONS WITH HIGHER PRECISION

2.3338D 01	2.3037507522147550D 01
8.1246D-02	8.1046232145E-02 1.700D-02
1.0235D 02	1.023564642431840D 02
1.3465D-01	1.3465725756933410D-01
1.7456D-01	1.3465720756933750D-01
1.2371D-01	1.2370910123153840D-01
1.2371D-01	1.2370910123153790D-01

Figure 4-6r Analytical Statistical Mode Check Problem Output

LIMITING CASES		DEPENDENT VARIABLE									
INDEPENDENT VARIABLE											
NAME	VALUE	TOLERANCE	DE(106)	DE(207)	DE(308)	DE(309)	DE(310)	DE(311)	DE(312)	DE(313)	
			DE(315)	DE(216)							
123	3.10000-01	1.79000-03	-6.00400-12	0.0	0.0	0.0	0.0	2.72850-12	0.0	0.0	
			0.0	0.0							
			-9.00400-12	0.0	0.0	0.0	0.0	2.72850-12	0.0	0.0	
			0.0	0.0							

NOTE: This frame illustrates only the format of the LIMITING CASE printout.
It was not generated by the input of Figure 4-5.

Figure 4-600 Analytical Statistical Mode Check Problem Output

4.4 Monte Carlo Mode

This problem demonstrates the Monte Carlo evaluation of dispersion statistics and probability distributions. The Monte Carlo mode determines dispersion by performing numerical experiments with the full non-linear trajectory equations rather than the Taylor series approximation used by the Analytical Statistical mode. Thus it is inherently more accurate, although large numbers of experiments would have to be conducted to realize the advantage. The number of experiments impacts cost. This mode is of interest to projectile designers and those interested in projectile dispersion statistics.

Figure 4-7 illustrates the input deck for a Monte Carlo mode simulation. The first card is the Group 1 card defined by Table 3-1. The Group 1 card sets MCØPT = 1, which forces the Statistical Processor into the Monte Carlo mode after using the results of the Analytical Statistical mode to set up the histogram boundaries. With the exception of the first card, all cards shown in Figure 4-7 belong to Group 2. Only TYPE = 2, 3, 4, 5, 7, and 8 cards may appear in a Monte Carlo simulation. There must be a minimum of one TYPE = 2, 3, or 4 and one TYPE = 7 or 8. Figure 4-7 illustrates the use of System Control IFCRW (CØDE# = 201) to initialize the Fire Control parameters at the Real World nominals.

Figure 4-8 illustrates the printed output generated by a Monte Carlo mode simulation. Frames 'a' through 'p' are the results of the initial Analytical Statistical mode calculations. The user is referred to Section 4.3 for a discussion of this output. Subsequent to the Analytical Statistical calculations, INTERNAL RANGE histograms are set up for all TYPE = 7 and TYPE = 8 variables with the cell boundaries determined by the analytically computed mean values and standard deviations. USER RANGE histograms are also set up for all variables with the cell boundaries for the TYPE = 8 variables determined by the information supplied on the Group 2 cards, as described in Table 3-2. At this point the Monte Carlo experiments begin.

Figure 4-8, Frame 'q' and Frame 'r' illustrate the summary information printed for each of the Monte Carlo experiments. This sequence of experiments is generated by using a random number generator to simulate random variations in the projectile parameters consistent with the input statistics. A given random sequence may be repeated by reusing the value of IRANNØ on the Group 1 card (see Table 3-1 for instructions). A different sequence will result from every selection of IRANNØ. The CASE variable in Frame 'q' counts the experiments. The first line,

labeled IND. VAR., are the stochastic independent variable values appearing in the order given by Frame 'd'. The line labeled DEP. VAR. are the corresponding values of the dependent variables arranged in the order of Frame 'e'. Even though only a small number of experiments (100) were conducted, the summary print required ten pages (only two are included here). Thus, the user should consider suppressing the summary print by setting MCPRT = 0 in Card Group 1, whenever large numbers of experiments are conducted. The information in the summary print is used to update the histograms after each experiment but is not stored in any other way.

At the conclusion of the Monte Carlo experiments, the histogram information is printed. In general, there are two sets of histograms. The INTERNAL RANGE histograms are printed first and are usually followed by the USER RANGE histograms. Since the example problem input shown in Figure 4-7 contains no TYPE = 8 variables, the sample output of Figure 4-8 contains only INTERNAL RANGE histograms. There is no need to document the USER RANGE histograms separately because they are constructed in an identical fashion and are subject to the same interpretation. The only difference is in how the cell boundaries are set up, which has already been discussed. The histogram data for the example problem commences in Frame 's' of Figure 4-8. The first line of print defines (1) the total number of random experiments performed (SAMPLE), (2) the number of experiments which were rejected from the INTERNAL RANGE histogram set because at least one of the dependent variable histogram ranges were exceeded (INTERNAL REJECTS) and (3) the number of experiments rejected from the USER RANGE histogram set for the same reason. The latter is zero in the example because USER RANGE histograms were never set up. The histograms making up the INTERNAL RANGE histogram set are presented on a variable-by-variable basis in Frames 's' through 'y'. Independent variable histograms precede the dependent variable histograms. Generally, the block of data describing the variable, consists of (1) a line identifying the variable and statistics computed from the histogram, (2) a second line defining the number of experiments rejected from the set because of range limitations on this variable and Analytical Statistical mode statistics, and (3) multiple lines of print defining the histogram on a cell-by-cell basis. The columns define the cell boundaries, the number of occurrences (FREQ), and the probability density function (DIST. FUNCT.). Although the format of the block is the same for all, there are some differences in interpretation. Table 4-1 rigorously interprets each printed value. Hash marks delimit the extent of the comments contained in the right hand column of Table 4-1.

```

1  6  0      1      0  20  100  50  50  112001
206 5 0.0000000000
116 2 0.00000000+0 3.00000000+0 1.00000000+0
117 2 0.00000000+0 3.00000000+0 1.00000000+0
118 2 0.00000000+0 2.70000000+1 9.00000000+1
119 2 0.00000000+0 2.70000000+1 9.00000000+1
128 5 0.00000000+0
201 5 1.00000000+0
109 5 1.17300000+3
205 5 -8.00000000+0
306 7
307 7
308 7
320 7
317 7
322 7
323 7
324 7
518 7
510 7
-1

```

Figure 4-7 Monte Carlo Mode Check Problem Input

Figure 4-8a Monte Carlo Mode Check Problem Output

SEE SIMULATION INPUT SUMMARY ***

SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CONTROLS

IV = 1 IOPRINT = 5 ISPRINT = 0

MONTÉ CARLO CONTROLS

MCOPT = 1 MCALC = 0 NCELL = 20 NTRIAL = 100
IRJ1 = 50 IRJ2 = 50 MCPRNT = 1 IRANNO = 1200
IOPL0T = 0 ISPL0T = 0

Figure 4-8b. Monte Carlo Mode Check Problem Output

*** INPUT VARIABLE SPECIFICATIONS ***

CODE NUMBER	VARIABLE TYPE	NOMINAL VALUE	TOLERANCE	STANDARD DEVIATION	SUBSEQUENT POINTS
206	5	0.0	0.0	0.0	0
116	2	0.0	3.00000 00	1.00000 00	0
117	2	0.0	3.00000 00	1.00000 00	0
118	2	0.0	2.70000 01	9.00000 01	0
119	2	0.0	2.70000 01	9.00000 01	0
120	5	0.0	0.0	0.0	0
201	5	1.00000 00	0.0	0.0	0
109	5	1.17300 03	0.0	0.0	0
205	5	-6.00000 00	0.0	0.0	0
306	7	0.0	0.0	0.0	0
307	7	0.0	0.0	0.0	0
308	7	0.0	0.0	0.0	0
320	7	0.0	0.0	0.0	0
317	7	0.0	0.0	0.0	0
322	1	0.0	0.0	0.0	0
323		0.0	0.0	0.0	0
324		0.0	0.0	0.0	0
518	7	0.0	0.0	0.0	0
510	2	0.0	0.0	0.0	0

Figure 4-8c Monte Carlo Mode Check Problem Output

*** STOCHASTIC INDEPENDENT VARIABLES ***

THETO REAL WORLD	CODE NUMBER 116
0510 REAL WORLD	CODE NUMBER 117
10010 REAL WORLD	CODE NUMBER 118
10010 REAL WORLD	CODE NUMBER 119

Figure 4-8d Monte Carlo Mode Check Problem Input

*** DEPENDENT VARIABLES ***

DX1 STATISTICAL VARIABLE	CODE NUMBER 306
DY1 STATISTICAL VARIABLE	CODE NUMBER 307
DZ1 STATISTICAL VARIABLE	CODE NUMBER 308
R101 STATISTICAL VARIABLE	CODE NUMBER 320
DY1 STATISTICAL VARIABLE	CODE NUMBER 317
DX1 STATISTICAL VARIABLE	CODE NUMBER 322
DY1 STATISTICAL VARIABLE	CODE NUMBER 323
DZ1 STATISTICAL VARIABLE	CODE NUMBER 324
TX TRAJECTORY VARIABLE	CODE NUMBER 518
VX1 TRAJECTORY VARIABLE	CODE NUMBER 510

Figure 4-8e Monte Carlo Mode Check Problem Output

*** VARIABLES RESET BY INPUT SEQUENCE ***

IN REAL WORLD	CODE NUMBER 109
P REAL WORLD	CODE NUMBER 128
IFCRN SYSTEM CONTROL	CODE NUMBER 201
ICMNV SYSTEM CONTROL	CODE NUMBER 205
ICNCL SYSTEM CONTROL	CODE NUMBER 206

Figure 4-8f Monte Carlo Mode Check Problem Output

*** VARIABLES ASSIGNED PRESET VALUES ***

CX1 FIRE CONTROL	CODE NUMBER	5
V1 FIRE CONTROL	CODE NUMBER	6
CX2 FIRE CONTROL	CODE NUMBER	7
V2 FIRE CONTROL	CODE NUMBER	8
CXN FIRE CONTROL	CODE NUMBER	9
SM FIRE CONTROL	CODE NUMBER	10
CNA FIRE CONTROL	CODE NUMBER	11
CMFA FIRE CONTROL	CODE NUMBER	12
ATRMS FIRE CONTROL	CODE NUMBER	13
BTMS FIRE CONTROL	CODE NUMBER	14
PAETO FIRE CONTROL	CODE NUMBER	15
TDOTO FIRE CONTROL	CODE NUMBER	16
PCMO FIRE CONTROL	CODE NUMBER	17
GAMO FIRE CONTROL	CODE NUMBER	18
A70 FIRE CONTROL	CODE NUMBER	19
ELL FIRE CONTROL	CODE NUMBER	20
V FIRE CONTROL	CODE NUMBER	21
Y FIRE CONTROL	CODE NUMBER	22
W FIRE CONTROL	CODE NUMBER	23
ATX FIRE CONTROL	CODE NUMBER	24
ATY FIRE CONTROL	CODE NUMBER	25
P FIRE CONTROL	CODE NUMBER	26
ALPCON FIRE CONTROL	CODE NUMBER	27
CAA FIRE CONTROL	CODE NUMBER	28
VO REAL WORLD	CODE NUMBER	29
WX REAL WORLD	CODE NUMBER	30
WZ REAL WORLD	CODE NUMBER	101
RHO REAL WORLD	CODE NUMBER	102
CX1 REAL WORLD	CODE NUMBER	103
V1 REAL WORLD	CODE NUMBER	104
CX2 REAL WORLD	CODE NUMBER	105
V2 REAL WORLD	CODE NUMBER	106
CNA REAL WORLD	CODE NUMBER	107
SM REAL WORLD	CODE NUMBER	108
CMQ REAL WORLD	CODE NUMBER	109
CMFA REAL WORLD	CODE NUMBER	110
ATRMS REAL WORLD	CODE NUMBER	111
BTMS REAL WORLD	CODE NUMBER	112
VO FIRE CONTROL	CODE NUMBER	113
WX FIRE CONTROL	CODE NUMBER	114
WZ FIRE CONTROL	CODE NUMBER	115
RHO FIRE CONTROL	CODE NUMBER	1
CX1 FIRE CONTROL	CODE NUMBER	2
V1 FIRE CONTROL	CODE NUMBER	3
CX2 FIRE CONTROL	CODE NUMBER	4
V2 FIRE CONTROL	CODE NUMBER	120
CNA FIRE CONTROL	CODE NUMBER	121
SM FIRE CONTROL	CODE NUMBER	122
CMFA FIRE CONTROL	CODE NUMBER	123
ATRMS FIRE CONTROL	CODE NUMBER	124
BTMS FIRE CONTROL	CODE NUMBER	125
VO FIRE CONTROL	CODE NUMBER	126
WX FIRE CONTROL	CODE NUMBER	127
WZ FIRE CONTROL	CODE NUMBER	128
RHO FIRE CONTROL	CODE NUMBER	129
CX1 FIRE CONTROL	CODE NUMBER	130
V1 FIRE CONTROL	CODE NUMBER	131
CX2 FIRE CONTROL	CODE NUMBER	132
V2 FIRE CONTROL	CODE NUMBER	133
CNA FIRE CONTROL	CODE NUMBER	134
SM FIRE CONTROL	CODE NUMBER	135
CMFA FIRE CONTROL	CODE NUMBER	202

Figure 4-8g Monte Carlo Mode Check Problem Output

DUMP SYSTEM CONTROL

CODE NUMBER 204

Figure 4-8h Monte Carlo Mode Check Problem Output

[illegible]

REAL NCARD		PARAMETERS		PARAMETERS		FT/SEC		SL-CFT	
V1	1.1000D 04	FT/SEC	0.0	0.0	0.0	0.0	0.0	0.0	0.0
V2	1.5540D 02	(NDNE)	1.6740D 04	FT/SEC	1.19510-01	(NDNE)	1.9510-01	(NDNE)	1.9510-01
CN1	1.730D 03	FT/SEC	1.0767D 00	1/RAD	0.2030D-02	100/L	0.2030D-02	100/L	0.2030D-02
CN2	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN3	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN4	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN5	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN6	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN7	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN8	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN9	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN10	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN11	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN12	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN13	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN14	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN15	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN16	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN17	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN18	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN19	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN20	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN21	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN22	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN23	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN24	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN25	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN26	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN27	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN28	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN29	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN30	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN31	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN32	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN33	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN34	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN35	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN36	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN37	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN38	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN39	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN40	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN41	0.0		0.0	DEG	0.0	DEG	0.0	DEG	0.0
CN4									

SYSTEMS CONTACTS			
IFCRN =	IFCRC =	IPC =	IDUMP =
1 (NONE)	0 (NONE)	1 (NONE)	0 (NONE)

[COM] =	-8 (NONE)	[CNCL] =	0 (NONE)	COUNTED QUANTITIES

... XEARTI = 0.0	FT	... YEARTI = 0.0	FT
... ZBARTI = 0.0	FT	... ZBARTI = 0.0	FT

[illegible][illegible]

RHOX = 0.0
 JF = 0.0
 SEC
 DXTI = 0.0
 RADT = 0.0
 FTY
 OYTI = 0.0
 MAUX = 0.0
 FTY
 OZTI = 0.0
 FTY

[illegible][illegible]

SQ-FT	DYDZTI = 0.0	SQ-FT	DVDZXI = 0.0	SQ-FT	DXDYFE = 0.0
SQ-FT	DVDTFE = 0.0	SQ-FT	DVDTXE = 0.0	SQ-FT	

ALPHA =	RAD	ATOT =	SAD	ALTRM =	RAC	ALPHO0 =	RAD/SEC
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

[illegible]

COR =	3.10370-02 (NONE)	CMA =	-6.07870-01 1/RAD	CMTHC =	-2.1207D-05 SEC	CMTHFA =	0.0	SEC
CALC =	9.0000E+00	CALC =	0.0	CALC =	4.2107D-02 (NONE)	CALC =	1.2107D-02 (NONE)	(NONE)

CAY1 =	9.0	RAD
DELTY =	0.2	SEC
CAY2 =	0.0	RAD
FY/SEC	=	0.0
DELT =	0.2	SEC
CYCD =	4821730-0.2	(NOV)
OAS =	4021950-02	(JAN)
OEX =	0.0	----
FY	----	----

CYXG =	0.0	RAD
CZXC =	0.0	RAD
OYDIO =	0.0	FT/SEC
EVEE =	7.60070-07	SO-SEC
SVD =	8.42200-02	
CZDT =	0.0	
ST/SEC		

SOLN =	6.57390-02 (NONE)	SOLN =	0.00000 00 (NCNE)	F =	1.06700-03 1/FY	H =	2.55200 05 FY/SCL
SOLN =	1.75000 00 (NONE)	SOLN =	1.75000 00 (NONE)	F =	1.06700-03 1/FY	H =	2.55200 05 FY/SCL

PSIDJ = 0.0 RAD/SC PHIC = -1.59940 00 RAD
0512 = 0.0 RAD R1 = 0.0 RAD
PHI1 = -1.59940 00 RAD
R2 = 0.0 RAD
PHI2 = -1.59940 00 RAD
R3 = 0.0 RAD

RA = 5.0
RPM = 1.0000 00 (NCNE)
TC = 9.0000-08 SEC
TCI = 9.0000-09 SEC

$\tau_3 =$	$1.0953D-01$	SEC	$\gamma G =$	0.0	SEC	$\gamma P =$	$1.036A D-01$	SEC	$\text{THE} \Delta C =$	0.0	KAD
$C_{U-M} =$	0.0	RAD/SC	$T_N =$	$1.0000-05$	(VNE)	$V_C =$	1.0000	04 FT/SEC	$VCC =$	1.0000	04 FT/SEC

Run	Time	Temp	Pressure	Flow	Conc	Yield	Quality
1	10.0	100	1.0	1.0	1.0	1.0	1.0
2	10.0	100	1.0	1.0	1.0	1.0	1.0
3	10.0	100	1.0	1.0	1.0	1.0	1.0
4	10.0	100	1.0	1.0	1.0	1.0	1.0
5	10.0	100	1.0	1.0	1.0	1.0	1.0
6	10.0	100	1.0	1.0	1.0	1.0	1.0
7	10.0	100	1.0	1.0	1.0	1.0	1.0
8	10.0	100	1.0	1.0	1.0	1.0	1.0
9	10.0	100	1.0	1.0	1.0	1.0	1.0
10	10.0	100	1.0	1.0	1.0	1.0	1.0

Variable	Value	Unit
W1	8.91420	02.1/SEC
W2	8.91420	02.1/SEC
W3	8.91420	02.1/SEC
W4	8.91420	02.1/SEC
W5	8.91420	02.1/SEC
W6	8.91420	02.1/SEC
W7	8.91420	02.1/SEC
W8	8.91420	02.1/SEC
W9	8.91420	02.1/SEC
W10	8.91420	02.1/SEC
W11	8.91420	02.1/SEC
W12	8.91420	02.1/SEC
W13	8.91420	02.1/SEC
W14	8.91420	02.1/SEC
W15	8.91420	02.1/SEC
W16	8.91420	02.1/SEC
W17	8.91420	02.1/SEC
W18	8.91420	02.1/SEC
W19	8.91420	02.1/SEC
W20	8.91420	02.1/SEC
W21	8.91420	02.1/SEC
W22	8.91420	02.1/SEC
W23	8.91420	02.1/SEC
W24	8.91420	02.1/SEC
W25	8.91420	02.1/SEC
W26	8.91420	02.1/SEC
W27	8.91420	02.1/SEC
W28	8.91420	02.1/SEC
W29	8.91420	02.1/SEC
W30	8.91420	02.1/SEC
W31	8.91420	02.1/SEC
W32	8.91420	02.1/SEC
W33	8.91420	02.1/SEC
W34	8.91420	02.1/SEC
W35	8.91420	02.1/SEC
W36	8.91420	02.1/SEC
W37	8.91420	02.1/SEC
W38	8.91420	02.1/SEC
W39	8.91420	02.1/SEC
W40	8.91420	02.1/SEC
W41	8.91420	02.1/SEC
W42	8.91420	02.1/SEC
W43	8.91420	02.1/SEC
W44	8.91420	02.1/SEC
W45	8.91420	02.1/SEC
W46	8.91420	02.1/SEC
W47	8.91420	02.1/SEC
W48	8.91420	02.1/SEC
W49	8.91420	02.1/SEC
W50	8.91420	02.1/SEC
W51	8.91420	02.1/SEC
W52	8.91420	02.1/SEC
W53	8.91420	02.1/SEC
W54	8.91420	02.1/SEC
W55	8.91420	02.1/SEC
W56	8.91420	02.1/SEC
W57	8.91420	02.1/SEC
W58	8.91420	02.1/SEC
W59	8.91420	02.1/SEC
W60	8.91420	02.1/SEC
W61	8.91420	02.1/SEC
W62	8.91420	02.1/SEC
W63	8.91420	02.1/SEC
W64	8.91420	02.1/SEC
W65	8.91420	02.1/SEC
W66	8.91420	02.1/SEC
W67	8.91420	02.1/SEC
W68	8.91420	02.1/SEC
W69	8.91420	02.1/SEC
W70	8.91420	02.1/SEC
W71	8.91420	02.1/SEC
W72	8.91420	02.1/SEC
W73	8.91420	02.1/SEC
W74	8.91420	02.1/SEC
W75	8.91420	02.1/SEC
W76	8.91420	02.1/SEC
W77	8.91420	02.1/SEC
W78	8.91420	02.1/SEC
W79	8.91420	02.1/SEC
W80	8.91420	02.1/SEC
W81	8.91420	02.1/SEC
W82	8.91420	02.1/SEC
W83	8.91420	02.1/SEC
W84	8.91420	02.1/SEC
W85	8.91420	02.1/SEC
W86	8.91420	02.1/SEC
W87	8.91420	02.1/SEC
W88	8.91420	02.1/SEC
W89	8.91420	02.1/SEC
W90	8.91420	02.1/SEC
W91	8.91420	02.1/SEC
W92	8.91420	02.1/SEC
W93	8.91420	02.1/SEC
W94	8.91420	02.1/SEC
W95	8.91420	02.1/SEC
W96	8.91420	02.1/SEC
W97	8.91420	02.1/SEC
W98	8.91420	02.1/SEC
W99	8.91420	02.1/SEC

[illegible][illegible][illegible][illegible]

$V_{T1} = 1.0$	$V_D = 6.9$	$F_T = 0.0$	$ZR = 0.0$	$F_T = 0.0$	$VXR = 6.0$	F_T/SEC
$V_{D1} = 0.0$	$V_{D2} = 0.0$	E_T/SEC	$TY = 0.0$	SFC		

33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100

Figure 4-8i Monte Carlo Mode Check Problem Output

100

1

1. The first part of the paper is devoted to the study of the properties of the function $f(x)$ defined by the equation

12

*** PROJECTILE DISPERSION CASE SUMMARY ***

FIRE CONTROL PARAMETERS									
VX	1.10000 04	FT/SEC	WZ	0.0	FT/SEC	RHO	2.37800 03	SL/FT	
CX1	1.58547 02	(NONE)	CZ	1.19510 01	(NONE)	V2	2.50600 03	FT/SEC	
CNA	1.17300 03	FT	CZ2	6.20000 02	100/L	CNO	-7.50000 00	(NONE)	
CMPA	0.0	(NONE)	SM	0.0	DEG	THETO	0.0	DEG	
PS10	0.0	DEG	BTMS	0.0	RAD/SC	GANO	0.0	DEG	
A70	0.0	DEG	PDOT	0.0	RAD/SC	P	6.25000 02	RAD/SEC	
W	1.10000 01	LBS	ELL	2.11030 05	SL-SFT				
ALPCON	5.00000 01	DEG	ATV	0.0	1/50EG				
REAL WORLD PARAMETERS									
VX	1.10000 04	FT/SEC	WZ	0.0	FT/SEC	RHO	2.37800 03	SL/FT	
CX1	1.58547 02	(NONE)	CZ	1.19510 01	(NONE)	V2	2.50600 03	FT/SEC	
CNA	1.17300 03	FT	CZ2	6.20000 02	100/L	CNO	-7.50000 00	(NONE)	
CMPA	0.0	(NONE)	SM	0.0	DEG	THETO	0.0	DEG	
PS10	0.0	DEG	BTMS	0.0	RAD/SC	GANO	0.0	DEG	
AZ0	0.0	DEG	PDOT	0.0	RAD/SC	P	6.25000 02	RAD/SEC	
W	1.10000 01	LBS	ELL	2.11030 05	SL-SFT				
ALPCON	5.00000 01	DEG	ATV	0.0	1/50EG				
SYSTEMS CONTROLS									
IFCM	1 (NONE)		IFCC	0 (NONE)		IDMP	0 (NONE)		
ICFW	-9 (NONE)		ICNCL	0 (NONE)					
COMPLETED QUANTITIES									
YEAR1	0.0	FT	YEAR1	0.0	FT	YEAR1	0.0	FT	
TRAK1	0.0	FT	TRAK1	0.0	FT	TRAK1	0.0	FT	
DX1	0.0	FT/SEC	DX1	0.0	FT/SEC	DX1	0.0	FT/SEC	
DX2	0.0	FT/SEC	DX2	0.0	FT/SEC	DX2	0.0	FT/SEC	
DX3	0.0	FT/SEC	DX3	0.0	FT/SEC	DX3	0.0	FT/SEC	
DX4	0.0	FT/SEC	DX4	0.0	FT/SEC	DX4	0.0	FT/SEC	
DX5	0.0	FT/SEC	DX5	0.0	FT/SEC	DX5	0.0	FT/SEC	
DX6	0.0	FT/SEC	DX6	0.0	FT/SEC	DX6	0.0	FT/SEC	
DX7	0.0	FT/SEC	DX7	0.0	FT/SEC	DX7	0.0	FT/SEC	
DX8	0.0	FT/SEC	DX8	0.0	FT/SEC	DX8	0.0	FT/SEC	
DX9	0.0	FT/SEC	DX9	0.0	FT/SEC	DX9	0.0	FT/SEC	
DX10	0.0	FT/SEC	DX10	0.0	FT/SEC	DX10	0.0	FT/SEC	
DX11	0.0	FT/SEC	DX11	0.0	FT/SEC	DX11	0.0	FT/SEC	
DX12	0.0	FT/SEC	DX12	0.0	FT/SEC	DX12	0.0	FT/SEC	
DX13	0.0	FT/SEC	DX13	0.0	FT/SEC	DX13	0.0	FT/SEC	
DX14	0.0	FT/SEC	DX14	0.0	FT/SEC	DX14	0.0	FT/SEC	
DX15	0.0	FT/SEC	DX15	0.0	FT/SEC	DX15	0.0	FT/SEC	
DX16	0.0	FT/SEC	DX16	0.0	FT/SEC	DX16	0.0	FT/SEC	
DX17	0.0	FT/SEC	DX17	0.0	FT/SEC	DX17	0.0	FT/SEC	
DX18	0.0	FT/SEC	DX18	0.0	FT/SEC	DX18	0.0	FT/SEC	
DX19	0.0	FT/SEC	DX19	0.0	FT/SEC	DX19	0.0	FT/SEC	
DX20	0.0	FT/SEC	DX20	0.0	FT/SEC	DX20	0.0	FT/SEC	
DX21	0.0	FT/SEC	DX21	0.0	FT/SEC	DX21	0.0	FT/SEC	
DX22	0.0	FT/SEC	DX22	0.0	FT/SEC	DX22	0.0	FT/SEC	
DX23	0.0	FT/SEC	DX23	0.0	FT/SEC	DX23	0.0	FT/SEC	
DX24	0.0	FT/SEC	DX24	0.0	FT/SEC	DX24	0.0	FT/SEC	
DX25	0.0	FT/SEC	DX25	0.0	FT/SEC	DX25	0.0	FT/SEC	
DX26	0.0	FT/SEC	DX26	0.0	FT/SEC	DX26	0.0	FT/SEC	
DX27	0.0	FT/SEC	DX27	0.0	FT/SEC	DX27	0.0	FT/SEC	
DX28	0.0	FT/SEC	DX28	0.0	FT/SEC	DX28	0.0	FT/SEC	
DX29	0.0	FT/SEC	DX29	0.0	FT/SEC	DX29	0.0	FT/SEC	
DX30	0.0	FT/SEC	DX30	0.0	FT/SEC	DX30	0.0	FT/SEC	
DX31	0.0	FT/SEC	DX31	0.0	FT/SEC	DX31	0.0	FT/SEC	
DX32	0.0	FT/SEC	DX32	0.0	FT/SEC	DX32	0.0	FT/SEC	
DX33	0.0	FT/SEC	DX33	0.0	FT/SEC	DX33	0.0	FT/SEC	
DX34	0.0	FT/SEC	DX34	0.0	FT/SEC	DX34	0.0	FT/SEC	
DX35	0.0	FT/SEC	DX35	0.0	FT/SEC	DX35	0.0	FT/SEC	
DX36	0.0	FT/SEC	DX36	0.0	FT/SEC	DX36	0.0	FT/SEC	
DX37	0.0	FT/SEC	DX37	0.0	FT/SEC	DX37	0.0	FT/SEC	
DX38	0.0	FT/SEC	DX38	0.0	FT/SEC	DX38	0.0	FT/SEC	
DX39	0.0	FT/SEC	DX39	0.0	FT/SEC	DX39	0.0	FT/SEC	
DX40	0.0	FT/SEC	DX40	0.0	FT/SEC	DX40	0.0	FT/SEC	
DX41	0.0	FT/SEC	DX41	0.0	FT/SEC	DX41	0.0	FT/SEC	
DX42	0.0	FT/SEC	DX42	0.0	FT/SEC	DX42	0.0	FT/SEC	
DX43	0.0	FT/SEC	DX43	0.0	FT/SEC	DX43	0.0	FT/SEC	
DX44	0.0	FT/SEC	DX44	0.0	FT/SEC	DX44	0.0	FT/SEC	
DX45	0.0	FT/SEC	DX45	0.0	FT/SEC	DX45	0.0	FT/SEC	
DX46	0.0	FT/SEC	DX46	0.0	FT/SEC	DX46	0.0	FT/SEC	
DX47	0.0	FT/SEC	DX47	0.0	FT/SEC	DX47	0.0	FT/SEC	
DX48	0.0	FT/SEC	DX48	0.0	FT/SEC	DX48	0.0	FT/SEC	
DX49	0.0	FT/SEC	DX49	0.0	FT/SEC	DX49	0.0	FT/SEC	
DX50	0.0	FT/SEC	DX50	0.0	FT/SEC	DX50	0.0	FT/SEC	
DX51	0.0	FT/SEC	DX51	0.0	FT/SEC	DX51	0.0	FT/SEC	
DX52	0.0	FT/SEC	DX52	0.0	FT/SEC	DX52	0.0	FT/SEC	
DX53	0.0	FT/SEC	DX53	0.0	FT/SEC	DX53	0.0	FT/SEC	
DX54	0.0	FT/SEC	DX54	0.0	FT/SEC	DX54	0.0	FT/SEC	
DX55	0.0	FT/SEC	DX55	0.0	FT/SEC	DX55	0.0	FT/SEC	
DX56	0.0	FT/SEC	DX56	0.0	FT/SEC	DX56	0.0	FT/SEC	
DX57	0.0	FT/SEC	DX57	0.0	FT/SEC	DX57	0.0	FT/SEC	
DX58	0.0	FT/SEC	DX58	0.0	FT/SEC	DX58	0.0	FT/SEC	
DX59	0.0	FT/SEC	DX59	0.0	FT/SEC	DX59	0.0	FT/SEC	
DX60	0.0	FT/SEC	DX60	0.0	FT/SEC	DX60	0.0	FT/SEC	
DX61	0.0	FT/SEC	DX61	0.0	FT/SEC	DX61	0.0	FT/SEC	
DX62	0.0	FT/SEC	DX62	0.0	FT/SEC	DX62	0.0	FT/SEC	
DX63	0.0	FT/SEC	DX63	0.0	FT/SEC	DX63	0.0	FT/SEC	
DX64	0.0	FT/SEC	DX64	0.0	FT/SEC	DX64	0.0	FT/SEC	
DX65	0.0	FT/SEC	DX65	0.0	FT/SEC	DX65	0.0	FT/SEC	
DX66	0.0	FT/SEC	DX66	0.0	FT/SEC	DX66	0.0	FT/SEC	
DX67	0.0	FT/SEC	DX67	0.0	FT/SEC	DX67	0.0	FT/SEC	
DX68	0.0	FT/SEC	DX68	0.0	FT/SEC	DX68	0.0	FT/SEC	
DX69	0.0	FT/SEC	DX69	0.0	FT/SEC	DX69	0.0	FT/SEC	
DX70	0.0	FT/SEC	DX70	0.0	FT/SEC	DX70	0.0	FT/SEC	
DX71	0.0	FT/SEC	DX71	0.0	FT/SEC	DX71	0.0	FT/SEC	
DX72	0.0	FT/SEC	DX72	0.0	FT/SEC	DX72	0.0	FT/SEC	
DX73	0.0	FT/SEC	DX73	0.0	FT/SEC	DX73	0.0	FT/SEC	
DX74	0.0	FT/SEC	DX74	0.0	FT/SEC	DX74	0.0	FT/SEC	
DX75	0.0	FT/SEC	DX75	0.0	FT/SEC	DX75	0.0	FT/SEC	
DX76	0.0	FT/SEC	DX76	0.0	FT/SEC	DX76	0.0	FT/SEC	
DX77	0.0	FT/SEC	DX77	0.0	FT/SEC	DX77	0.0	FT/SEC	
DX78	0.0	FT/SEC	DX78	0.0	FT/SEC	DX78	0.0	FT/SEC	
DX79	0.0	FT/SEC	DX79	0.0	FT/SEC	DX79	0.0	FT/SEC	
DX80	0.0	FT/SEC	DX80	0.0	FT/SEC	DX80	0.0	FT/SEC	
DX81	0.0	FT/SEC	DX81	0.0	FT/SEC	DX81	0.0	FT/SEC	
DX82	0.0	FT/SEC	DX82	0.0	FT/SEC	DX82	0.0	FT/SEC	
DX83	0.0	FT/SEC	DX83	0.0	FT/SEC	DX83	0.0	FT/SEC	
DX84	0.0	FT/SEC	DX84	0.0	FT/SEC	DX84	0.0	FT/SEC	
DX85	0.0	FT/SEC	DX85	0.0	FT/SEC	DX85	0.0	FT/SEC	
DX86	0.0	FT/SEC	DX86	0.0	FT/SEC	DX86	0.0	FT/SEC	
DX87	0.0	FT/SEC	DX87	0.0	FT/SEC	DX87	0.0	FT/SEC	
DX88	0.0	FT/SEC	DX88	0.0	FT/SEC	DX88	0.0	FT/SEC	
DX89	0.0	FT/SEC	DX89	0.0	FT/SEC	DX89	0.0	FT/SEC	
DX90	0.0	FT/SEC	DX90	0.0	FT/SEC	DX90	0.0	FT/SEC	
DX91	0.0	FT/SEC	DX91	0.0	FT/SEC	DX91	0.0	FT/SEC	
DX92	0.0	FT/SEC	DX92	0.0	FT/SEC	DX92	0.0	FT/SEC	
DX93	0.0	FT/SEC	DX93	0.0	FT/SEC	DX93	0.0	FT/SEC	
DX94	0.0	FT/SEC	DX94	0.0	FT/SEC	DX94	0.0	FT/SEC	
DX95	0.0	FT/SEC	DX95	0.0	FT/SEC	DX95	0.0	FT/SEC	
DX96	0.0	FT/SEC	DX96	0.0	FT/SEC	DX96	0.0	FT/SEC	
DX97	0.0	FT/SEC	DX97	0.0	FT/SEC	DX97	0.0	FT/SEC	
DX98	0.0	FT/SEC	DX98	0.0	FT/SEC	DX98	0.0	FT/SEC	
DX99	0.0	FT/SEC	DX99	0.0	FT/SEC	DX99	0.0	FT/SEC	
DX100	0.0	FT/SEC	DX100	0.0	FT/SEC	DX100	0.0	FT/SEC	

Figure 8-j Monte Carlo Mode Check Problem Output

*** NOTICE ***

THE SPECIFICATION OF EITHER RNDT OR RNDX (IE
VARIABLES 320 OR 321) ON INITIAL OUTPUT VARIABLES
NECESSITATES UP-DATING THE INITIAL VALUES AND DERIVATIVES.
MEANS, AND VARIANCES. THE INITIAL CASES FOR AND PARTIAL
DERIVATIVES OF RNDT AND RNDX ARE MEANINGLESS, THUS THE
FULL PROJECTILE DISPERSION CODE SHOULD BE UTILIZED IN
MONTE CARLO SIMULATIONS. CALC = 01 TO OBTAIN
MEANINGFUL VALUES OF RNDT AND/OR RNDX. IF A TAYLOR
SERIES APPROXIMATION OF THE PROJECTILE DISPERSION CODE
(MCALC = 11, 2, OR 3) IS USED IN MONTE CARLO
SIMULATIONS THE RNDT AND RNDX RESULTS ARE MEANINGLESS.

Figure 4-8k Monte Carlo Mode Check Problem Output

NOMINAL CASE, INDEPENDENT VARIABLES ...

OE(116) = 0.0 OE(117) = 0.0 OE(118) = 0.0 OE(119) = 0.0

Figure 4-81 Monte Carlo Mode Check Problem Output

MINIMAL CASE. DEPENDENT VARIABLES ***

DE(305)	DE(307)	DE(308)	DE(320)	DE(317)	DE(322)	DE(323)	DE(324)	DE(518)	DE(510)
-5.40310-12	0.0	0.0	3.89340	0.0	-5.40010-12	0.0	0.0	1.09530-01	1.04250 04

Figure 4-8m Monte Carlo Mode Check Problem Output

DERIVATIVES		DEPENDENT VARIABLE									
IND. VAR.		306	307	308	320	317	322	323	324	518	519
116	1ST	0.0	2.19390-02	3.06790-17	0.0	0.0	0.0	2.19390-02	3.06790-17	0.0	0.0
	2ND	-1.064170-01	0.0	3.25440-19	0.0	9.99720-06	-1.064170-01	0.0	3.25440-19	9.99720-06	-1.35793 00
117	1ST	0.0	-3.06790-17	2.19390-02	0.0	0.0	0.0	-3.06790-17	2.19390-02	0.0	0.0
	2ND	-1.064170-01	0.0	3.06790-17	0.0	9.99720-06	-1.064170-01	0.0	3.06790-17	9.99720-06	-1.35793 00
118	1ST	0.0	3.41770-02	-3.90910-21	0.0	0.0	0.0	3.41770-02	-3.90910-21	0.0	0.0
	2ND	-4.67800-04	0.0	2.89570-22	0.0	4.67770-05	-4.67800-04	0.0	2.89570-22	4.67770-05	-5.33893-03
119	1ST	0.0	0.0	3.41770-02	0.0	0.0	0.0	0.0	3.41770-02	0.0	0.0
	2ND	-4.67800-04	0.0	0.0	0.0	4.67770-05	-4.67800-04	0.0	0.0	4.67770-05	-5.33893-03

Figure 4-8n Monte Carlo Mode Check-Problem Output

SECOND ORDER MIXED PARTIALS		DEPENDENT VARIABLES									
IND. VAR.		306	307	308	320	317	322	323	324	318	310
114	117	0.0	2.69850-17	4.81870-19	0.0	0.0	0.0	2.69850-17	4.81870-19	0.0	0.0
115	118	-2.38490-04	-1.37060-04	2.94490-19	0.0	2.28830-05	-2.38490-04	-1.37060-04	2.94490-19	2.28830-08	-2.74580-03
115	119	0.0	9.85150-19	0.0	0.0	0.0	0.0	9.85150-19	0.0	0.0	0.0
117	119	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
117	110	-2.38490-04	0.0	2.74130-18	0.0	2.28830-05	-2.38490-04	0.0	2.74130-18	2.28830-02	-2.74580-03
118	110	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Figure 4-80 Monte Carlo Mode Check Problem Output

2ND ORDER MEANS AND VARIANCES ASSUMING GAUSSIAN INDEPENDENT VARIABLES

	DEPENDENT VARIABLES									
	326	307	308	320	317	322	323	324	518	510
MEANS	-3.67140 00	0.0	1.67550-17	5.28870 00	3.73500-04	-3.89340 00	0.0	1.67550-17	1.09900-01	1.03220 04
VARIANCES	1.43700 01	9.46160 00	9.46160 00	4.58150 00	1.32240-07	1.41700 01	9.46160 00	9.46160 00	1.32240-07	1.67210 03
STD. DEV.	3.79580 00	3.07600 00	3.07600 00	2.23190 00	3.63650-04	3.79940 00	3.07600 00	3.07600 00	3.63650-04	4.12590 01

STANDARD DEVIATIONS WITH HIGHER PRECISION

3.79250 00	3.7907513795174500 00
3.07500 00	3.0755670705402220 00
3.07600 00	3.0755670705402220 00
2.23100 00	2.23103273273410500 00
3.63650-04	3.63650508212737000-04
3.79240 00	3.7907513795174500 00
3.07600 00	3.0759670705402220 00
3.07600 00	3.0755670705402220 00
3.63650-04	3.63650508212737000-04
4.12580 01	4.12675017194675100 01

Figure 4-8p Monte Carlo Mode Check Problem Output

*** SUMMARY OF MONTE CARLO RANDOM EXPERIMENTS ***

CASE 1	IND. VAR.	0.24600-02	-5.37240-01	2.64420 02	-5.68350 01	1.15950-03	-6.05980 00	5.95270 00	-1.93370 00	1.10690-01	1.02170 04
	DEP. VAR.	-1.19630 01	8.95270 00	-1.93570 00	1.22270 01						
CASE 2	IND. VAR.	3.99970-01	-5.65220-02	-2.61200 02	6.12150 01	1.13830-03	-7.85300 00	-8.03700 00	2.07150 00	1.10670-01	1.02210 04
	DEP. VAR.	-1.17460 01	-8.83700 00	2.67130 00	1.20020 01						
CASE 3	IND. VAR.	-1.72350-01	1.07330 00	-6.53380 01	7.30430 01	1.66930-04	2.15780 00	-2.23400 00	2.51670 00	1.09700-01	1.03970 04
	DEP. VAR.	-1.73350 00	-2.23340 00	2.51670 00	3.99750 00						
CASE 4	IND. VAR.	2.43750-01	-1.94070-02	6.95230 00	8.42670 01	1.28110-04	2.56040 00	3.11060-01	2.89730 00	1.09660-01	1.04050 04
	DEP. VAR.	-1.33300 00	3.11000-01	2.89730 00	3.87900 00						
CASE 5	IND. VAR.	-2.06840 00	2.01070 00	-2.92200 01	-2.11400 01	2.04570-04	1.74790 00	3.43080 00	-6.77380-01	1.09730-01	1.03500 04
	DEP. VAR.	-2.12540 00	-2.43060 00	-6.77380-01	3.91850 00						
CASE 6	IND. VAR.	-1.29670 00	-2.83260 00	-1.11840 02	2.34790 01	2.46430-04	1.33520 00	-3.84330 00	7.38940-01	1.09780-01	1.03820 04
	DEP. VAR.	-2.55810 00	-3.84330 00	7.38940-01	4.13520 00						
CASE 7	IND. VAR.	2.32300-01	-1.21310 00	-3.84770 01	2.12500 02	7.38790-04	-3.70760 00	-1.30210 00	7.19560 00	1.10270-01	1.02920 04
	DEP. VAR.	-7.60300 00	-1.30210 00	7.19560 00	2.15460 00						
CASE 8	IND. VAR.	6.44290-01	6.90250-01	1.13970 02	1.73620 01	2.15510-04	1.65500 00	3.90270 00	6.07490-01	1.09750-01	1.03870 04
	DEP. VAR.	-2.25840 00	3.90270 00	6.07490-01	4.28240 00						
CASE 9	IND. VAR.	-4.11170-02	4.61390-01	4.77750 01	-1.11220 02	2.30840-04	1.49870 00	1.62900 00	-3.78430 00	1.09780-01	1.03840 04
	DEP. VAR.	-2.39460 00	1.62900 00	-3.78430 00	4.38410 00						
CASE 10	IND. VAR.	-1.61450-01	9.24210-02	-3.87440 01	-8.75030 01	1.55110-04	2.28030 00	-1.32610 00	-2.98500 00	1.09690-01	1.04000 04
	DEP. VAR.	-1.61310 00	-1.32610 00	-2.98500 00	3.52360 00						

Figure 4-8q Monte Carlo Mode Check Problem Output

CASE 95	-1.3260D 00	-5.2271D-01	1.4574D 02	2.1560D 01	4.1963D-04	-4.4873D-01	4.9351D 00	2.4258D 00	1.0995D-01	1.0350D 04
IND. VAR.	-1.3260D 00	-5.2271D-01	1.4574D 02	2.1560D 01	4.1963D-04	-4.4873D-01	4.9351D 00	2.4258D 00	1.0995D-01	1.0350D 04
DEP. VAR.	-1.3260D 00	-5.2271D-01	1.4574D 02	2.1560D 01	4.1963D-04	-4.4873D-01	4.9351D 00	2.4258D 00	1.0995D-01	1.0350D 04
CASE 96	-1.2033D-01	3.4159D-01	6.3289D 01	8.4753D 01	1.8442D-04	1.9767D 00	2.1573D 00	2.9068D 00	1.0971D-01	1.0394D 04
IND. VAR.	-1.2033D-01	3.4159D-01	6.3289D 01	8.4753D 01	1.8442D-04	1.9767D 00	2.1573D 00	2.9068D 00	1.0971D-01	1.0394D 04
DEP. VAR.	-1.2033D-01	3.4159D-01	6.3289D 01	8.4753D 01	1.8442D-04	1.9767D 00	2.1573D 00	2.9068D 00	1.0971D-01	1.0394D 04
CASE 97	-1.1267D 00	-1.2769D-01	-1.7649D 01	-2.4597D 00	1.4574D-05	3.7414D 00	-6.2791D-01	-8.6669D-02	1.0954D-01	1.0424D 04
IND. VAR.	-1.1267D 00	-1.2769D-01	-1.7649D 01	-2.4597D 00	1.4574D-05	3.7414D 00	-6.2791D-01	-8.6669D-02	1.0954D-01	1.0424D 04
DEP. VAR.	-1.1267D 00	-1.2769D-01	-1.7649D 01	-2.4597D 00	1.4574D-05	3.7414D 00	-6.2791D-01	-8.6669D-02	1.0954D-01	1.0424D 04
CASE 98	-3.9821D-01	2.5419D-01	-1.2800D 02	-4.6585D 01	2.2326D-04	1.5750D 00	-3.7065D 00	-1.5538D 00	1.0973D-01	1.0305D 04
IND. VAR.	-3.9821D-01	2.5419D-01	-1.2800D 02	-4.6585D 01	2.2326D-04	1.5750D 00	-3.7065D 00	-1.5538D 00	1.0973D-01	1.0305D 04
DEP. VAR.	-3.9821D-01	2.5419D-01	-1.2800D 02	-4.6585D 01	2.2326D-04	1.5750D 00	-3.7065D 00	-1.5538D 00	1.0973D-01	1.0305D 04
CASE 99	5.9933D-01	1.2144D-01	9.9932D 01	-2.3536D 02	1.0031D-03	-5.3750D 00	3.0312D 00	-7.9758D 00	1.1053D-01	1.0245D 04
IND. VAR.	5.9933D-01	1.2144D-01	9.9932D 01	-2.3536D 02	1.0031D-03	-5.3750D 00	3.0312D 00	-7.9758D 00	1.1053D-01	1.0245D 04
DEP. VAR.	5.9933D-01	1.2144D-01	9.9932D 01	-2.3536D 02	1.0031D-03	-5.3750D 00	3.0312D 00	-7.9758D 00	1.1053D-01	1.0245D 04
CASE 100	4.1297D-01	6.5879D-01	4.0767D 00	-9.3185D 00	4.7415D-05	3.8439D 00	1.4841D-01	-3.0404D-01	1.0954D-01	1.0424D 04
IND. VAR.	4.1297D-01	6.5879D-01	4.0767D 00	-9.3185D 00	4.7415D-05	3.8439D 00	1.4841D-01	-3.0404D-01	1.0954D-01	1.0424D 04
DEP. VAR.	4.1297D-01	6.5879D-01	4.0767D 00	-9.3185D 00	4.7415D-05	3.8439D 00	1.4841D-01	-3.0404D-01	1.0954D-01	1.0424D 04

Figure 4-8r Monte Carlo Mode Check Problem Output

MONTE CARLO RESULTS SAMPLE = 100 INTERNAL REJECTS = 12 USER REJECTS = 0

DEPENDENT VARIABLES WITH INTERNAL RANGES

CODE NUMBER = 306 MEAN = -0.2394300 C1 VAR = 0.4807050 C1 INTERVAL = 0.1137230 C1
 NO. REJECTS = 1 (MEAN) = -0.33603370 C1 (VAR) = 0.1436980 C2 TOLERANCE = 0.1137230 C2

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST.	FUNCT.
-0.1463700 C2	-0.1565500 C2	-0.1412940 C2	0	0.0	
-0.1412940 C2	-0.1412940 C2	-0.1299120 C2	0	0.0	
-0.1299120 C2	-0.1299120 C2	-0.1185300 C2	0	0.0	
-0.1185300 C2	-0.1185300 C2	-0.1071470 C2	0	0.0	
-0.1071470 C2	-0.1071470 C2	-0.0957650 C2	0	0.0	
-0.0957650 C2	-0.0957650 C2	-0.0843820 C2	0	0.0	
-0.0843820 C2	-0.0843820 C2	-0.0730000 C2	0	0.0	
-0.0730000 C2	-0.0730000 C2	-0.0616170 C2	0	0.0	
-0.0616170 C2	-0.0616170 C2	-0.0502350 C2	0	0.0	
-0.0502350 C2	-0.0502350 C2	-0.0388530 C2	0	0.0	
-0.0388530 C2	-0.0388530 C2	-0.0274700 C2	0	0.0	
-0.0274700 C2	-0.0274700 C2	-0.0160870 C2	0	0.0	
-0.0160870 C2	-0.0160870 C2	-0.0047050 C2	0	0.0	
-0.0047050 C2	-0.0047050 C2	0.0066770 C1	3	0.0	
0.0066770 C1	0.0066770 C1	0.0180590 C1	7	0.0	
0.0180590 C1	0.0180590 C1	0.0294410 C1	23	0.0	
0.0294410 C1	0.0294410 C1	0.0408230 C1	31	0.0	
0.0408230 C1	0.0408230 C1	0.0522050 C1	10	0.0	
0.0522050 C1	0.0522050 C1	0.0635870 C1	0	0.0	
0.0635870 C1	0.0635870 C1	0.0749690 C1	0	0.0	

CODE NUMBER = 307 MEAN = -0.1675300 C0 VAR = 0.8180490 C1 INTERVAL = 0.9227900 C1
 NO. REJECTS = 0 (MEAN) = 0.0 (VAR) = 0.9461570 C1 TOLERANCE = 0.9227900 C1

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST.	FUNCT.
-0.8766510 C1	-0.9227900 C1	-0.8305110 C1	0	0.0	
-0.8305110 C1	-0.8305110 C1	-0.7782320 C1	0	0.0	
-0.7782320 C1	-0.7782320 C1	-0.7259530 C1	0	0.0	
-0.7259530 C1	-0.7259530 C1	-0.6736740 C1	0	0.0	
-0.6736740 C1	-0.6736740 C1	-0.6213950 C1	0	0.0	
-0.6213950 C1	-0.6213950 C1	-0.5691160 C1	0	0.0	
-0.5691160 C1	-0.5691160 C1	-0.5168370 C1	0	0.0	
-0.5168370 C1	-0.5168370 C1	-0.4645580 C1	0	0.0	
-0.4645580 C1	-0.4645580 C1	-0.4122790 C1	0	0.0	
-0.4122790 C1	-0.4122790 C1	-0.3600000 C1	0	0.0	
-0.3600000 C1	-0.3600000 C1	-0.3077210 C1	0	0.0	
-0.3077210 C1	-0.3077210 C1	-0.2554420 C1	0	0.0	
-0.2554420 C1	-0.2554420 C1	-0.2031630 C1	0	0.0	
-0.2031630 C1	-0.2031630 C1	-0.1508840 C1	0	0.0	
-0.1508840 C1	-0.1508840 C1	-0.0986050 C1	0	0.0	
-0.0986050 C1	-0.0986050 C1	-0.0463260 C1	0	0.0	
-0.0463260 C1	-0.0463260 C1	0.0059570 C1	0	0.0	
0.0059570 C1	0.0059570 C1	0.0582380 C1	0	0.0	
0.0582380 C1	0.0582380 C1	0.1105190 C1	0	0.0	
0.1105190 C1	0.1105190 C1	0.1628000 C1	0	0.0	
0.1628000 C1	0.1628000 C1	0.2150810 C1	0	0.0	
0.2150810 C1	0.2150810 C1	0.2673620 C1	0	0.0	
0.2673620 C1	0.2673620 C1	0.3196430 C1	0	0.0	
0.3196430 C1	0.3196430 C1	0.3719240 C1	0	0.0	
0.3719240 C1	0.3719240 C1	0.4242050 C1	0	0.0	
0.4242050 C1	0.4242050 C1	0.4764860 C1	0	0.0	
0.4764860 C1	0.4764860 C1	0.5287670 C1	0	0.0	
0.5287670 C1	0.5287670 C1	0.5810480 C1	0	0.0	
0.5810480 C1	0.5810480 C1	0.6333290 C1	0	0.0	
0.6333290 C1	0.6333290 C1	0.6856100 C1	0	0.0	
0.6856100 C1	0.6856100 C1	0.7378910 C1	0	0.0	
0.7378910 C1	0.7378910 C1	0.7901720 C1	0	0.0	
0.7901720 C1	0.7901720 C1	0.8424530 C1	0	0.0	
0.8424530 C1	0.8424530 C1	0.8947340 C1	0	0.0	
0.8947340 C1	0.8947340 C1	0.9470150 C1	0	0.0	
0.9470150 C1	0.9470150 C1	0.9992960 C1	0	0.0	

CODE NUMBER = 308 MEAN = -0.1048630 C1 VAR = 0.8205650 C1 INTERVAL = 0.9227900 C1
 NO. REJECTS = 0 (MEAN) = 0.1675520 C1 (VAR) = 0.9461570 C1 TOLERANCE = 0.9227900 C1

Figure 4-8s Monte Carlo Mode Check Problem Output


```
CONF NUMBERS = 322      MEAN = 0.141790 01      VAR = 0.479753 01      INTERVAL = 0.113723 01
                        TOFF = 0.147680 02      TOFFRANCE = 0.113723 02      INTERVAL = 0.113723 01
```

CLASS MARK	LOWER ROUND	UPPER ROUND	FREQ	DIST.	FUNCT.
145700	0	0	0	0	0
145750	0	0	0	0	0
145800	0	0	0	0	0
145850	0	0	0	0	0
145900	0	0	0	0	0
145950	0	0	0	0	0
146000	0	0	0	0	0
146050	0	0	0	0	0
146100	0	0	0	0	0
146150	0	0	0	0	0
146200	0	0	0	0	0
146250	0	0	0	0	0
146300	0	0	0	0	0
146350	0	0	0	0	0
146400	0	0	0	0	0
146450	0	0	0	0	0
146500	0	0	0	0	0
146550	0	0	0	0	0
146600	0	0	0	0	0
146650	0	0	0	0	0
146700	0	0	0	0	0
146750	0	0	0	0	0
146800	0	0	0	0	0
146850	0	0	0	0	0
146900	0	0	0	0	0
146950	0	0	0	0	0
147000	0	0	0	0	0
147050	0	0	0	0	0
147100	0	0	0	0	0
147150	0	0	0	0	0
147200	0	0	0	0	0
147250	0	0	0	0	0
147300	0	0	0	0	0
147350	0	0	0	0	0
147400	0	0	0	0	0
147450	0	0	0	0	0
147500	0	0	0	0	0
147550	0	0	0	0	0
147600	0	0	0	0	0
147650	0	0	0	0	0
147700	0	0	0	0	0
147750	0	0	0	0	0
147800	0	0	0	0	0
147850	0	0	0	0	0
147900	0	0	0	0	0
147950	0	0	0	0	0
148000	0	0	0	0	0
148050	0	0	0	0	0
148100	0	0	0	0	0
148150	0	0	0	0	0
148200	0	0	0	0	0
148250	0	0	0	0	0
148300	0	0	0	0	0
148350	0	0	0	0	0
148400	0	0	0	0	0
148450	0	0	0	0	0
148500	0	0	0	0	0
148550	0	0	0	0	0
148600	0	0	0	0	0
148650	0	0	0	0	0
148700	0	0	0	0	0
148750	0	0	0	0	0
148800	0	0	0	0	0
148850	0	0	0	0	0
148900	0	0	0	0	0
148950	0	0	0	0	0
149000	0	0	0	0	0
149050	0	0	0	0	0
149100	0	0	0	0	0
149150	0	0	0	0	0
149200	0	0	0	0	0
149250	0	0	0	0	0
149300	0	0	0	0	0
149350	0	0	0	0	0
149400	0	0	0	0	0
149450	0	0	0	0	0
149500	0	0	0	0	0
149550	0	0	0	0	0
149600	0	0	0	0	0
149650	0	0	0	0	0
149700	0	0	0	0	0
149750	0	0	0	0	0
149800	0	0	0	0	0
149850	0	0	0	0	0
149900	0	0	0	0	0
149950	0	0	0	0	0
150000	0	0	0	0	0

```
CODE NUMBER = 223      MEAN = -0.188750 00      VAR = 0.818949 01      INTERVAL = 0.922790 00
CONFIDENCE = 0.95      (MEAN) = 0.0      (VAR) = 0.0461570 01      TOLERANCE = 0.922790 01
```

CLASS MARK	LOWER ROUND	UPPER ROUND	FREQ	DIST. FUNCT.
35	0.927500	0.932500	0	0.0000
36	0.932500	0.937500	0	0.0000
37	0.937500	0.942500	0	0.0000
38	0.942500	0.947500	0	0.0000
39	0.947500	0.952500	0	0.0000
40	0.952500	0.957500	0	0.0000
41	0.957500	0.962500	0	0.0000
42	0.962500	0.967500	0	0.0000
43	0.967500	0.972500	0	0.0000
44	0.972500	0.977500	0	0.0000
45	0.977500	0.982500	0	0.0000
46	0.982500	0.987500	0	0.0000
47	0.987500	0.992500	0	0.0000
48	0.992500	0.997500	0	0.0000
49	0.997500	1.002500	0	0.0000
50	1.002500	1.007500	0	0.0000
51	1.007500	1.012500	0	0.0000
52	1.012500	1.017500	0	0.0000
53	1.017500	1.022500	0	0.0000
54	1.022500	1.027500	0	0.0000
55	1.027500	1.032500	0	0.0000
56	1.032500	1.037500	0	0.0000
57	1.037500	1.042500	0	0.0000
58	1.042500	1.047500	0	0.0000
59	1.047500	1.052500	0	0.0000
60	1.052500	1.057500	0	0.0000
61	1.057500	1.062500	0	0.0000
62	1.062500	1.067500	0	0.0000
63	1.067500	1.072500	0	0.0000
64	1.072500	1.077500	0	0.0000
65	1.077500	1.082500	0	0.0000
66	1.082500	1.087500	0	0.0000
67	1.087500	1.092500	0	0.0000
68	1.092500	1.097500	0	0.0000
69	1.097500	1.102500	0	0.0000
70	1.102500	1.107500	0	0.0000
71	1.107500	1.112500	0	0.0000
72	1.112500	1.117500	0	0.0000
73	1.117500	1.122500	0	0.0000
74	1.122500	1.127500	0	0.0000
75	1.127500	1.132500	0	0.0000
76	1.132500	1.137500	0	0.0000
77	1.137500	1.142500	0	0.0000
78	1.142500	1.147500	0	0.0000
79	1.147500	1.152500	0	0.0000
80	1.152500	1.157500	0	0.0000
81	1.157500	1.162500	0	0.0000
82	1.162500	1.167500	0	0.0000
83	1.167500	1.172500	0	0.0000
84	1.172500	1.177500	0	0.0000
85	1.177500	1.182500	0	0.0000
86	1.182500	1.187500	0	0.0000
87	1.187500	1.192500	0	0.0000
88	1.192500	1.197500	0	0.0000
89	1.197500	1.202500	0	0.0000
90	1.202500	1.207500	0	0.0000
91	1.207500	1.212500	0	0.0000
92	1.212500	1.217500	0	0.0000
93	1.217500	1.222500	0	0.0000
94	1.222500	1.227500	0	0.0000
95	1.227500	1.232500	0	0.0000
96	1.232500	1.237500	0	0.0000
97	1.237500	1.242500	0	0.0000
98	1.242500	1.247500	0	0.0000
99	1.247500	1.252500	0	0.0000
100	1.252500	1.257500	0	0.0000

Figure 4-8u Monte Carlo Mode Check Problem Output

C-509814D 01 0-553674D 01 0-645553D 01 1- 0-123144D-01
 C-509815D 01 0-645553D 01 0-738232D 01 1- 0-123144D-01
 C-509816D 01 0-738232D 01 0-830511D 01 1- 0-123144D-01
 C-509817D 01 0-830511D 01 0-922790D 01 0- 0-0

CODE NUMBER = 324 MEAN = -0.104863D-01 VAR = 0.820565D-01 INTERVAL = 0.922790D-01
 NO. REJECTS = 0 (MEAN) = 0.167552D-16 (VAR) = 0.946157D-01 TOLERANCE = 0.922790D-01

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
0-509814D 01	0-553674D 01	0-645553D 01	1	0-123144D-01
0-509815D 01	0-645553D 01	0-738232D 01	1	0-123144D-01
0-509816D 01	0-738232D 01	0-830511D 01	1	0-123144D-01
0-509817D 01	0-830511D 01	0-922790D 01	0	0-0
0-509818D 01	0-922790D 01	0-101568D 01	0	0-0
0-509819D 01	0-101568D 01	0-110395D 01	0	0-0
0-509820D 01	0-110395D 01	0-119222D 01	0	0-0
0-509821D 01	0-119222D 01	0-128049D 01	0	0-0
0-509822D 01	0-128049D 01	0-136876D 01	0	0-0
0-509823D 01	0-136876D 01	0-145703D 01	0	0-0
0-509824D 01	0-145703D 01	0-154530D 01	0	0-0
0-509825D 01	0-154530D 01	0-163357D 01	0	0-0
0-509826D 01	0-163357D 01	0-172184D 01	0	0-0
0-509827D 01	0-172184D 01	0-181011D 01	0	0-0
0-509828D 01	0-181011D 01	0-189838D 01	0	0-0
0-509829D 01	0-189838D 01	0-198665D 01	0	0-0
0-509830D 01	0-198665D 01	0-207492D 01	0	0-0
0-509831D 01	0-207492D 01	0-216319D 01	0	0-0
0-509832D 01	0-216319D 01	0-225146D 01	0	0-0
0-509833D 01	0-225146D 01	0-233973D 01	0	0-0
0-509834D 01	0-233973D 01	0-242800D 01	0	0-0
0-509835D 01	0-242800D 01	0-251627D 01	0	0-0
0-509836D 01	0-251627D 01	0-260454D 01	0	0-0
0-509837D 01	0-260454D 01	0-269281D 01	0	0-0
0-509838D 01	0-269281D 01	0-278108D 01	0	0-0
0-509839D 01	0-278108D 01	0-286935D 01	0	0-0
0-509840D 01	0-286935D 01	0-295762D 01	0	0-0
0-509841D 01	0-295762D 01	0-304589D 01	0	0-0
0-509842D 01	0-304589D 01	0-313416D 01	0	0-0
0-509843D 01	0-313416D 01	0-322243D 01	0	0-0
0-509844D 01	0-322243D 01	0-331070D 01	0	0-0
0-509845D 01	0-331070D 01	0-339897D 01	0	0-0
0-509846D 01	0-339897D 01	0-348724D 01	0	0-0
0-509847D 01	0-348724D 01	0-357551D 01	0	0-0
0-509848D 01	0-357551D 01	0-366378D 01	0	0-0
0-509849D 01	0-366378D 01	0-375205D 01	0	0-0
0-509850D 01	0-375205D 01	0-384032D 01	0	0-0
0-509851D 01	0-384032D 01	0-392859D 01	0	0-0
0-509852D 01	0-392859D 01	0-401686D 01	0	0-0
0-509853D 01	0-401686D 01	0-410513D 01	0	0-0
0-509854D 01	0-410513D 01	0-419340D 01	0	0-0
0-509855D 01	0-419340D 01	0-428167D 01	0	0-0
0-509856D 01	0-428167D 01	0-436994D 01	0	0-0
0-509857D 01	0-436994D 01	0-445821D 01	0	0-0
0-509858D 01	0-445821D 01	0-454648D 01	0	0-0
0-509859D 01	0-454648D 01	0-463475D 01	0	0-0
0-509860D 01	0-463475D 01	0-472302D 01	0	0-0
0-509861D 01	0-472302D 01	0-481129D 01	0	0-0
0-509862D 01	0-481129D 01	0-489956D 01	0	0-0
0-509863D 01	0-489956D 01	0-498783D 01	0	0-0
0-509864D 01	0-498783D 01	0-507610D 01	0	0-0
0-509865D 01	0-507610D 01	0-516437D 01	0	0-0
0-509866D 01	0-516437D 01	0-525264D 01	0	0-0
0-509867D 01	0-525264D 01	0-534091D 01	0	0-0
0-509868D 01	0-534091D 01	0-542918D 01	0	0-0
0-509869D 01	0-542918D 01	0-551745D 01	0	0-0
0-509870D 01	0-551745D 01	0-560572D 01	0	0-0
0-509871D 01	0-560572D 01	0-569399D 01	0	0-0
0-509872D 01	0-569399D 01	0-578226D 01	0	0-0
0-509873D 01	0-578226D 01	0-587053D 01	0	0-0
0-509874D 01	0-587053D 01	0-595880D 01	0	0-0
0-509875D 01	0-595880D 01	0-604707D 01	0	0-0
0-509876D 01	0-604707D 01	0-613534D 01	0	0-0
0-509877D 01	0-613534D 01	0-622361D 01	0	0-0
0-509878D 01	0-622361D 01	0-631188D 01	0	0-0
0-509879D 01	0-631188D 01	0-640015D 01	0	0-0
0-509880D 01	0-640015D 01	0-648842D 01	0	0-0
0-509881D 01	0-648842D 01	0-657669D 01	0	0-0
0-509882D 01	0-657669D 01	0-666496D 01	0	0-0
0-509883D 01	0-666496D 01	0-675323D 01	0	0-0
0-509884D 01	0-675323D 01	0-684150D 01	0	0-0
0-509885D 01	0-684150D 01	0-692977D 01	0	0-0
0-509886D 01	0-692977D 01	0-701804D 01	0	0-0
0-509887D 01	0-701804D 01	0-710631D 01	0	0-0
0-509888D 01	0-710631D 01	0-719458D 01	0	0-0
0-509889D 01	0-719458D 01	0-728285D 01	0	0-0
0-509890D 01	0-728285D 01	0-737112D 01	0	0-0
0-509891D 01	0-737112D 01	0-745939D 01	0	0-0
0-509892D 01	0-745939D 01	0-754766D 01	0	0-0
0-509893D 01	0-754766D 01	0-763593D 01	0	0-0
0-509894D 01	0-763593D 01	0-772420D 01	0	0-0
0-509895D 01	0-772420D 01	0-781247D 01	0	0-0
0-509896D 01	0-781247D 01	0-790074D 01	0	0-0
0-509897D 01	0-790074D 01	0-798901D 01	0	0-0
0-509898D 01	0-798901D 01	0-807728D 01	0	0-0
0-509899D 01	0-807728D 01	0-816555D 01	0	0-0
0-509900D 01	0-816555D 01	0-825382D 01	0	0-0

CODE NUMBER = 518 MEAN = 0.109761D-00 VAR = 0.459456D-07 INTERVAL = 0.109995D-03
 NO. REJECTS = 0 (MEAN) = 0.109995D-03 TOLERANCE = 0.109995D-03

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
0-509814D 01	0-553674D 01	0-645553D 01	1	0-123144D-01
0-509815D 01	0-645553D 01	0-738232D 01	1	0-123144D-01
0-509816D 01	0-738232D 01	0-830511D 01	1	0-123144D-01
0-509817D 01	0-830511D 01	0-922790D 01	0	0-0
0-509818D 01	0-922790D 01	0-101568D 01	0	0-0
0-509819D 01	0-101568D 01	0-110395D 01	0	0-0
0-509820D 01	0-110395D 01	0-119222D 01	0	0-0
0-509821D 01	0-119222D 01	0-128049D 01	0	0-0
0-509822D 01	0-128049D 01	0-136876D 01	0	0-0
0-509823D 01	0-136876D 01	0-145703D 01	0	0-0
0-509824D 01	0-145703D 01	0-154530D 01	0	0-0
0-509825D 01	0-154530D 01	0-163357D 01	0	0-0
0-509826D 01	0-163357D 01	0-172184D 01	0	0-0
0-509827D 01	0-172184D 01	0-181011D 01	0	0-0
0-509828D 01	0-181011D 01	0-189838D 01	0	0-0
0-509829D 01	0-189838D 01	0-198665D 01	0	0-0
0-509830D 01	0-198665D 01	0-207492D 01	0	0-0
0-509831D 01	0-207492D 01	0-216319D 01	0	0-0
0-509832D 01	0-216319D 01	0-225146D 01	0	0-0
0-509833D 01	0-225146D 01	0-233973D 01	0	0-0
0-509834D 01	0-233973D 01	0-242800D 01	0	0-0
0-509835D 01	0-242800D 01	0-251627D 01	0	0-0
0-509836D 01	0-251627D 01	0-260454D 01	0	0-0
0-509837D 01	0-260454D 01	0-269281D 01	0	0-0
0-509838D 01	0-269281D 01	0-278108D 01	0	0-0
0-509839D 01	0-278108D 01	0-286935D 01	0	0-0
0-509840D 01	0-286935D 01	0-295762D 01	0	0-0
0-509841D 01	0-295762D 01	0-304589D 01	0	0-0
0-509842D 01	0-304589D 01	0-313416D 01	0	0-0
0-509843D 01	0-313416D 01	0-322243D 01	0	0-0
0-509844D 01	0-322243D 01	0-331070D 01	0	0-0
0-509845D 01	0-331070D 01	0-339897D 01	0	0-0
0-509846D 01	0-339897D 01	0-348724D 01	0	0-0
0-509847D 01	0-348724D 01	0-357551D 01	0	0-0
0-509848D 01	0-357551D 01	0-366378D 01	0	0-0
0-509849D 01	0-366378D 01	0-375205D 01	0	0-0
0-509850D 01	0-375205D 01	0-384032D 01	0	0-0
0-509851D 01	0-384032D 01	0-392859D 01	0	0-0
0-509852D 01	0-392859D 01	0-401686D 01	0	0-0
0-509853D 01	0-401686D 01	0-410513D 01	0	0-0
0-509854D 01	0-410513D 01	0-419340D 01	0	0-0
0-509855D 01	0-419340D 01	0-428167D 01	0	0-0
0-509856D 01	0-428167D 01	0-436994D 01	0	0-0
0-509857D 01	0-436994D 01	0-445821D 01	0	0-0
0-509858D 01	0-445821D 01	0-454648D 01	0	0-0
0-509859D 01	0-454648D 01	0-463475D 01	0	0-0
0-509860D 01	0-463475D 01	0-472302D 01	0	0-0
0-509861D 01	0-472302D 01	0-481129D 01	0	0-0
0-509862D 01	0-481129D 01	0-489956D 01	0	0-0
0-509863D 01	0-489956D 01	0-498783D 01	0	0-0
0-509864D 01	0-498783D 01	0-507610D 01	0	0-0
0-509865D 01	0-507610D 01	0-516437D 01	0	0-0
0-509866D 01	0-516437D 01	0-525264D 01	0	0-0
0-509867D 01	0-525264D 01	0-534091D 01	0	0-0
0-509868D 01	0-534091D 01	0-542918D 01	0	0-0
0-509869D 01	0-542918D 01	0-551745D 01	0	0-0
0-509870D 01	0-551745D 01	0-560572D 01	0	0-0
0-509871D 01	0-560572D 01	0-569399D 01	0	0-0
0-509872D 01	0-569399D 01	0-578226D 01	0	0-0
0-509873D 01	0-578226D 01	0-587053D 01	0	0-0
0-509874D 01	0-587053D 01	0-595880D 01	0	0-0
0-509875D 01	0-595880D 01	0-604707D 01	0	0-0
0-509876D 01	0-604707D 01	0-613534D 01	0	0-0
0-509877D 01	0-613534D 01	0-622361D 01	0	0-0
0-509878D 01	0-622361D 01	0-631188D 01	0	0-0
0-509879D 01	0-631188D 01	0-640015D 01	0	0-0
0-509880D 01	0-640015D 01	0-648842D 01	0	0-0
0-509881D 01	0-648842D 01	0-657669D 01	0	0-0
0-509882D 01	0-657669D 01	0-666496D 01	0	0-0
0-509883D 01	0-666496D 01	0-675323D 01	0	0-0
0-509884D 01	0-675323D 01	0-684150D 01	0	0-0
0-509885D 01	0-684150D 01	0-692977D 01	0	0-0
0-509886D 01	0-692977D 01	0-701804D 01	0	0-0
0-509887D 01	0-701804D 01	0-710631D 01	0	0-0
0-509888D 01	0-710631D 01	0-719458D 01	0	0-0
0-509889D 01	0-719458D 01	0-728285D 01	0	0-0
0-509890D 01	0-728285D 01	0-737112D 01	0	0-0
0-509891D 01	0-737112D 01	0-745939D 01	0	0-0
0-509892D 01	0-			

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
0.125000	0.125170	0.125340	1	0.000000
0.125340	0.125510	0.125680	1	0.000000
0.125680	0.125850	0.126020	1	0.000000
0.126020	0.126190	0.126360	1	0.000000
0.126360	0.126530	0.126700	1	0.000000
0.126700	0.126870	0.127040	1	0.000000
0.127040	0.127210	0.127380	1	0.000000
0.127380	0.127550	0.127720	1	0.000000
0.127720	0.127890	0.128060	1	0.000000
0.128060	0.128230	0.128400	1	0.000000
0.128400	0.128570	0.128740	1	0.000000
0.128740	0.128910	0.129080	1	0.000000
0.129080	0.129250	0.129420	1	0.000000
0.129420	0.129590	0.129760	1	0.000000
0.129760	0.129930	0.130100	1	0.000000
0.130100	0.130270	0.130440	1	0.000000
0.130440	0.130610	0.130780	1	0.000000
0.130780	0.130950	0.131120	1	0.000000
0.131120	0.131290	0.131460	1	0.000000
0.131460	0.131630	0.131800	1	0.000000
0.131800	0.131970	0.132140	1	0.000000
0.132140	0.132310	0.132480	1	0.000000
0.132480	0.132650	0.132820	1	0.000000
0.132820	0.132990	0.133160	1	0.000000
0.133160	0.133330	0.133500	1	0.000000
0.133500	0.133670	0.133840	1	0.000000
0.133840	0.134010	0.134180	1	0.000000
0.134180	0.134350	0.134520	1	0.000000
0.134520	0.134690	0.134860	1	0.000000
0.134860	0.135030	0.135200	1	0.000000
0.135200	0.135370	0.135540	1	0.000000
0.135540	0.135710	0.135880	1	0.000000
0.135880	0.136050	0.136220	1	0.000000
0.136220	0.136390	0.136560	1	0.000000
0.136560	0.136730	0.136900	1	0.000000
0.136900	0.137070	0.137240	1	0.000000
0.137240	0.137410	0.137580	1	0.000000
0.137580	0.137750	0.137920	1	0.000000
0.137920	0.138090	0.138260	1	0.000000
0.138260	0.138430	0.138600	1	0.000000
0.138600	0.138770	0.138940	1	0.000000
0.138940	0.139110	0.139280	1	0.000000
0.139280	0.139450	0.139620	1	0.000000
0.139620	0.139790	0.139960	1	0.000000
0.139960	0.140130	0.140300	1	0.000000
0.140300	0.140470	0.140640	1	0.000000
0.140640	0.140810	0.140980	1	0.000000
0.140980	0.141150	0.141320	1	0.000000
0.141320	0.141490	0.141660	1	0.000000
0.141660	0.141830	0.142000	1	0.000000
0.142000	0.142170	0.142340	1	0.000000
0.142340	0.142510	0.142680	1	0.000000
0.142680	0.142850	0.143020	1	0.000000
0.143020	0.143190	0.143360	1	0.000000
0.143360	0.143530	0.143700	1	0.000000
0.143700	0.143870	0.144040	1	0.000000
0.144040	0.144210	0.144380	1	0.000000
0.144380	0.144550	0.144720	1	0.000000
0.144720	0.144890	0.145060	1	0.000000
0.145060	0.145230	0.145400	1	0.000000
0.145400	0.145570	0.145740	1	0.000000
0.145740	0.145910	0.146080	1	0.000000
0.146080	0.146250	0.146420	1	0.000000
0.146420	0.146590	0.146760	1	0.000000
0.146760	0.146930	0.147100	1	0.000000
0.147100	0.147270	0.147440	1	0.000000
0.147440	0.147610	0.147780	1	0.000000
0.147780	0.147950	0.148120	1	0.000000
0.148120	0.148290	0.148460	1	0.000000
0.148460	0.148630	0.148800	1	0.000000
0.148800	0.148970	0.149140	1	0.000000
0.149140	0.149310	0.149480	1	0.000000
0.149480	0.149650	0.149820	1	0.000000
0.149820	0.150000	0.150170	1	0.000000

INDEPENDENT VARIABLES WITH INTERNAL RANGES

CODE NUMBER = 116 MEAN = -0.102730-01 VAR = 0.156671D 01 INTERVAL = 0.300000D 00
 NO. REJECTS = 0 (MEAN) = C.0 (VAR) = 0.100000D 01 TOLERANCE = 0.300000D 01

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
0.125000	0.125170	0.125340	1	0.000000
0.125340	0.125510	0.125680	1	0.000000
0.125680	0.125850	0.126020	1	0.000000
0.126020	0.126190	0.126360	1	0.000000
0.126360	0.126530	0.126700	1	0.000000
0.126700	0.126870	0.127040	1	0.000000
0.127040	0.127210	0.127380	1	0.000000
0.127380	0.127550	0.127720	1	0.000000
0.127720	0.127890	0.128060	1	0.000000
0.128060	0.128230	0.128400	1	0.000000
0.128400	0.128570	0.128740	1	0.000000
0.128740	0.128910	0.129080	1	0.000000
0.129080	0.129250	0.129420	1	0.000000
0.129420	0.129590	0.129760	1	0.000000
0.129760	0.129930	0.130100	1	0.000000
0.130100	0.130270	0.130440	1	0.000000
0.130440	0.130610	0.130780	1	0.000000
0.130780	0.130950	0.131120	1	0.000000
0.131120	0.131290	0.131460	1	0.000000
0.131460	0.131630	0.131800	1	0.000000
0.131800	0.131970	0.132140	1	0.000000
0.132140	0.132310	0.132480	1	0.000000
0.132480	0.132650	0.132820	1	0.000000
0.132820	0.132990	0.133160	1	0.000000
0.133160	0.133330	0.133500	1	0.000000
0.133500	0.133670	0.133840	1	0.000000
0.133840	0.134010	0.134180	1	0.000000
0.134180	0.134350	0.134520	1	0.000000
0.134520	0.134690	0.134860	1	0.000000
0.134860	0.135030	0.135200	1	0.000000
0.135200	0.135370	0.135540	1	0.000000
0.135540	0.135710	0.135880	1	0.000000
0.135880	0.136050	0.136220	1	0.000000
0.136220	0.136390	0.136560	1	0.000000
0.136560	0.136730	0.136900	1	0.000000
0.136900	0.137070	0.137240	1	0.000000
0.137240	0.137410	0.137580	1	0.000000
0.137580	0.137750	0.137920	1	0.000000
0.137920	0.138090	0.138260	1	0.000000
0.138260	0.138430	0.138600	1	0.000000
0.138600	0.138770	0.138940	1	0.000000
0.138940	0.139110	0.139280	1	0.000000
0.139280	0.139450	0.139620	1	0.000000
0.139620	0.139790	0.139960	1	0.000000
0.139960	0.140130	0.140300	1	0.000000
0.140300	0.140470	0.140640	1	0.000000
0.140640	0.140810	0.140980	1	0.000000
0.140980	0.141150	0.141320	1	0.000000
0.141320	0.141490	0.141660	1	0.000000
0.141660	0.141830	0.142000	1	0.000000
0.142000	0.142170	0.142340	1	0.000000
0.142340	0.142510	0.142680	1	0.000000
0.142680	0.142850	0.143020	1	0.000000
0.143020	0.143190	0.143360	1	0.000000
0.143360	0.143530	0.143700	1	0.000000
0.143700	0.143870	0.144040	1	0.000000
0.144040	0.144210	0.144380	1	0.000000
0.144380	0.144550	0.144720	1	0.000000
0.144720	0.144890	0.145060	1	0.000000
0.145060	0.145230	0.145400	1	0.000000
0.145400	0.145570	0.145740	1	0.000000
0.145740	0.145910	0.146080	1	0.000000
0.146080	0.146250	0.146420	1	0.000000
0.146420	0.146590	0.146760	1	0.000000
0.146760	0.146930	0.147100	1	0.000000
0.147100	0.147270	0.147440	1	0.000000
0.147440	0.147610	0.147780	1	0.000000
0.147780	0.147950	0.148120	1	0.000000
0.148120	0.148290	0.148460	1	0.000000
0.148460	0.148630	0.148800	1	0.000000
0.148800	0.148970	0.149140	1	0.000000
0.149140	0.149310	0.149480	1	0.000000
0.149480	0.149650	0.149820	1	0.000000
0.149820	0.150000	0.150170	1	0.000000

CODE NUMBER = 117 MEAN = -0.255692D 00 VAR = 0.139304D 01 INTERVAL = 0.300000D 00
 NO. REJECTS = 0 (MEAN) = C.0 (VAR) = 0.100000D 01 TOLERANCE = 0.300000D 01

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
0.255000	0.255170	0.255340	1	0.000000
0.255340	0.255510	0.255680	1	0.000000
0.255680	0.255850	0.256020	1	0.000000
0.256020	0.256190	0.256360	1	0.000000
0.256360	0.256530	0.256700	1	0.000000
0.256700	0.256870	0.257040	1	0.000000
0.257040	0.257210	0.257380	1	0.000000
0.257380	0.257550	0.257720	1	0.000000
0.257720	0.257890	0.258060	1	0.000000
0.258060	0.258230	0.258400	1	0.000000
0.258400	0.258570	0.258740	1	0.000000
0.258740	0.258910	0.259080	1	0.000000
0.259080	0.259250	0.259420	1	0.000000
0.259420	0.259590	0.259760	1	0.000000
0.259760	0.259930	0.260100	1	0.000000
0.260100	0.260270	0.260440	1	0.000000
0.260440	0.260610	0.260780	1	0.000000
0.260780	0.260950	0.261120	1	0.000000
0.261120	0.261290	0.261460	1	0.000000
0.261460	0.261630	0.261800	1	0.000000
0.261800	0.261970	0.262140	1	0.000000
0.262140	0.262310	0.262480	1	0.000000
0.262480	0.262650	0.262820	1	0.000000
0.262820	0.262990	0.263160	1	0.000000
0.263160	0.263330	0.263500	1	0.000000
0.263500	0.263670	0.263840	1	0.000000
0.263840	0.264010	0.264180	1	0.000000
0.264180	0.264350	0.264520	1	0.000000
0.264520	0.264690	0.264860	1	0.000000
0.264860	0.265030	0.265200	1	0.000000
0.265200	0.265370	0.265540	1	0.000000
0.265540	0.265710	0.265880	1	0.000000
0.265880	0.266050	0.266220	1	0.000000
0.266220	0.266390	0.266560	1	0.000000
0.266560	0.266730	0.266900	1	0.000000
0.266900	0.267070	0.267240	1	0.000000
0.267240	0.267410	0.267580	1	0.000000
0.267580	0.267750	0.267920	1	0.000000
0.267920	0.2			

16533000 01	180000 01	0.180000 01	2.	0.113350 00
17000000 01	120000 01	0.120000 01	4.	0.151350 00
17500000 01	100000 01	0.100000 01	10.	0.178750 00
18000000 01	800000 00	0.800000 00	14.	0.189750 00
18500000 01	600000 00	0.600000 00	18.	0.197500 00
19000000 01	400000 00	0.400000 00	22.	0.204500 00
19500000 01	200000 00	0.200000 00	26.	0.211500 00
20000000 01	100000 00	0.100000 00	30.	0.218500 00
20500000 01	500000 00	0.500000 00	34.	0.225500 00
21000000 01	300000 00	0.300000 00	38.	0.232500 00
21500000 01	100000 00	0.100000 00	42.	0.239500 00
22000000 01	800000 00	0.800000 00	46.	0.246500 00
22500000 01	600000 00	0.600000 00	50.	0.253500 00
23000000 01	400000 00	0.400000 00	54.	0.260500 00
23500000 01	200000 00	0.200000 00	58.	0.267500 00
24000000 01	100000 00	0.100000 00	62.	0.274500 00
24500000 01	500000 00	0.500000 00	66.	0.281500 00
25000000 01	300000 00	0.300000 00	70.	0.288500 00
25500000 01	100000 00	0.100000 00	74.	0.295500 00
26000000 01	800000 00	0.800000 00	78.	0.302500 00
26500000 01	600000 00	0.600000 00	82.	0.309500 00
27000000 01	400000 00	0.400000 00	86.	0.316500 00
27500000 01	200000 00	0.200000 00	90.	0.323500 00
28000000 01	100000 00	0.100000 00	94.	0.330500 00
28500000 01	500000 00	0.500000 00	98.	0.337500 00
29000000 01	300000 00	0.300000 00	102.	0.344500 00
29500000 01	100000 00	0.100000 00	106.	0.351500 00
30000000 01	800000 00	0.800000 00	110.	0.358500 00
30500000 01	600000 00	0.600000 00	114.	0.365500 00
31000000 01	400000 00	0.400000 00	118.	0.372500 00
31500000 01	200000 00	0.200000 00	122.	0.379500 00
32000000 01	100000 00	0.100000 00	126.	0.386500 00
32500000 01	500000 00	0.500000 00	130.	0.393500 00
33000000 01	300000 00	0.300000 00	134.	0.400500 00
33500000 01	100000 00	0.100000 00	138.	0.407500 00
34000000 01	800000 00	0.800000 00	142.	0.414500 00
34500000 01	600000 00	0.600000 00	146.	0.421500 00
35000000 01	400000 00	0.400000 00	150.	0.428500 00
35500000 01	200000 00	0.200000 00	154.	0.435500 00
36000000 01	100000 00	0.100000 00	158.	0.442500 00
36500000 01	500000 00	0.500000 00	162.	0.449500 00
37000000 01	300000 00	0.300000 00	166.	0.456500 00
37500000 01	100000 00	0.100000 00	170.	0.463500 00
38000000 01	800000 00	0.800000 00	174.	0.470500 00
38500000 01	600000 00	0.600000 00	178.	0.477500 00
39000000 01	400000 00	0.400000 00	182.	0.484500 00
39500000 01	200000 00	0.200000 00	186.	0.491500 00
40000000 01	100000 00	0.100000 00	190.	0.498500 00
40500000 01	500000 00	0.500000 00	194.	0.505500 00
41000000 01	300000 00	0.300000 00	198.	0.512500 00
41500000 01	100000 00	0.100000 00	202.	0.519500 00
42000000 01	800000 00	0.800000 00	206.	0.526500 00
42500000 01	600000 00	0.600000 00	210.	0.533500 00
43000000 01	400000 00	0.400000 00	214.	0.540500 00
43500000 01	200000 00	0.200000 00	218.	0.547500 00
44000000 01	100000 00	0.100000 00	222.	0.554500 00
44500000 01	500000 00	0.500000 00	226.	0.561500 00
45000000 01	300000 00	0.300000 00	230.	0.568500 00
45500000 01	100000 00	0.100000 00	234.	0.575500 00
46000000 01	800000 00	0.800000 00	238.	0.582500 00
46500000 01	600000 00	0.600000 00	242.	0.589500 00
47000000 01	400000 00	0.400000 00	246.	0.596500 00
47500000 01	200000 00	0.200000 00	250.	0.603500 00
48000000 01	100000 00	0.100000 00	254.	0.610500 00
48500000 01	500000 00	0.500000 00	258.	0.617500 00
49000000 01	300000 00	0.300000 00	262.	0.624500 00
49500000 01	100000 00	0.100000 00	266.	0.631500 00
50000000 01	800000 00	0.800000 00	270.	0.638500 00
50500000 01	600000 00	0.600000 00	274.	0.645500 00
51000000 01	400000 00	0.400000 00	278.	0.652500 00
51500000 01	200000 00	0.200000 00	282.	0.659500 00
52000000 01	100000 00	0.100000 00	286.	0.666500 00
52500000 01	500000 00	0.500000 00	290.	0.673500 00
53000000 01	300000 00	0.300000 00	294.	0.680500 00
53500000 01	100000 00	0.100000 00	298.	0.687500 00
54000000 01	800000 00	0.800000 00	302.	0.694500 00
54500000 01	600000 00	0.600000 00	306.	0.701500 00
55000000 01	400000 00	0.400000 00	310.	0.708500 00
55500000 01	200000 00	0.200000 00	314.	0.715500 00
56000000 01	100000 00	0.100000 00	318.	0.722500 00
56500000 01	500000 00	0.500000 00	322.	0.729500 00
57000000 01	300000 00	0.300000 00	326.	0.736500 00
57500000 01	100000 00	0.100000 00	330.	0.743500 00
58000000 01	800000 00	0.800000 00	334.	0.750500 00
58500000 01	600000 00	0.600000 00	338.	0.757500 00
59000000 01	400000 00	0.400000 00	342.	0.764500 00
59500000 01	200000 00	0.200000 00	346.	0.771500 00
60000000 01	100000 00	0.100000 00	350.	0.778500 00

CODE NUMBER = 118 MEAN = -0.58295D 01 VAR = 0.72560D 04 INTERVAL = 0.27000D 02
 NO. REJECTS = 0.0 (MEAN) = 0.0 (VAR) = 0.81000D 04 TOLERANCE = 0.27000D 03

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
1.255000 01	0.250000 00	0.260000 00	0.	0.0
1.255000 01	0.260000 00	0.270000 00	0.	0.0
1.255000 01	0.270000 00	0.280000 00	0.	0.0
1.255000 01	0.280000 00	0.290000 00	0.	0.0
1.255000 01	0.290000 00	0.300000 00	0.	0.0
1.255000 01	0.300000 00	0.310000 00	0.	0.0
1.255000 01	0.310000 00	0.320000 00	0.	0.0
1.255000 01	0.320000 00	0.330000 00	0.	0.0
1.255000 01	0.330000 00	0.340000 00	0.	0.0
1.255000 01	0.340000 00	0.350000 00	0.	0.0
1.255000 01	0.350000 00	0.360000 00	0.	0.0
1.255000 01	0.360000 00	0.370000 00	0.	0.0
1.255000 01	0.370000 00	0.380000 00	0.	0.0
1.255000 01	0.380000 00	0.390000 00	0.	0.0
1.255000 01	0.390000 00	0.400000 00	0.	0.0
1.255000 01	0.400000 00	0.410000 00	0.	0.0
1.255000 01	0.410000 00	0.420000 00	0.	0.0
1.255000 01	0.420000 00	0.430000 00	0.	0.0
1.255000 01	0.430000 00	0.440000 00	0.	0.0
1.255000 01	0.440000 00	0.450000 00	0.	0.0
1.255000 01	0.450000 00	0.460000 00	0.	0.0
1.255000 01	0.460000 00	0.470000 00	0.	0.0
1.255000 01	0.470000 00	0.480000 00	0.	0.0
1.255000 01	0.480000 00	0.490000 00	0.	0.0
1.255000 01	0.490000 00	0.500000 00	0.	0.0
1.255000 01	0.500000 00	0.510000 00	0.	0.0
1.255000 01	0.510000 00	0.520000 00	0.	0.0
1.255000 01	0.520000 00	0.530000 00	0.	0.0
1.255000 01	0.530000 00	0.540000 00	0.	0.0
1.255000 01	0.540000 00	0.550000 00	0.	0.0
1.255000 01	0.550000 00	0.560000 00	0.	0.0
1.255000 01	0.560000 00	0.570000 00	0.	0.0
1.255000 01	0.570000 00	0.580000 00	0.	0.0
1.255000 01	0.580000 00	0.590000 00	0.	0.0
1.255000 01	0.590000 00	0.600000 00	0.	0.0
1.255000 01	0.600000 00	0.610000 00	0.	0.0
1.255000 01	0.610000 00	0.620000 00	0.	0.0
1.255000 01	0.620000 00	0.630000 00	0.	0.0
1.255000 01	0.630000 00	0.640000 00	0.	0.0
1.255000 01	0.640000 00	0.650000 00	0.	0.0
1.255000 01	0.650000 00	0.660000 00	0.	0.0
1.255000 01	0.660000 00	0.670000 00	0.	0.0
1.255000 01	0.670000 00	0.680000 00	0.	0.0
1.255000 01	0.680000 00	0.690000 00	0.	0.0
1.255000 01	0.690000 00	0.700000 00	0.	0.0
1.255000 01	0.700000 00	0.710000 00	0.	0.0
1.255000 01	0.710000 00	0.720000 00	0.	0.0
1.255000 01	0.720000 00	0.730000 00	0.	0.0
1.255000 01	0.730000 00	0.740000 00	0.	0.0
1.255000 01	0.740000 00	0.750000 00	0.	0.0
1.255000 01	0.750000 00	0.760000 00	0.	0.0
1.255000 01	0.760000 00	0.770000 00	0.	0.0
1.255000 01	0.770000 00	0.780000 00	0.	0.0
1.255000 01	0.780000 00	0.790000 00	0.	0.0
1.255000 01	0.790000 00	0.800000 00	0.	0.0
1.255000 01	0.800000 00	0.810000 00	0.	0.0
1.255000 01	0.810000 00	0.820000 00	0.	0.0
1.255000 01	0.820000 00	0.830000 00	0.	0.0
1.255000 01	0.830000 00	0.840000 00	0.	0.0
1.255000 01	0.840000 00	0.850000 00	0.	0.0
1.255000 01	0.850000 00	0.860000 00	0.	0.0
1.255000 01	0.860000 00	0.870000 00	0.	0.0
1.255000 01	0.870000 00	0.880000 00	0.	0.0
1.255000 01	0.880000 00	0.890000 00	0.	0.0
1.255000 01	0.890000 00	0.900000 00	0.	0.0
1.255000 01	0.900000 00	0.910000 00	0.	0.0
1.255000 01	0.910000 00	0.920000 00	0.	0.0
1.255000 01	0.920000 00	0.930000 00	0.	0.0
1.255000 01	0.930000 00	0.940000 00	0.	0.0
1.255000 01	0.940000 00	0.950000 00	0.	0.0
1.255000 01	0.950000 00	0.960000 00	0.	0.0
1.255000 01	0.960000 00	0.970000 00	0.	0.0
1.255000 01	0.970000 00	0.980000 00	0.	0.0
1.255000 01	0.980000 00	0.990000 00	0.	0.0
1.255000 01	0.990000 00	1.000000 00	0.	0.0

CODE NUMBER = 119 MEAN = -0.53956D -13 VAR = 0.71499D 04 INTERVAL = 0.27000D 02
 NO. REJECTS = 0.0 (MEAN) = 0.0 (VAR) = 0.81000D 04 TOLERANCE = 0.27000D 03

CLASS MARK	LOWER BOUND	UPPER BOUND	FREQ	DIST. FUNCT.
1.255000 01	0.250000 00	0.260000 00	0.	0.0
1.255000 01	0.260000 00	0.270000 00	0.	0.0
1.255000 01	0.270000 00	0.280000 00	0.	0.0
1.255000 01	0.280000 00	0.290000 00	0.	0.0
1.255000 01	0.290000 00	0.300000 00	0.	0.0
1.255000 01	0.300000 00	0.310000 00	0.	0.0
1.255000 01	0.310000 00	0.320000 00	0.	0.0
1.255000 01	0.320000 00	0.330000 00	0.	0.0
1.255000 01	0.330000 00	0.340000 00	0.	0.0
1.255000 01	0.340000 00	0.350000 00	0.	0.0
1.255000 01	0.350000 00	0.360000 00	0.	0.0
1.255000 01	0.360000 00	0.370000 00	0.	0.0
1.255000 01	0.370000 00	0.380000 00	0.	0.0
1.255000 01	0.380000 00	0.390000 00	0.	0.0
1.255000 01	0.390000 00	0.400000 00	0.	0.0
1.255000 01	0.400000 00	0.410000 00	0.	0.0
1.255000 01	0.410000 00	0.420000 00	0.	0.0
1.255000 01	0.420000 00	0.430000 00	0.	0.0
1.255000 01	0.430000 00	0.440000 00	0.	0.0
1.255000 01	0.440000 00	0.450000 00	0.	0.0
1.255000 01	0.450000 00	0.460000 00	0.	0.0
1.255000 01	0.460000 00	0.470000 00	0.	0.0
1.255000 01	0.470000 00	0.480000 00	0.	0.0
1.255000 01	0.480000 00	0.490000 00	0.	0.0
1.255000 01	0.490000 00	0.500000 00	0.	0.0
1.255000 01	0.500000 00	0.510000 00	0.	0.0
1.255000 01	0.510000 00	0.520000 00	0.	0.0
1.255000 01	0.520000 00	0.530000 00	0.	0.0
1.255000 01	0.530000 00	0.540000 00	0.	0.0
1.255000 01	0.540000 00	0.550000 00	0.	0.0
1.255000 01	0.550000 00	0.560000 00	0.	0.0
1.255000 01	0.560000 00	0.570000 00	0.	0.0
1.255000 01	0.570000 00	0.580000 00	0.	0.0
1.255000 01	0.580000 00	0.590000 00	0.	0.0
1.255000 01	0.590000 00	0.600000 00	0.	0.0
1.255000 01	0.600000 00	0.610000 00	0.	0.0
1.255000 01	0.610000 00	0.620000 00	0.	0.0
1.255000 01	0.620000 00	0.630000 00	0.	0.0
1.255000 01	0.630000 00	0.640000 00	0.	0.0
1.255000 01	0.640000 00	0.650000 00	0.	0.0
1.255000 01	0.650000 00	0.660000 00	0.	0.0
1.255000 01	0.660000 00	0.670000 00	0.	0.0
1.255000 01	0.670000 00	0.680000 00	0.	0.0
1.255000 01	0.680000 00	0.690000 00	0.	0.0
1.255000 01	0.690000 00	0.700000 00	0.	0.0
1.255000 01	0.700000 00	0.710000 00	0.	0.0
1.255000 01	0.710000 00	0.720000 00	0.	0.0
1.255000 01	0.720000 00	0.730000 00	0.	0.0
1.255000 01	0.730000 00	0.740000 00	0.	0.0
1.255000 01	0.740000 00	0.750000 00	0.	0.0
1.255000 01	0.750000 00	0.760000 00	0.	0.0
1.255000 01	0.760000 00	0.770000 00	0.	0.0
1.255000 01	0.770000 00	0.780000 00	0.	0.0
1.255000 01	0.780000 00	0.790000 00	0.	0.0
1.255000 01	0.790000 00	0.800000 00	0.	0.0
1.255000 01	0.800000 00	0.810000 00	0.	0.0
1.255000 01	0.810000 00	0.820000 00	0.	0.0
1.255000 01	0.820000 00	0.830000 00	0.	0.0
1.255000 01	0.830000 00	0.840000 00	0.	0.0
1.255000 01	0.840000 00	0.850000 00	0.	0.0
1.255000 01	0.850000 00	0.860000 00	0.	0.0
1.255000 01	0.860000 00	0.870000 00	0.	0.0
1.255000 01	0.870000 00	0.880000 00	0.	0.0
1.255000 01	0.880000 00	0.890000 00	0.	0.0
1.255000 01	0.890000 00	0.900000 00	0.	0.0
1.255000 01	0.900000 00	0.910000 00	0.	0.0
1.255000 01	0.910000 00	0.920000 00	0.	0.0
1.255000 01	0.920000 00	0.930000 00	0.	0.0
1.255000 01	0.930000 00	0.940000 00	0.	0.0
1.255000 01	0.940000 00	0.950000 00	0.	0.0
1.255000 01	0.950000 00	0.960000 00	0.	0.0
1.255000 01	0.960000 00	0.970000 00	0.	0.0
1.255000 01	0.970000 00	0.980000 00	0.	0.0
1.255000 01	0.980000 00	0.990000 00	0.	0.0
1.255000 01	0.990000 00	1.000000 00	0.	0.0

0.045000 02 0.410000 02 0.100000 03 9. 0.378760 02
0.121500 03 0.100000 03 0.135000 03 2. 0.417510 03
0.143500 03 0.135000 03 0.162000 03 1. 0.4208750 03
0.175500 03 0.162000 03 0.189000 03 1. 0.4208750 03
0.202500 03 0.189000 03 0.216000 03 1. 0.4208750 03
0.229500 03 0.216000 03 0.243000 03 1. 0.4208750 03
0.256500 03 0.243000 03 0.270000 03 0. 0.0

Figure 4-8y. Monte Carlo Mode Check Problem Output

Table 4-1 Monte Carlo Mode Output Interpretation

VARIABLE NAME	DESCRIPTION	COMMENTS
CODE Number	Independent/dependent variable identification number.	- See Table 3-3 for definitions. - CODE NUMBER list is identical to CODE# list defined by Card Group 2.
MEAN	Expected or mean value of the variable as determined by histogram.	See Eq. (B.1-22) for algorithm.
VAR	Variance of the variable as determined by the histogram.	See Eq. (B.1-24) for algorithm.
INTERVAL	Histogram cell size: $\text{INTERVAL} = 2 * \frac{\text{TOLERANCE}}{\text{NCELL}}$	- NCELL is the number of cells as defined by Card Group 1. - TOLERANCE is defined later in this table.
NO. REJECTS	Number of Monte Carlo experiments rejected from all histograms due to range limitations on this variable.	- Rejects occur only on dependent variables (i.e., TYPE = 7 or TYPE = 8) - See Appendix B.2.3, paragraph <u>Monte Carlo</u>, for a discussion of rejects.
(MEAN)	Defines midpoint of histogram. The value is generally related to the mean or expected value of the variable. However, it is subject to varying definitions. TYPE = 2, 3 Mean or expected value of variable as determined by VALUE on Group 2 card. TYPE = 4 Midpoint of abscissa table used to define the probability density function.	- For TYPE = 2 or TYPE = 3, the input mean value is repeated for reference purposes. - For TYPE = 4, the actual mean value may be determined from the LIMITING CASES printed output. VALUE (as given there) is the mean value determined by the probability density function.

Table 4-1 Monte Carlo Mode Output Interpretation (cont'd)

VARIABLE NAME	DESCRIPTION	COMMENTS
(MEAN) (cont'd)	TYPE = 7 The expected or mean value as determined by the Statistical Processor. TYPE = 8 Definition depends on histogram range: <u>-INTERNAL RANGES - TYPE = 7</u> definition applies. <u>-USER RANGES - Equal to VALUE</u> as defined by Group 2 Card.	- See Eq. (B.1-5) for algorithm. - Details of the calculation depend on the IOPRNT variable defined by Group 1 Card.
(VAR)	Value sometimes related to variance (σ^2) of the variable. However, it is subject to varying definitions. TYPE = 2, 3 Variance of variable as determined by STDEV on Group 2 Card: (VAR) = STDEV**2 TYPE = 4 Maximum tabulated value of probability density function.	- For TYPE = 2 or TYPE = 3, the input variance is repeated for reference purposes. - For TYPE = 4, the actual variance (σ^2) may be determined from the LIMITING CASES printed output. TOLERANCE (as defined there) is related to the standard deviation (σ) determined by the input probability density function according to $\sigma = 3 * TOLERANCE$ so that the variance is $\sigma^2 = 9 * TOLERANCE**2$ - Maximum value determined by search of probability density function table and used in generating random samples. See Appendix A.5.3 for full discussion.

Table 4-1 Monte Carlo Mode Output Interpretation (cont'd)

VARIABLE NAME	DESCRIPTION	COMMENTS
(VAR) (cont'd)	TYPE = 7 The variance (σ^2) as determined by the Statistical Processor. TYPE = 8 Definition depends on histogram range: -INTERNAL RANGES - TYPE = 7 definition applies. -USER RANGES - VALUE determined by STDEV value on Group 2 Card according to (VAR) = STDEV**2	- See Eq. (B.1-7) for algorithm. - Details of calculation depend on I0PRNT variable defined by Group 1 card.
TOLERANCE	Defines the histogram half range. Subject to varying definitions. TYPE = 2 The larger of 3*STDEV or TOL, both as defined on the Group 2 Card. TYPE = 3 The value of TOL as defined by the Group 2 Card. TYPE = 4 Half the interval covered by the probability density function abscissa table.	Normally TOL is selected to be small for the purposes of evaluating the derivatives. Thus, the half range of the histogram will usually be TOLERANCE = 3*STDEV

Table 4-1 Monte Carlo Mode Output Interpretation (concl'd)

VARIABLE NAME	DESCRIPTION	COMMENTS
TOLERANCE (cont'd)	TYPE = 7 The σ value where σ is defined by the Statistical Processor, i.e., $\sigma = \sqrt{\text{VAR}}$ TYPE = 8 Definition depends on histogram range: -INTERNAL RANGES - TYPE = 7 definition applies. -USER RANGES - TOL as defined by the Group 2 Card.	
CLASS MARK	Midpoint of histogram cells.	
LOWER BOUND	Left hand cell boundary.	
UPPER BOUND	Right hand cell boundary.	
FREQ.	Number of occurrences of cell during Monte Carlo experiments.	Total of FREQ Column is equal to the number of trials (NTRIAL) on Group 1 Card) less the number of rejects.
DIST.FUNCT.	Estimated probability density function value for this interval.	See Eq. (B.1-18) for computational algorithm.

5.0 SUMMARY

This report is a User's Manual for the Hypervelocity¹ Inflight Trajectory Scatter (HITS) computer code. The primary purpose of the code is to evaluate "projectile dispersion" which is defined here as the miss distance associated with inflight behavior not anticipated by Fire Control. Projectile dispersion poses a fundamental limit on overall weapon system effectiveness. The code is central to a scientific, systematic approach to evaluating projectile dispersion. The method consists of (1) constructing a comprehensive model for each source of projectile dispersion (i.e., the error source model) and (2) processing it through the HITS code to obtain a detailed projectile dispersion error budget, replete with sensitivity coefficients. The error budget identifies the larger dispersion sources and those with significant potential, as well as the total dispersion. Both crossrange and downrange dispersion are computed. The error budget defines dispersion on a level conducive to interpretation and a comprehensive understanding.

The HITS code evaluates projectile dispersion statistics by manipulating the inputs to a set of trajectory equations representing the true trajectory and comparing the computed flight path to the trajectory predicted by a second set of equations representing Fire Control. The true trajectory is varied in accord with the error source model; the Fire Control Trajectory is based on nominal projectile characteristics. At the selection of the user, variations are governed by either analytical or Monte Carlo statistical methods.

The two sets of trajectory equations are closed form approximations to the full six degree of freedom (6 DOF) equations of motion. They consist of (1) a particle trajectory with a velocity dependent drag coefficient and (2) two perturbation equations which correct for angle of attack effects. Extensive 6 DOF simulations have been conducted to verify the approximations. These equations also have potential application to operational Fire Control computers, the computation of firing tables, and any other situation requiring rapid, accurate, low-cost trajectory determination.

¹"Hypervelocity" is used throughout this report to indicate the limitation of the present HITS code to projectiles whose velocity does not become transonic at any point along the trajectory. No other connotation is intended.

The HITS code consists of two portions. One portion performs all input/output functions and the statistical calculations. This portion is an adaptation of existing AVCO software (i.e., YP-58). It brings to the HITS code additional capability as a projectile design tool; a capability the code would not otherwise have had. The closed form trajectory equations constitute the second portion of the code. These equations were specifically developed for dispersion assessment under an earlier contract.¹ The code was assembled under the earlier contract and used to evaluate the projectile dispersion of a reference design. The code, however, was developed only to the level of a research program and documented only to the extent necessary to support the analyses of the study. Detailed input/output information was not presented.

Under the present effort, the HITS code was advanced to the level of a production code. The code is maintained in the AVCO engineering computer code library as Production Code 5127. This report is the associated User's Manual. It places maximum emphasis on input/output information. The text contains all the day-to-day necessities required to operate the code and interpret the results of computation. Input/output information is summarized by tables featuring a quick-reference step-by-step format. Example problems are presented and discussed. They verify proper installation and illustrate the output format. The appendices discuss theoretical and programming aspects of the code. They contain the theoretical development of the closed form trajectory equations and a complete listing of the code.

¹Gustafson, T. G., Crimi, P., Bellaire, R. G., "Dispersion of Increased Velocity Projectiles - Feasibility Phase," Avco Corporation, AVSD-0200-75-CR, July 1975.

APPENDIX A

ELEMENTS OF PROBABILITY THEORY AND STATISTICS

This appendix presents an overview of the pertinent aspects of probability theory. Sections A.1 through A.5 review basic concepts and theoretical relationships. Sections A.5 and A.6 relate the theory to the practical problems associated with Monte Carlo methods. More detailed treatments may be found in text books.^{1,2}

A.1 Random Phenomenon and Probabilities

Random phenomenon are those phenomena whose fundamental processes are either not completely understood or are too complex to define in an entirely satisfactory manner. In either case, the scientific interest in the phenomenon centers around the "average" rather than the "precise" outcome. Probability theory is a mathematical discipline for quantifying random phenomenon.

A.1.1 Fundamental Concepts

Probability theory has three building blocks: (1) the sample description space, (2) a collection of events, and (3) a probability function. This section discusses these concepts.

Sample Description Space

The sample description space is the collection of all possible outcomes of the random phenomena. Each occurrence is thought of as the result of an experiment. For instance, the sample description space for a coin flipping experiment would consist of the two possible outcomes (H, T). The sample description space for a temperature measurement taken at randomly selected time and location would include all values from $-\infty$ to $+\infty$, $(-\infty, +\infty)$.

Events

Random events (or simply "Events") are the groupings of the fundamental outcomes. They are subsets of the sample description space. Events are defined by the analyst. For instance, a meteorologist interested in the likelihood of sub-freezing temperatures would

¹Parzen, E., Modern Probability Theory and its Applications, Wiley, New York, 1960.

²Davenport, W. B., and Root, W. L., An Introduction to the Theory of Random Signals and Noise, McGraw-Hill, New York, 1958.

select only two events to subdivide the sample description space of all possible temperatures: $(-\infty, 32)$ and $(32, +\infty)$.

Probability Function

The probability function, P , assigns to each event a number between zero and one corresponding to the relative likelihood of occurrence of the event. The probability of an event, A , is the mathematical ratio of the number of times it would occur, N , to the total number of experiments, N_T .

$$P[A] = \frac{N}{N_T} \quad (A.1-1)$$

(Monte Carlo procedures estimate probabilities by counting the results of experiments.) An event which (almost) never occurs has a probability of zero. An event which (almost) always occurs has a probability of one. For instance, if freezing temperatures occurred 40% of the time the probability of that event would be

$$P[(-\infty, 32)] = 0.40 \quad (A.1-2)$$

Venn Diagrams

A Venn diagram is a graphical method of depicting the relationship between the sample description space and events. (It also gives a convenient method for illustrating the rules of combination for probabilities.) Figure A-1 shows a Venn diagram. There are three events A , B , and C . Events A and B overlap so A and B simultaneously occur for some experimental outcomes. Event C is exclusive of A and B , since neither A nor B can occur when C occurs.

A.1.2 Rules of Combination

Let the probabilities of the individual events in Figure A-1 be $P(A)$, $P(B)$, and $P(C)$. There are rules of combination. Let $A + B$ denote the occurrence of either event, and let AB denote the simultaneous occurrence of both events. The probability of $A + B$ is

$$P(A + B) = P(A) + P(B) - P(AB) \quad (A.1-3)$$

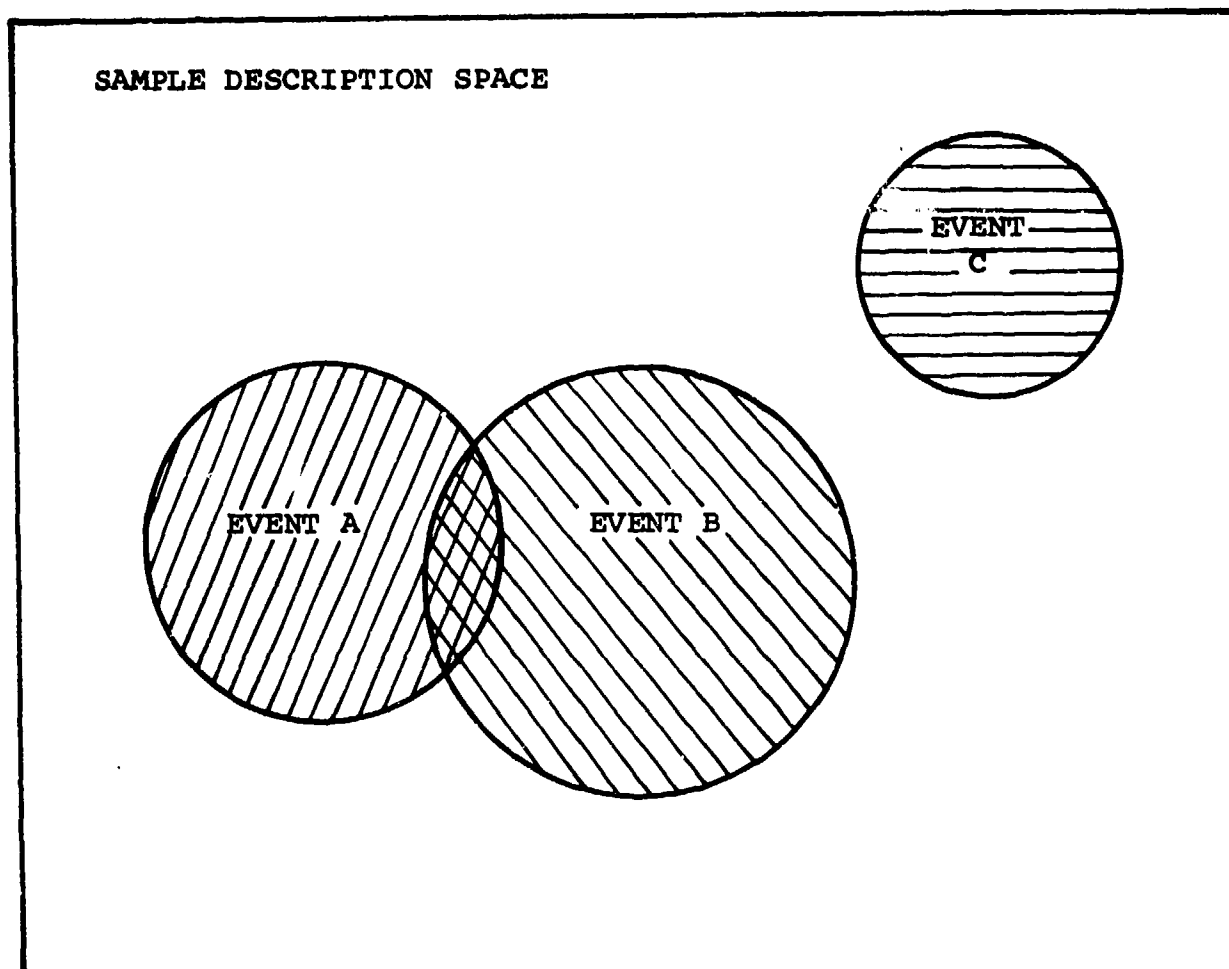


FIGURE A-1 THE VENN DIAGRAM

Referring to Figure A-1, $A + B$ is the area encompassed by either events A or B, and AB is the area common to both A and B. With these visualizations, Eq. (A.1-3) simply states that $P(A+B)$ is equal to $P(A) + P(B)$ less a correction, $P(AB)$, for the common area which is counted twice in the sum $P(A) + P(B)$.

Since probabilities are positive numbers, Eq. (A.1-3) implies

$$P(A + B) \leq P(A) + P(B) \quad (\text{A.1-4})$$

Considering a collection of events: A, B, C, ... with the probabilities $P(A)$, $P(B)$, $P(C)$..., the probability that at least one of the events will occur is $P(A + B + C + \dots)$ which is related to the individual probabilities through the inequality:

$$P(A + B + C + \dots) \leq P(A) + P(B) + P(C) + \dots \quad (\text{A.1-5})$$

It is an inequality since it may be possible for at least two of the events to occur concurrently. If only one of the events can occur the events are said to be mutually exclusive and equality holds. If at least one of the events must occur, the ensemble of events is termed exhaustive in which case the left side of the above inequality is identically one.

A.1.3 Conditional Probabilities

Considering two events, A and B, the probability of their simultaneous or joint occurrence is $P(AB)$. The probability $P(A/B)$ is the conditional probability of A given B and is defined by:

$$P(A/B) = \frac{P(AB)}{P(B)} \quad (\text{A.1-6})$$

provided $P(B) > 0$. The conditional probability $P(A/B)$, is the probability A will occur given that B has occurred. Two events, A and B, are said to be statistically independent if knowledge that B has occurred is of no value in inferring that A will occur so $P(A/B) = P(A)$. This means

$$P(ABC\dots) = P(A)P(B)P(C)\dots \quad (\text{A.1-7})$$

if A, B, C, ... are mutually independent events. That is, for statistically independent events, the probability of simultaneous occurrence is equal to the product of the individual probabilities.

A.2 Random Variables

The discussion of the previous section dealt with random phenomena in general. The discussion from this point on is restricted to random variables, i.e., those whose outcomes are numerically valued. The outside temperature, the instantaneous voltage across a resistor, and the displacement of the impact point from the target point are just three examples of random variables.

A.2.1 Distribution Functions

A numerically valued random phenomenon is one which is completely summarized by the value of a descriptive variable (i.e., temperature, voltage, or misdistance in the foregoing examples). The variable is referred to as a random or stochastic variable. The sample description space is all real numbers, and the events are all intervals and combinations of intervals. The probability function is defined by the (probability) distribution function, F ,

$$F(X) = P(x \leq X)$$

(A.2-1)

The distribution function evaluated at X is the probability that the random variable x has a value less than or at the most equal to X . By definition, the cumulative distribution function has the properties $F(-\infty) = 0$, $F(+\infty) = 1$ and $F(x)$ is a positive, non-decreasing function of x . (Note: It is customary to notationally distinguish between a random variable, x , and a value it could assume, X , only when demanded by clarity.) A distribution function need not be a continuous function. It may have step-type discontinuities at discrete points corresponding to values with a positive probability of occurrence. Random variables may be "discrete," "continuous," or "mixed." The remainder of this section discusses these types.

A discrete random variable can take on specific discrete values $X_1, X_2, \dots, X_n$. The numerical value shown on a pair of dice is an

example of a discrete random variable. Each experiment must result in an integer value between two and twelve. If f_i is the probability, X_i occurs, the distribution function is given as

$$F(X) = \sum_{X_i \leq X} f_i = P(x \leq X) \quad (A.2-2)$$

and

$$F(X_N) = \sum_{i=1}^N f_i = 1 \quad (A.2-3)$$

If the distribution function derivative exists, the random variable is said to be continuous and the (probability) density function is defined according to

$$f(x) = \frac{d}{dx} F(x) \quad (A.2-4)$$

The density function at X , $f(X)$, is the probability of the event that the random variable x falls in the range

$$X \leq x \leq X + dX \quad (A.2-5)$$

The probability of occurrence of any specific value of a continuous random variable is zero, since

$$\lim_{\epsilon \rightarrow 0} P[X \leq x \leq X + \epsilon] = \lim_{\epsilon \rightarrow 0} \int_X^{X+\epsilon} f(x) dx \quad (A.2-6)$$

Air temperature is an example of a continuous random variable since the likelihood of any specific temperature is zero. As a result of the definitions

$$\int_{-\infty}^{\infty} f(x) dx = F(\infty) - F(-\infty) = 1 \quad (A.2-7)$$

The distribution of a mixed random variable contains both discrete values, with positive probability of occurrence, and a continuous distribution.

Most random variables treated by Monte Carlo methods are continuous. The density function is estimated by dividing up the range of the random variable into small intervals called cells. Histograms are then constructed which count the number of times the value of the random variable falls in each cell, N_i . The exact value is discarded and only the histogram is maintained during execution of the code. The density function is estimated as a piece-wise constant function by the relationship

$$f(x) = \frac{N_i}{N_T} \frac{1}{x_{b_{i+1}} - x_{b_i}} \quad (\text{A.2-8})$$

where N_T is the total number of trials and $x_{b_{i+1}}$ and x_b are the upper and lower boundaries of the i^{th} cell. It might be argued that Monte Carlo methods, in effect, approximate all random variables as discrete since only the histogram data is saved. Nevertheless, it is most convenient to treat only continuous random variables during the theoretical development. Equation (A.2-8) is all that need be recalled to make the translation back to the real world of Monte Carlo analysis.

A.2.2 Joint Distribution Functions

It is often necessary to describe two or more random variables in relation to each other. These situations are handled by treating each of the random variables x_i as an element of a random vector $\underline{x}$. The joint (probability) distribution function is defined as

$$F(\underline{X}) = P(x_1 \leq X_1, x_2 \leq X_2, \dots, x_n \leq X_N) \quad (\text{A.2-9})$$

From the definition, the following limits hold

$$\lim_{\substack{x_1 \rightarrow \infty \\ \vdots \\ x_n \rightarrow \infty}} F(\underline{x}) = 1 \quad (\text{A.2-10})$$

$$\begin{aligned}
& \lim_{x_1 \rightarrow \infty} F(\underline{x}) = F_1(x_1) \\
& \vdots \\
& x_{i-1} \rightarrow \infty \\
& x_{i+1} \rightarrow \infty \\
& \vdots \\
& x_n \rightarrow \infty
\end{aligned}
\tag{A.2-11}$$

where $F_1(x_1)$ is the distribution function of the i^{th} random variable. The joint (probability) density function is defined as

$$\frac{\partial^n}{\partial x_1 \partial x_2 \dots \partial x_n} F(\underline{x}) = f(\underline{x})
\tag{A.2-12}$$

A.2.3 Statistical Independence

Random variables are said to be statistically independent whenever the joint distribution functions factors such that

$$F(\underline{x}) = \prod_{i=1}^n F_i(x_i)
\tag{A.2-13}$$

or equivalently

$$f(\underline{x}) = \prod_{i=1}^n f_i(x_i)
\tag{A.2-14}$$

where F_i is the distribution function and f_i is the density function of the random variable x_i . Statistically independent random variables are independent in the sense that the value of one does not affect the value of another, and the definition is equivalent to the one given earlier for statistically independent random events.

A.3 Averages

Averages are of great practical importance. These statistics capture a random variable's pertinent characteristics: its average

value and quantitative assessments of randomness. Section A.3.1 discusses the mathematical operation of expectation, which is the means for computing averages. Specific averages (i.e., the moments) are discussed in Section A.3.2. Section A.3.3 discusses an important average used to assess the correlation between two random variables. Confidence limits are discussed in Section A.3.4.

A.3.1 Expectation

If $g(x)$ is any function of the random variable x , the expectation (average value) of $g(x)$ is denoted by $E[g(x)]$. For a discrete random variable

$$E[g(x)] = \sum_{i=1}^n g(X_i) f_i \quad (\text{A.3-1})$$

where f_i is the probability of occurrence of X_i . In the case of a continuous random variable

$$E[g(x)] = \int_{-\infty}^{\infty} g(x) f(x) dx \quad (\text{A.3-2})$$

where f is the density function. The development presented here considers only continuous random variables. However, it should be kept in mind that completely analogous formulas exist for all distribution types as illustrated by Eq. (A.3-1).

The expectation of a function of a random variable, as given by Eq. (A.3-2), is a weighted average of the function values with the weights determined by the distribution of the random variable. Expectation is a linear operator, which obeys

$$E \left[\sum_i a_i g_i(x_i) \right] = \sum_i a_i E[g_i(x_i)] \quad (\text{A.3-3})$$

where a_i is any non-random number. The distributive law of multiplication,

$$E \left[\prod_i g_i(x_i) \right] = \prod_i E[g_i(x_i)] \quad (\text{A.3-4})$$

holds only if the variables are statistically independent. The relation

$$E[g(x)] = g(E[x]) \quad (\text{A.3-5})$$

is generally not true either.

The expectation of a vector valued function $g(\underline{x})$ is defined as

$$E[g(\underline{x})] = \int_{-\infty}^{\infty} g(\underline{x}) f(\underline{x}) d\underline{x} \quad (\text{A.3-6})$$

where $f(\underline{x})$ is the joint probability density function and integration is with respect to all variables. The expectation operator is a linear operator with respect to vector random variables.

A.3.2 Moments

The quantity $E[x^k]$, is called the k^{th} moment of x

$$E[x^k] = \int_{-\infty}^{\infty} x^k f(x) dx \quad (\text{A.3-7})$$

In particular with $k = 1$, $E[x] = \bar{x}$ is the mean value of the variable x . The quantity defined by $E[(x - \bar{x})^k]$ is called the k^{th} central moment of x . Moments and central moments are related. For example, the second central moment given as $E[(x - \bar{x})^2]$ can be expressed in terms of the first two moments

$$E[(x - \bar{x})^2] = E[x^2] - \bar{x}^2 \quad (\text{A.3-8})$$

by virtue of the fact that $E[x] = \bar{x}$ and the linear property of the expectation operator.

Several of the low order moments have practical interpretations. By definition, the first central moment is zero. The second central moment given as $E[(x - \bar{x})^2]$ is called the variance of x and is denoted σ^2 . It is a measure of dispersion of x about the mean. The square root of the variance is called the standard deviation and denoted by σ .

Based on their frequency of general usage, the mean and variance are by far the most important moments. Occasionally, for non-zero mean random variables a coefficient of variation is defined

$$C_\sigma = \frac{\sigma}{\bar{x}} \quad (\text{A.3-9})$$

Of less frequent usage are the third and fourth central moments. The third central moment provides a measure of the asymmetry of the distribution about the mean value and is referred to as skewness. The fourth central moment is referred to as kurtosis and provides an additional measure of the clustering of the distribution about its mean value.

A.3.3 Correlation

Let x and y be random variables with mean values $\bar{x}$ and $\bar{y}$. The expectation of the product

$$E[(x - \bar{x})(y - \bar{y})] = \iint_{-\infty}^{\infty} (x - \bar{x})(y - \bar{y}) f(x, y) dx dy \quad (\text{A.3-10})$$

is called the covariance of x and y , $\text{Cov}(x, y)$. If x and y are statistically independent, in accordance with (A.2-4), then

$$\text{Cov}(x, y) = E[x - \bar{x}] E[y - \bar{y}] = 0 \quad (\text{A.3-11})$$

The converse is not true. That is, in general, the vanishing of the covariance is not sufficient to insure statistical independence. Whenever the covariance is zero, the variables are said to be "uncorrelated."

Frequently a correlation coefficient is used rather than the covariance. The correlation coefficient is defined as

$$\rho_{xy} = \frac{\text{Cov}(x, y)}{\sigma_x \sigma_y} \quad (\text{A.3-12})$$

where the σ_x and σ_y are the standard deviations of x and y . It can be shown that the correlation coefficient always has values in the range -1 to 1 . Again, a non-zero value of the correlation coefficient implies statistical dependence but a vanishing correlation coefficient does not imply statistical independence.

The interrelationship between two random phenomena, x and y , is describable to first order by the correlation coefficient, ρ_{xy} . The correlation coefficient is most clearly understood by an example. One method of assessing the presence of correlation between phenomenon y and phenomenon x would be to plot y versus x . Any clustering of data points in the resulting "scatter diagram" would indicate correlation. The correlation can be quantified by fitting a straight line

$$y = ax \quad (\text{A.3-13})$$

to the data (assuming zero means) using least squares techniques to determine a . The result would be the best linear prediction formula for y given x . This formula is referred to as the regression line. The regression line formula can be written in terms of the standard deviations and correlation coefficient

$$\frac{\hat{y}}{\sigma_y} = \rho_{xy} \frac{x}{\sigma_x} \quad (\text{A.3-14})$$

Equation (A.3-14) shows that the correlation coefficient is the slope of the best-fit straight line when the data has been normalized with respect to the standard deviations. Since both x and y are random, the data will be scattered about the regression line. The scatter can be quantified by calculating the root mean square error

$$\text{rms} = \sqrt{\sum_{i=1}^n [y_i - \hat{y}(x_i)]^2} \quad (\text{A.3-15})$$

where i denotes the individual data points. The rms error can be shown to be

$$\text{rms} = \sqrt{1 - \rho_{xy}^2} \sigma_y \quad (\text{A.3-16})$$

Thus the correlation coefficient is a measure of the scatter about the regression line. If ρ_{xy} is zero, the rms error is equal to the uncertainty in the data itself, σ_y . Thus uncorrelated random variables (i.e., $\rho_{xy} = 0$) are completely random relative to each other and no first order interrelationship exists. Since the rms error must be positive, Eq. (A.3-16) requires

$$-1 \leq \rho_{xy} \leq 1 \quad (\text{A.3-17})$$

If $\rho_{xy} = \pm 1$ then the rms error is zero, all the data falls on the regression line, y depends linearly on x , and y is not random relative to x .

A.3.4 Confidence Limits

Confidence limits bound the uncertainty in the value of a random variable. The limits are selected to insure the random variable will fall inside, with a certain level of confidence. Since a random variable x is expected to take on its average value $\bar{x}$, the K -confidence limit, a_k , is defined as the value for which

$$P[|x - \bar{x}| \leq a_k] = K \quad (\text{A.3-18})$$

This defines a_k in terms of K and the distribution. The random variable will fall inside the interval

$$\bar{x} - a_k \leq x \leq \bar{x} + a_k \quad (\text{A.3-19})$$

with a relative frequency of K .

The Chebyshev inequality

$$P[|\bar{x} - x| \leq h\sigma] \geq 1 - \frac{1}{h^2} \quad (\text{A.3-20})$$

is plotted in Figure A-2 along with other representative distribution types. It can be seen to be a conservative lower bound. Substituting Eq. (A.3-20) into (A.3-18) and solving for the confidence limit yields

$$a_k \leq \frac{\sigma}{\sqrt{1 - K}} \quad (\text{A.3-21})$$

Thus a random variable always has a value within 2σ of its mean 75% of the time and within 3σ of its mean 89% of the time. This illustrates that the standard deviation, σ , quantifies randomness regardless of the distribution.

A.4 Typical Distributions

It is generally difficult to precisely determine the distribution of a random variable. However, it is usually possible to assess characteristic properties which make it possible to select a distribution model. Presented here are four commonly used distribution types.

A.4.1 Uniformly Distributed Random Variable (TYPE = 3)

A uniformly distributed random variable is equally likely to take on any value in the interval $\bar{x} - T$ to $\bar{x} + T$. The density function is

$$f(x) = \begin{cases} 0 & |x - \bar{x}| > T \\ \frac{1}{2T} & |x - \bar{x}| \leq T \end{cases} \quad (\text{A.4-1})$$

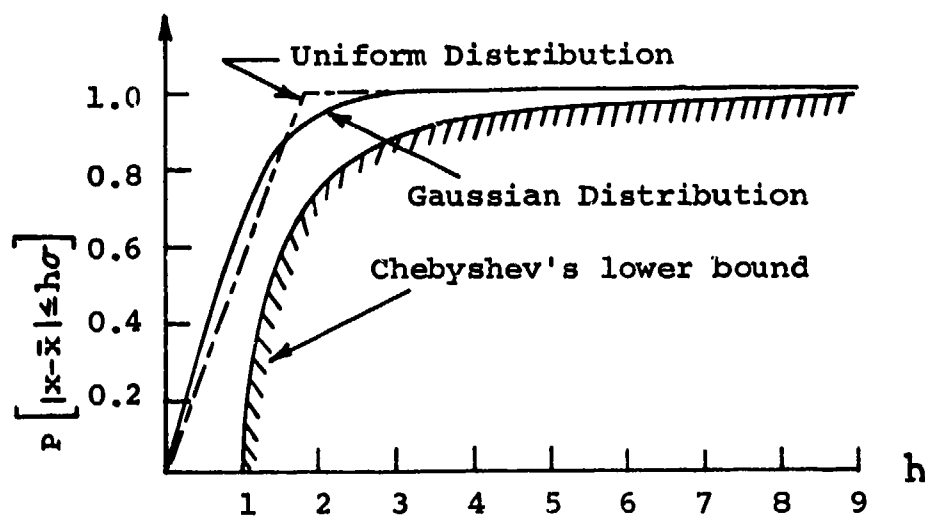


FIGURE A-2 CHEBYSHEV'S LOWER BOUND

The mean value is $\bar{x}$; T is referred to as the tolerance, and

$$\sigma = \frac{T}{\sqrt{3}} \quad (\text{A.4-2})$$

A uniform distribution is characteristic of error sources whose magnitude is limited by quality control procedures. Projectile length is a good example. Uncertainties in projectile length can be easily controlled by measuring each projectile after final machining. Those exceeding quality control limits would be discarded. Thus, off-nominal lengths beyond these limits do not occur. Within these limits, the length is equally likely to be any value.

A.4.2 Gaussian Distributed Random Variable (TYPE = 2)

A Gaussian distributed random variable is one whose distribution about the mean has the familiar bell shape. The density function is

$$f(x) = \frac{1}{\sqrt{2\pi} \sigma} e^{-\frac{1}{2} \left(\frac{x - \bar{x}}{\sigma} \right)^2} \quad (\text{A.4-3})$$

where $\bar{x}$ is the mean value and σ is the standard deviation. Figure A-3 illustrates the Gaussian curve. Since the interval $\bar{x} \pm 3\sigma$ contains better than 99 percent of the outcomes, the tolerance T is defined as

$$T = 3\sigma \quad (\text{A.4-4})$$

The effect of a change in value of σ is indicated in Figure A-4. Increasing σ effects a simultaneous lowering of the peak value and a spreading of the tails of the curve. Changes in the mean value $\bar{x}$ results in a translation of the curve.

The Gaussian form is particularly interesting when sums of random variables are considered. For example, suppose there is a linear functional relation

$$z = ax + by \quad (\text{A.4-5})$$

where a, b are constants and x, y are random variables. The

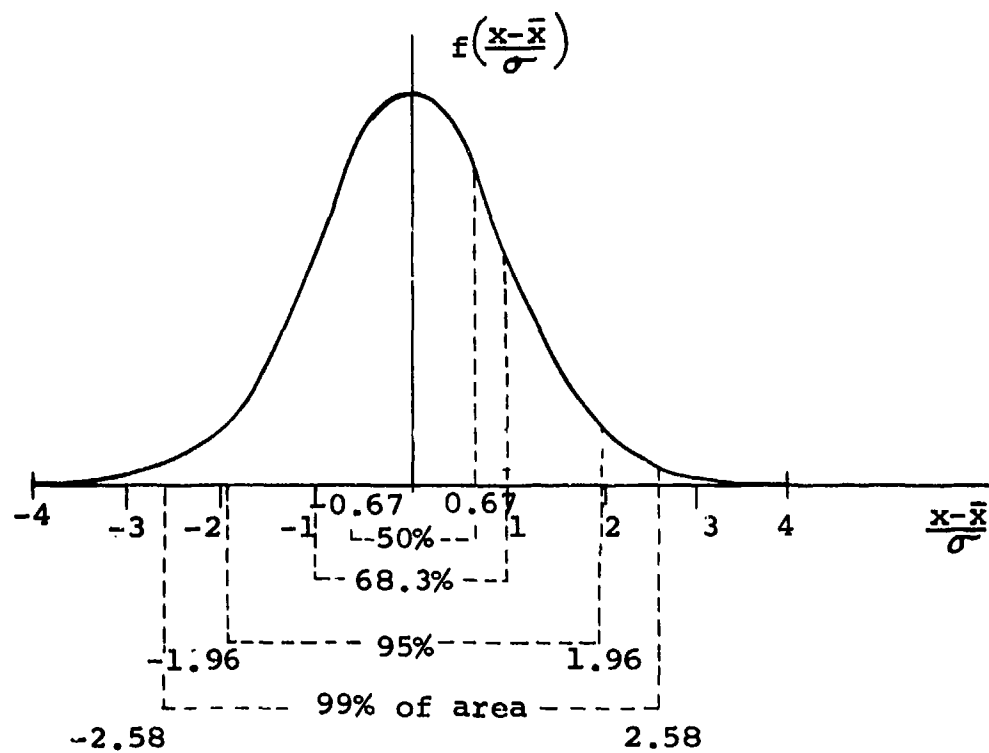


FIGURE A-3 GAUSSIAN DENSITY FUNCTION

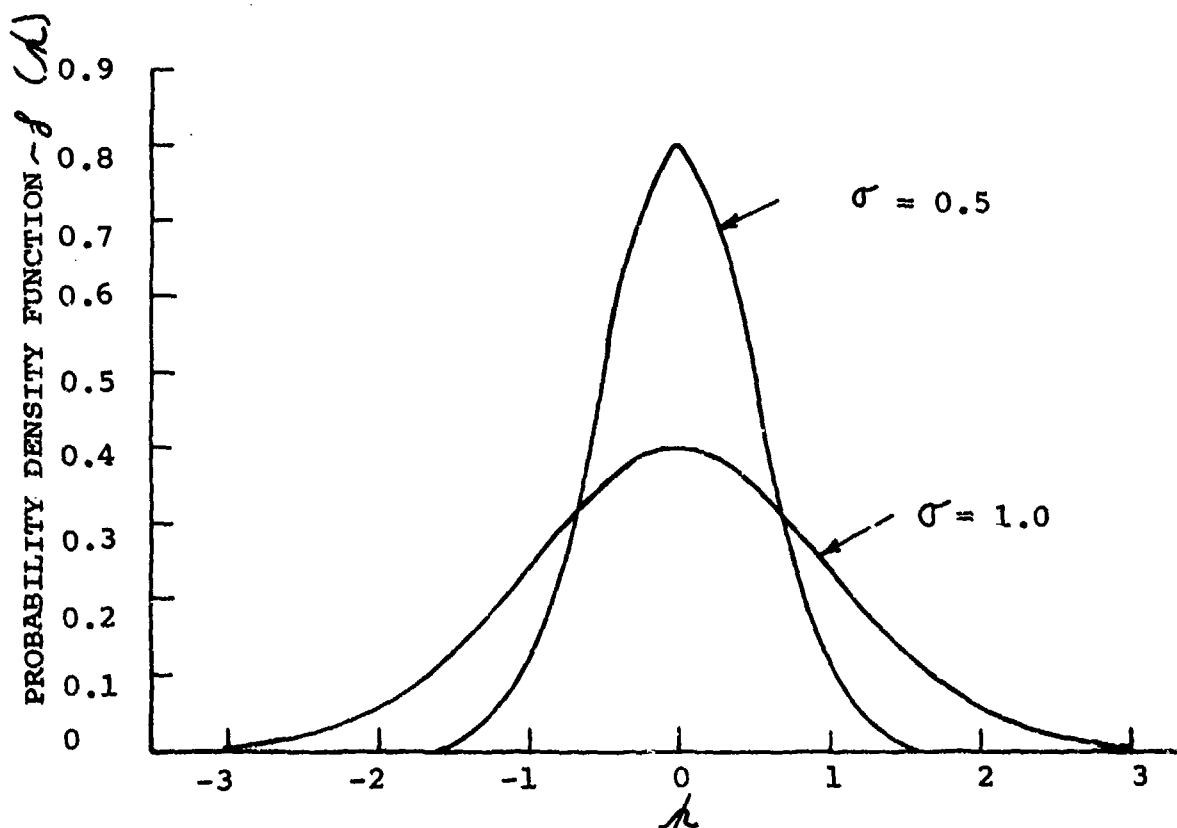


FIGURE A-4 A COMPARISON OF TWO GAUSSIAN DENSITY FUNCTIONS WITH DIFFERENT STANDARD DEVIATIONS

variable z , by virtue of being a function of random variables, is also a random variable. If x and y are independent then one can conclude that

$$\bar{z} = a\bar{x} + b\bar{y}$$

$$\sigma_z^2 = a^2 \sigma_x^2 + b^2 \sigma_y^2 \quad (\text{A.4-6})$$

by application of the principles previously presented. Now in general, the knowledge of the mean value of z and its variance are insufficient to infer the distribution of z . For example, Figure A-5 illustrates two different density functions which have the same mean and variance. However, it can be shown that if x and y are Gaussian distributed then z will also be Gaussian. This result can be extended to linear combinations of any number of Gaussian random variables.

The Central Limit Theorem states that under suitable conditions the sum of arbitrarily distributed independent random variables will become Gaussian as the number of variables becomes large. The necessary condition is that no single term of the sum can dominate. That is, the variance of any one term must not be of the same order as the sum of the variance of the other terms. For instance, the sum of twenty random variables, which are uniformly distributed between 0 and 1,

$$z = \frac{1}{20} \sum_{i=1}^{20} x_i \quad (\text{A.4-7})$$

is very nearly Gaussian.

In summary, a Gaussian uncertainty is one whose distribution about the mean has the familiar bell shape. It is appropriate for effects formed from a large number of independent random occurrences. For instance, variations in muzzle velocity are the result of a large number of independent effects occurring while the projectile traverses the barrel. Thus, the distribution of muzzle velocity would be expected to be Gaussian.

A.4.3 Jointly Gaussian Random Variables

Let x and y be jointly distributed Gaussian random variables. Their joint density function is

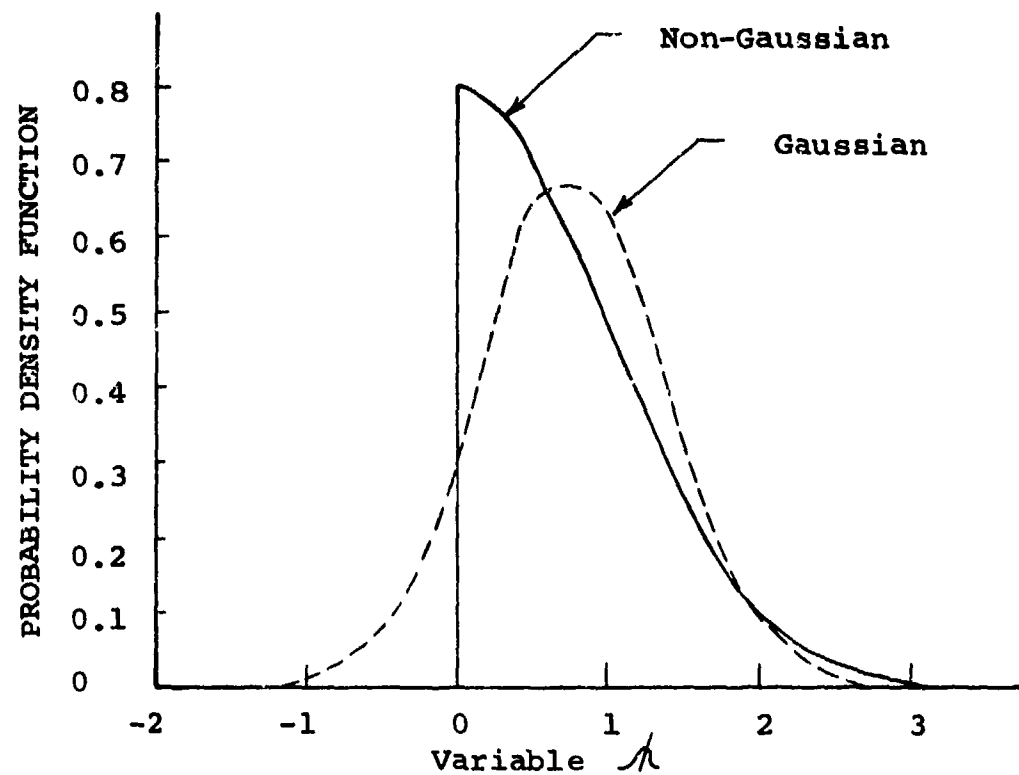


FIGURE A-5 TWO PROBABILITY DENSITY FUNCTIONS WITH IDENTICAL MEANS AND VARIANCES

$$f(x, y) = \frac{1}{2\pi\sigma_x\sigma_y(1-\rho_{xy}^2)^{1/2}} \text{Exp} \left[-\frac{\tilde{x}^2 - 2\rho_{xy}\tilde{x}\tilde{y} + \tilde{y}^2}{2(1-\rho_{xy}^2)} \right] \quad (\text{A.4-8})$$

where

$$\tilde{x} = \frac{x - \bar{x}}{\sigma_x} \quad (\text{A.4-9})$$

$$\tilde{y} = \frac{y - \bar{y}}{\sigma_y} \quad (\text{A.4-10})$$

and $\bar{x}$ and $\bar{y}$ are the mean values of x and y , σ_x and σ_y are the standard deviations of x and y , and ρ_{xy} is the correlation coefficient of x and y . Considered separately, x and y are Gaussian distributed. If the correlation coefficient is zero, the density function factors and x and y are statistically independent. Thus, uncorrelated Gaussian random variables are statistically independent.

Six indices are commonly used to summarize the randomness of two jointly distributed random variables. The standard deviation, σ , has already been described. There is a 50% probability the point (x, y) will lie between two parallel lines which are equidistant from the origin and are separated by twice the Linear Error Probable (LEP). The Circular Error Probable (CEP) is the radius of the circle with a 50% probability of occurrence,

$$P\left[\sqrt{x^2 + y^2} \leq \text{CEP}\right] = 0.5 \quad (\text{A.4-11})$$

The radius of the 80% circle is denoted R_{80} . The mean radius $\bar{J}$ is the average displacement of the point (x, y) from the origin

$$\bar{J} = E\left[\sqrt{x^2 + y^2}\right] \quad (\text{A.4-12})$$

(If x and y are the cross range deflections expressed as a fraction of the range, J is referred to as the "average jump angle".) The Radial Standard Deviation (RSD) is the rms value of x and y

$$\text{RSD} = \sqrt{E[x^2 + y^2]} \quad (\text{A.4-13})$$

If x and y are jointly distributed Gaussian random variables, a readily available text book¹ gives a detailed development of the analytic expressions relating the CEP to σ_x , σ_y and ρ_{xy} . The CEP is closely approximated by

$$\text{CEP} = 0.589 (\sigma_x + \sigma_y) \quad (\text{A.4-14})$$

when ρ_{xy} is zero and σ_x and σ_y differ by no more than 80% of the larger. Provided σ_x and $\sigma_y = \sigma$ and $\rho_{xy} = 0$, all four indices are proportional to each other as indicated in Figure A-6. Each of these statistics may be used to define a circle which have the probabilities of occurrence as shown in Table A-1.

The above discussion has dealt exclusively with two jointly distributed Gaussian random variables. It is possible to have an arbitrarily large number of jointly distributed Gaussian random variables: $x_1, x_2, \dots, x_n$. The joint density function is defined elsewhere.² The distribution function is completely determined by the $N \times N$ covariance matrix

$$\text{Cov}(\underline{x}) = \begin{bmatrix} \sigma_{x_1}^2 & \text{Cov}(x_1, x_2) & \dots \\ \text{Cov}(x_2, x_1) & \sigma_{x_2}^2 & \dots \\ \vdots & \vdots & \ddots \end{bmatrix} \quad (\text{A.4-15})$$

¹Pitman, G. F., Inertial Guidance, Wiley, New York, 1962

²Davenport, N. B., and Root, W. L., An Introduction to the Theory of Random Signals and Noise, McGraw-Hill, New York, 1958.

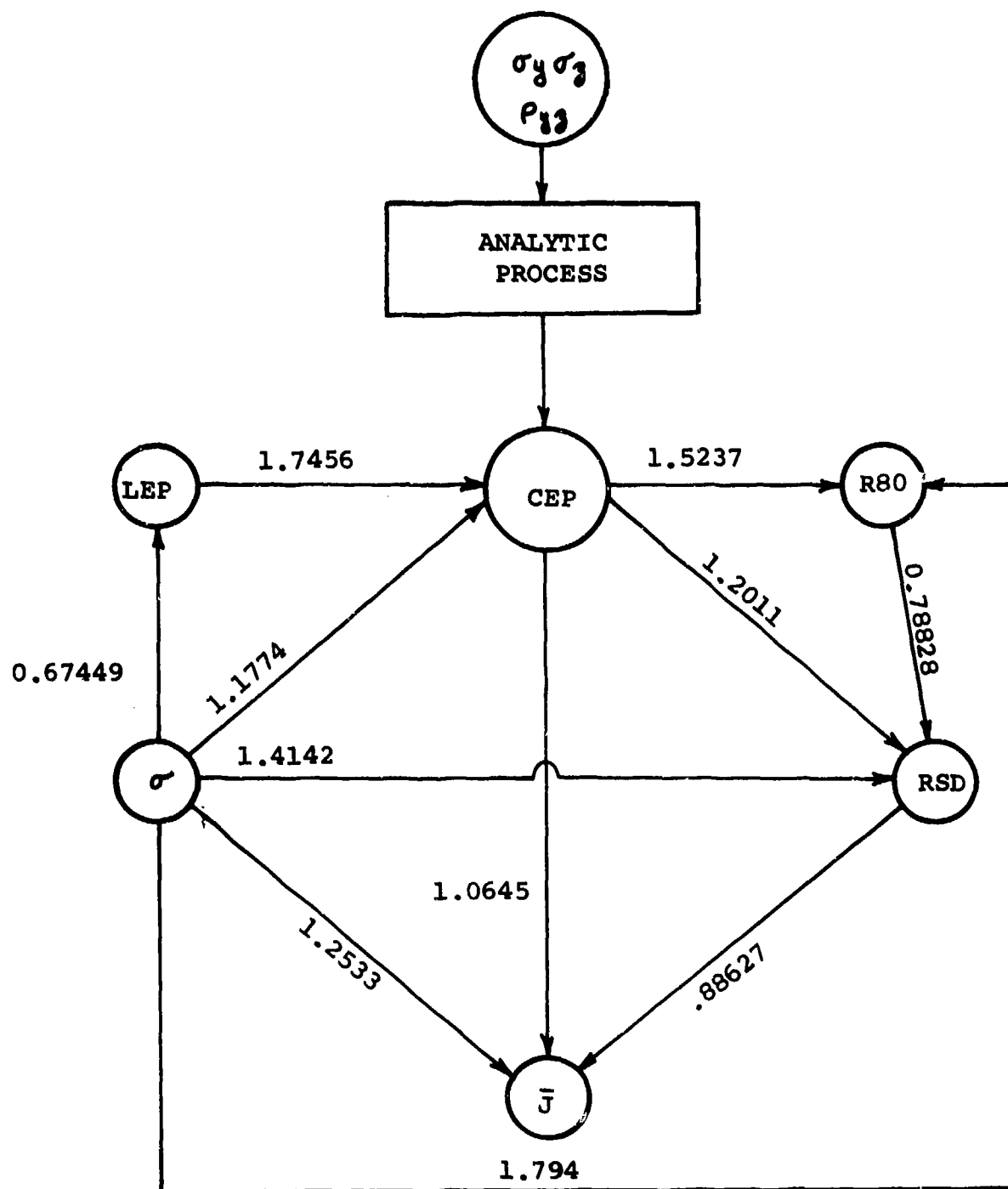


FIGURE A-6 GAUSSIAN SCALE FACTORS

Table A-1 Probabilities of Occurrence

Circle Radius	Probability of Occurrence (%)
LEP	20.3
σ	39.3
CEP	50.0
J	54.4
RSD	63.2
R80	80.0

The covariance matrix is symmetric, positive definite and completely determined by N the standard deviations σ_{x_i} and the correlation coefficients $\rho_{x_i x_j}$.

A.4.4 Rayleigh Distributed Random Variable

Let J be a Rayleigh distributed random variable. The probability density function is

$$f(J) = \frac{\pi}{2} \frac{J}{\bar{J}^2} \text{Exp} \left[-\frac{\pi}{4} \left(\frac{J}{\bar{J}} \right)^2 \right] \quad (\text{A.4-16})$$

where $\bar{J}$ is the mean value of the random variable J . The distribution is completely specified by $\bar{J}$ and

$$\sigma_J = \sqrt{\frac{4}{\pi} - 1} \bar{J} = 0.52272 \bar{J} \quad (\text{A.4-17})$$

The Rayleigh distribution is interesting because of its relationship to two jointly distributed Gaussian random variables. If x and y are identically distributed (i.e., $\sigma_x = \sigma_y = \sigma$) independent (i.e., $\rho_{xy} = 0$) Gaussian random variables, the radius

$$J = \sqrt{x^2 + y^2} \quad (\text{A.4-18})$$

is Rayleigh distributed with mean $\bar{J}$ (i.e., as determined by Figure A-5). The orientation angle

$$\theta = \tan^{-1} y/x \quad (\text{A.4-19})$$

is uniformly distributed between $\pm \pi$ radians. Thus the Rayleigh/uniform distribution describes the Gaussian distribution in polar coordinates.

A.5 Random Variable Generation

In Monte Carlo analyses it is necessary to generate random variables with known distributions. It is desired to make this procedure as simple as possible to conserve running time, simplify conversion to other machines, and not overcomplicate the programming.

The HITS code contains a random number generator which provides uniformly distributed random numbers in the interval from zero to one. This section defines transformations which convert the output of the random number generator into one of three random variable types: uniformly distributed, Gaussian, or arbitrary (tabularly defined) distributed. It is convenient to concurrently describe the method of incrementing the histogram cell counters to record the distribution of the generated random sequence. For this discussion N_c is defined to be the number of histogram cells. R is the uniformly distributed number on the interval zero to one obtained from the random number generator.

A.5.1 Uniformly Distributed Variables (TYPE = 3)

For uniformly distributed variables it is assumed $\bar{x}$ is the mean value and T is the tolerance. The upper and lower bounds defining the extent of the histogram are given by

$$\begin{aligned}x_l &= \bar{x} - T \\x_u &= \bar{x} + T\end{aligned}\tag{A.5-1}$$

The random variable

$$x = \bar{x} + (2R - 1)T\tag{A.5-2}$$

is uniformly distributed on between x_l and x_u . The histogram cell number whose counter should be increased is

$$I = \text{Int}[N_c R] + 1\tag{A.5-3}$$

where Int yields the largest integer which does not exceed the argument.

A.5.2 Gaussian Distributed Variables (TYPE = 2)

For the Gaussian distributed variables, it is assumed $\bar{x}$ is the mean value and the tolerance T is 3σ where σ is the standard deviation. The upper and lower bounds for the histogram are

$$x_l = \bar{x} - 3\sigma \quad (\text{A.5-4})$$

$$x_u = \bar{x} + 3\sigma$$

so that the total range of the histograms is 6σ .

A uniformly distributed random number R is converted into a Gaussian random number by the following procedure which uses the conversion table:

$r_1 = 0.0$	$r_6 = 0.50$	$r_{11} = 1.00$
$r_2 = 0.28$	$r_7 = 0.54$	
$r_3 = 0.36$	$r_8 = 0.58$	
$r_4 = 0.41$	$r_9 = 0.64$	
$r_5 = 0.46$	$r_{10} = 0.78$	

An index J is calculated according to

$$J = \text{Int} [10 R] + 1 \quad (\text{A.5-5})$$

where the integer function Int has previously been defined. The Gaussian distributed number is then calculated from

$$R_G = (10 R - J + 1) (r_{J+1} - r_J) + r_J \quad (\text{A.5-6})$$

R_G is Gaussian with a mean of $\frac{1}{2}$ and a standard deviation of $1/6$. The sample value, R_G , is then scaled to form the sample of the desired Gaussian random variable

$$x = \bar{x} + (2R_G - 1) T \quad (\text{A.5-7})$$

and the appropriate histogram cell number is given by

$$(\text{A.5-8})$$

$$I = \text{Int} [N_c R_G] + 1$$

A.5.3 Arbitrarily Distributed Variable (TYPE = 4)

For the arbitrarily distributed random variables, the probability density function is defined by a sequence of points x_i , f_i . It is assumed that these values are tabulated with equal divisions of the x coordinate. Furthermore, the average value of the x coordinates is assumed equal to the mean value and occurs midway in the table, the probability of exceeding the x table is zero, and the maximum value for the distribution density function is known. That is

$$\bar{x} = \frac{1}{2} (x_1 + x_k) \quad (\text{A.5-9})$$

$$T = \frac{1}{2} (x_k - x_1) \quad (\text{A.5-10})$$

$$(f_i)_{\max} \text{ is known} \quad (\text{A.5-11})$$

Two uniformly distributed random numbers are generated and are designated R_1 and R_2 . Using R_1 , a tentative sample value is generated from the expression

$$x = \bar{x} + (2R_1 - 1) T \quad (\text{A.5-12})$$

with an associated histogram interval number being given as

$$I = \text{Int}[N_c R_1] + 1 \quad (\text{A.5-13})$$

An index J is also determined from the first random number by the parallel expression

$$J = \text{Int}[K R_1] + 1 \quad (\text{A.5-14})$$

where K is the number of tabulated points. The index J is an interpolative index which determines between which two values, x_J and x_{J+1} , R_1 actually lies. Using this index the value of the probability density function is approximated by linear interpolation

$$f(x) = f_J + \frac{(x - x_J)}{x_{J+1} - x_J} (f_{J+1} - f_J) \quad (\text{A.5-15})$$

The ratio of this value to the maximal value is calculated and compared with the second random number. If the relation,

$$\frac{f}{(f_i)_{\max}} \geq R_2 \quad (\text{A.5-16})$$

is satisfied then the sample value of the variable is accepted; if it is not, the sample value is rejected, two more random numbers are generated and the process repeated. This is performed until an acceptable pair of random numbers is found or a predetermined number of trials (20) have been performed in which case a system level error is generated and execution is suspended.

This method is called Von Newmann rejection sampling. The method will generate samples which emulate the given distribution function. However, it can be inefficient. Referring to Figure A-7, the area A_1 represents the area under the distribution function in the transformed plane of the two random numbers. The method consists of accepting pairs of numbers which lie under the transformed distribution density curve.

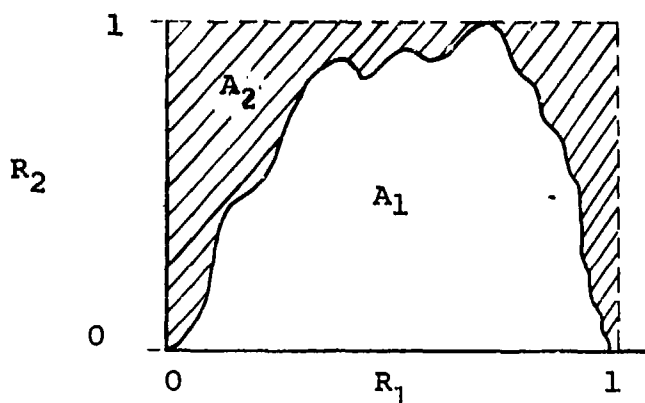


FIGURE A-7 VON NEWMANN REJECTION SAMPLING ILLUSTRATION

The area A_2 represents the region where rejection occurs. A sampling efficiency, η , can be defined as the ratio of sample points accepted to the total number of trials. This efficiency is

$$\eta = \frac{A_1}{A_1 + A_2} = A_1 = \frac{1}{(f_i)_{\max} 2T} \quad (\text{A.5-17})$$

Typically, it can be expected that on the order of 50% of the samples will be rejected, which is an acceptable rate. For very peaked distributions this method could prove to be inefficient. A large number of arbitrarily distributed variables would compound the inefficiency. If this inefficiency proves to be a hinderance, other methods would be more appropriate and should be substituted.

A.6 Monte Carlo Inaccuracies

This section discusses two sources of error in Monte Carlo techniques: the grouping error and the sampling error. Guidelines are developed for minimizing these errors. In summary, these errors are small whenever the histogram cells are small and the sample size is large.

A.6.1 Grouping Errors

In γ Monte Carlo analysis, the values resulting from the experiment are sorted into histogram cells and counted rather than recording the precise values. The purpose is to conserve computer storage. Subsequently, the histogram data is used to calculate moments. Since the exact values are lost in the process, an error, termed the grouping error, is introduced in the moments

To illustrate the problem, consider the random variable x with density function $f(x)$. During Monte Carlo experiments the observed values of x are grouped into cells which is tantamount to assuming that all observed values fell at the midpoint of the cell. This is an error. Thus, in reality the histogram does not correspond to the true distribution of x but rather to a discrete distribution where x can take only values associated with the midpoints of the cells. That is

$$x = x_{m_i} \quad i = 1, \dots, N_c \quad (A.6-1)$$

are the only possibilities where x_{m_i} are the midpoints. The probability associated with each of these values is

$$P \left[x_{b_i} \leq x \leq x_{b_{i+1}} \right] = \int_{x_{b_i}}^{x_{b_{i+1}}} f(x) dx \quad (A.6-2)$$

where x_{b_i} and $x_{b_{i+1}}$ are the lower and upper cell boundaries. The ensuing moment calculation based on the histogram is

$$\hat{x}^n = \sum_{i=1}^{N_c} x_{m_i}^n \int_{x_{b_i}}^{x_{b_{i+1}}} f(x) dx \quad (\text{A.6-3})$$

As a prelude to evaluating the grouping error, consider the true n^{th} moment of x given by the integral

$$\overline{x^n} = \int_{-\infty}^{\infty} x^n f(x) dx \quad (\text{A.6-4})$$

This integral can be subdivided into the contributions from each histogram cell to yield

$$\overline{x^n} = \sum_{i=1}^{N_c} \int_{x_{b_i}}^{x_{b_{i+1}}} f(x) x^n dx \quad (\text{A.6-5})$$

where x_{b_i} , $x_{b_{i+1}}$ are the boundary values of the cells and it is assumed

$$f(x) = 0 \quad \begin{array}{l} x < x_{b_0} \\ x > x_{b_{N_c+1}} \end{array} \quad (\text{A.6-6})$$

The grouping error is the difference between the calculated moment, Eq. (A.6-3), and the true moment as given by Eq. (A.6-5). The exact magnitude of the grouping error in the n^{th} moment depends on the specific distribution. It cannot be evaluated in general unless the density function is approximated. A good first order assumption is all that is required: it is assumed that the density function, $f(x)$, is constant over each histogram cell, i.e.,

$$f(x) = f_i \quad x_{b_i} < x \leq x_{b_{i+1}} \quad (\text{A.6-7})$$

and each of the cells are assumed to be equal in size, $\Delta x = x_{b_{i+1}} - x_{b_i}$. Thus, the midpoints of the cells are given by

$$x_{m_i} = x_{b_i} + \frac{\Delta x}{2} \quad (\text{A.6-8})$$

The grouping error, ϵ^N , is the difference between the true and calculated moments of order N

$$\epsilon^N = \overline{x^N} - \hat{x^N} = \sum_{i=1}^{N_c} f_i \left[\int_{x_{b_i}}^{x_{b_{i+1}}} x^N dx - x_{m_i}^N \Delta x \right] \quad (\text{A.6-9})$$

With the aid of the identity

$$\sum_{i=1}^n f_i \Delta x = \sum_{i=1}^n \int_{x_{b_i}}^{x_{b_{i+1}}} f(x) dx = 1 \quad (\text{A.6-10})$$

the grouping error can be evaluated for low order moments. In particular

$$\begin{aligned} \epsilon^1 &= 0 \\ \epsilon^2 &= \frac{\Delta x^2}{12} \end{aligned} \quad (\text{A.6-11})$$

Thus, to first order, there is no grouping error in the calculation of the mean value; but, there is an error in the calculated variance equal to $\Delta x^2/12$, where Δx is the cell size.

The expressions for the grouping error given by Eq. (A.6-11) is quite accurate. A more general derivation with a more accurate approximation to the distribution function yields the same result. Corrections for grouping errors are called Sheppards corrections and are treated in greater detail in appropriate texts on statistics.¹ These corrections are not applied in the present Monte Carlo code because they can be made arbitrarily small. Since cell sizes are selected in proportion to the standard deviation of the variables, the percentage error due to grouping is of the

¹Cromer, H., "Mathematical Methods of Statistics," Princeton University Press, 1946.

order $(10/N_C)^2$ where N_C is the number of histogram cells. Thus, using more than ten cells renders the grouping error on the order of one percent of the standard deviation or less, which is small in comparison to other sources of Monte Carlo uncertainties.

A.6.2 Sampling Errors

Monte Carlo methods are exact only in the limit as the number of experiments (or samples) becomes arbitrarily large. Results based on finite sample sizes contain errors called sampling errors, because finite sequences of random numbers are not representative of the entire distribution. This section treats the effects of finite sample size on estimates of the density function and the moments. In summary, sampling errors can be minimized by taking sufficiently large sample sizes. Guidelines are developed.

Estimated Density Function

A random number generator seeks to generate random numbers which are uniformly distributed over the interval zero to one. Let a sequence of N_T such numbers be sorted into N_C histogram cells of equal size covering the range zero to one. It would be expected that each cell would occur N_T/N_C times, or a relative frequency of $1/N_C$. Thus, for $N_C = 10$ and $N_T = 100$, 10 values should fall in each cell and the indicated probability of occurrence of any given cell would be $1/10$. However, when an actual sequence of random numbers is sorted it is generally found that neither of these theoretical expectations is true. This section discusses this problem. Only uniformly distributed random variables are discussed, although the results are generally true regardless of distribution type.

The above hypothetical situation can be explained with the aid of the binomial distribution. The binomial distribution states that if an event has a probability p of occurring and a probability $q = 1 - p$ of not occurring, then the probability of exactly X occurrences of the event in N trials is

$$P(X) = \frac{N!}{X! (N - X)!} p^X q^{N-X} \quad (A.6-12)$$

For any particular cell, the probability that the generated random number will fall in that cell is $p = 1/N_C$, with the nonoccurrence probability being $q = 1 - 1/N_C$. According to Eq. (A.6-12), the probability any cell will occur exactly N_T/N_C times in N_T

trials is

$$P\left(\frac{N_T}{N_c}\right) = \frac{N_T!}{\left(\frac{N_T}{N_c}\right)! \left[N_T \left(1 - \frac{N_T}{N_c}\right)\right]!} \left(\frac{1}{N_c}\right)^{\frac{N_T}{N_c}} \left(1 - \frac{1}{N_c}\right)^{N_T \left(1 - \frac{1}{N_c}\right)} \quad (\text{A.6-13})$$

where it is assumed that N_T/N_c is an integer quantity. With the assumption that the number of samples is large relative to the number of cells, $N_T \gg N_c$, Stirlings approximation

$$N! \approx \sqrt{2\pi N} N^N e^{-N} \quad (\text{A.6-14})$$

can be used to simplify Eq. (A.6-13):

$$P\left[\frac{N_T}{N_c}\right] \approx \sqrt{\frac{N_c}{2\pi N_T (1 - 1/N_c)}} \quad (\text{A.6-15})$$

Using this expression it is found that for a sequence of 100 random numbers sorted into 10 cells the probability any cell occurs exactly 10 times is about 1 chance in 8. Increasing the size of the sample to 1000 doesn't help, the probability of exactly 100 samples in a cell is about 1 chance in 24. Alternately, sorting the sequence of 100 into 5 cells doesn't help either; the probability of any cell occurring exactly 20 times is about 1 chance in 10. This illustrates that as the sample size is increased, or the number of cells is decreased, the probability of observing the theoretically expected number of occurrences per cell diminishes.

It would appear there is no means for obtaining a more exact definition of the probability density function. This is not true, since the density function is more precisely defined in an average sense by larger sample sizes, as illustrated by the following example.

A sequence of 100 random numbers sorted into five cells has an expected number of occurrences of 20 per cell. A 5% deviation

in the actual occurrences per cell would be either 19, 20 or 21 in a cell. The associated probability is

$$P(19) + P(20) + P(21) \approx 3 P(20) = 0.3 \quad (\text{A.6-16})$$

Increasing the sequence of random numbers tenfold, the expected number of occurrences is 200 per cell and a 5% deviation would span the twenty values 190, 191, ..., 210. The associated probability is

$$\sum_{n=190}^{210} P(n) \approx 20 P(200) = 0.6 \quad (\text{A.6-17})$$

Thus, the chances of being within 5% of the theoretical are about twice as good with 1000 samples as they are for 100 samples. This example illustrates that the number of occurrences tends to the theoretical as the number of trials tends to infinity in an average sort of way. Thus, larger sample sizes improve the estimate of the density function. A general rule of thumb is contained in the old saw, "a few hundred is too few and a thousand is plenty."

Estimated Moments

Monte Carlo methods estimate the moments using Eq. (A.6-3) by approximating the density function with the observed relative frequency

$$f(x) = \frac{N_i}{N_T} \frac{1}{x_{b_{i+1}} - x_{b_i}} \quad x_{b_i} \leq x \leq x_{b_{i+1}} \quad (\text{A.6-18})$$

so that

$$\int_{x_{b_i}}^{x_{b_{i+1}}} f(x) dx \approx \frac{N_i}{N_T} \quad (\text{A.6-19})$$

where N_i is the number of occurrences of the i th cell, N_T is the total number samples, and x_{b_i} and $x_{b_{i+1}}$ are the lower and upper bounds of the cell. Thus, errors in the estimated moments are the compound effect of grouping errors and density function uncertainties.

Section A.6.1 showed that grouping errors can be made arbitrarily small by using enough cells. This section treats the moment errors associated with uncertainties in the estimated density function. In order to evaluate the errors in the moments determined by Monte Carlo methods, let

$$y = y(x) \quad (\text{A.6-20})$$

where x is the independent random variable and y is dependent. Since x is a random variable and y is a function of x , y is also a random variable. Therefore, y has a density function f_y with moments $\bar{y}$, the mean value, and σ_y^2 , the variance, defined by

$$\bar{y} = \int_{-\infty}^{\infty} y f(y) dy \quad (\text{A.6-21})$$

$$\sigma_y^2 = \int_{-\infty}^{\infty} (y - \bar{y})^2 f(y) dy$$

The problem is to assess the errors in the mean value and variance computed from a finite Monte Carlo sequence.

In applying the Monte Carlo method a sequence of random values: $x_1, x_2, \dots, x_{N_T}$ is generated and the resulting sample sequence $y_1, y_2, \dots, y_{N_T}$ is computed. This latter sequence is a sequence of random numbers which in turn has a mean

$$\mu_1 = \frac{1}{N_T} \sum_{i=1}^{N_T} y_i \quad (\text{A.6-22})$$

If a second sample is generated using a different sequence of random numbers, the second sample will have a mean given as

$$\mu_2 = \frac{1}{N_T} \sum_{i=1}^{N_T} y_i \quad (\text{A.6-23})$$

which in general will be different from the first. If the process is repeated many times then a set of values, $\mu_1, \mu_2, \dots, \mu_J$

will result. This set of values has a distribution function and a mean value and variance associated with it. The question arises then as whether the important properties of the sample mean distribution can be determined without actually determining the sampling distribution itself. This would provide the necessary guidelines for minimizing the sample errors in the moments. This can be done. The sample mean

$$\mu = \frac{1}{N_T} \sum_{i=1}^{N_T} y_i \quad (\text{A.6-24})$$

is a function of N_T statistically independent variables; y_i , $i = 1, \dots, N$. Forming the expectation of Eq. (A.6-24) gives the mean value of the sample mean

$$E[\mu] = \frac{1}{N_T} \sum_{i=1}^{N_T} E[y_i] \quad (\text{A.6-25})$$

which, according to Eq. (A.6-21), is

$$E[\mu] = \bar{y} \quad (\text{A.6-26})$$

Therefore, the mean value of the sample mean is equal to the mean. The variance of the sample mean is found similarly. Thus noting

$$\mu - \bar{y} = \frac{1}{N_T} \sum_{i=1}^{N_T} (y_i - \bar{y}) \quad (\text{A.6-27})$$

squaring both sides and forming the expectation

$$\text{Var}(\mu) = E[(\mu - \bar{y})^2] = \frac{1}{N_T} E \left[\sum_{i=1}^{N_T} (y_i - \bar{y})^2 \right] \quad (\text{A.6-28})$$

$$\text{Var}(\mu) = \frac{1}{N_T^2} \sum_{i=1}^{N_T} E[(y_i - \bar{y})^2] \quad (\text{A.6-29})$$

Since $E (y_i - \mu)^2 = \sigma_y^2$ the variance of the sample mean is

$$\text{Var}(\mu) = \frac{\sigma_y^2}{N_T} \quad (\text{A.6-30})$$

Thus for a sample size of N_T of a random variable y , the standard deviation of the sample mean is $\sigma_y/\sqrt{N_T}$ where σ_y is the standard deviation of y and N_T is the sample size. On the basis of a single Monte Carlo sample of size N_T , one would report the estimate of the mean value $\bar{y}$ as

$$\bar{y} = \mu \pm \frac{\sigma_y}{\sqrt{N_T}} \quad (\text{A.6-31})$$

The $\sigma_y/\sqrt{N_T}$ is termed a sampling error for the mean. To reduce the sampling error by half requires a quadruple increase in the sample size. By the central limit theorem, for a large number of trials the distribution of the sample mean will tend to be Gaussian. It is therefore possible to associate confidence limits with it. For example, the true value of $\bar{y}$ should lie within $\mu \pm 3 \sigma_y/\sqrt{N_T}$ with 95% confidence.

Just as there is a distribution of the sample mean, there is also a distribution associated with the sample variance

$$S^2 = \frac{1}{N_T - 1} \sum_{i=1}^{N_T} (y_i - \mu)^2 \quad (\text{A.6-32})$$

The expected value of the sample variance is

$$E[S^2] = \sigma_y^2 \quad (\text{A.6-33})$$

The $(N_T - 1)$ factor in Eq. (A.6-32) is included so the sample variance is unbiased; that is, without an expected error. The standard deviation of the sample variance is

$$\sigma_{S^2} = \sigma_y^2 \sqrt{\frac{2}{N_T - 1}} \quad (\text{A.6-34})$$

To summarize, the moments computed from a Monte Carlo sample of size N_T have the following characteristics

$$\bar{y} = \mu \pm \sigma_y / \sqrt{N_T} \quad (\text{A.6-35})$$

$$\sigma^2 = S^2 \pm \sigma_y^2 \sqrt{\frac{2}{N_T - 1}} \quad (\text{A.6-36})$$

where μ and S^2 are the sample mean and sample variance, respectively. The HITS code calculates μ and S^2 for each of the variables of interest. The program does not determine the sampling errors involved. The sampling errors are plotted in Figures A-8 and A-9 for various confidence levels. Clearly, samples less than several hundred will produce moments with substantial errors. Samples exceeding a thousand provide adequate accuracy.

While the discussion of the sampling errors in the moments has been based on a single variable, the basic conclusions are applicable to the multiple variable case as long as the variables are statistically independent.

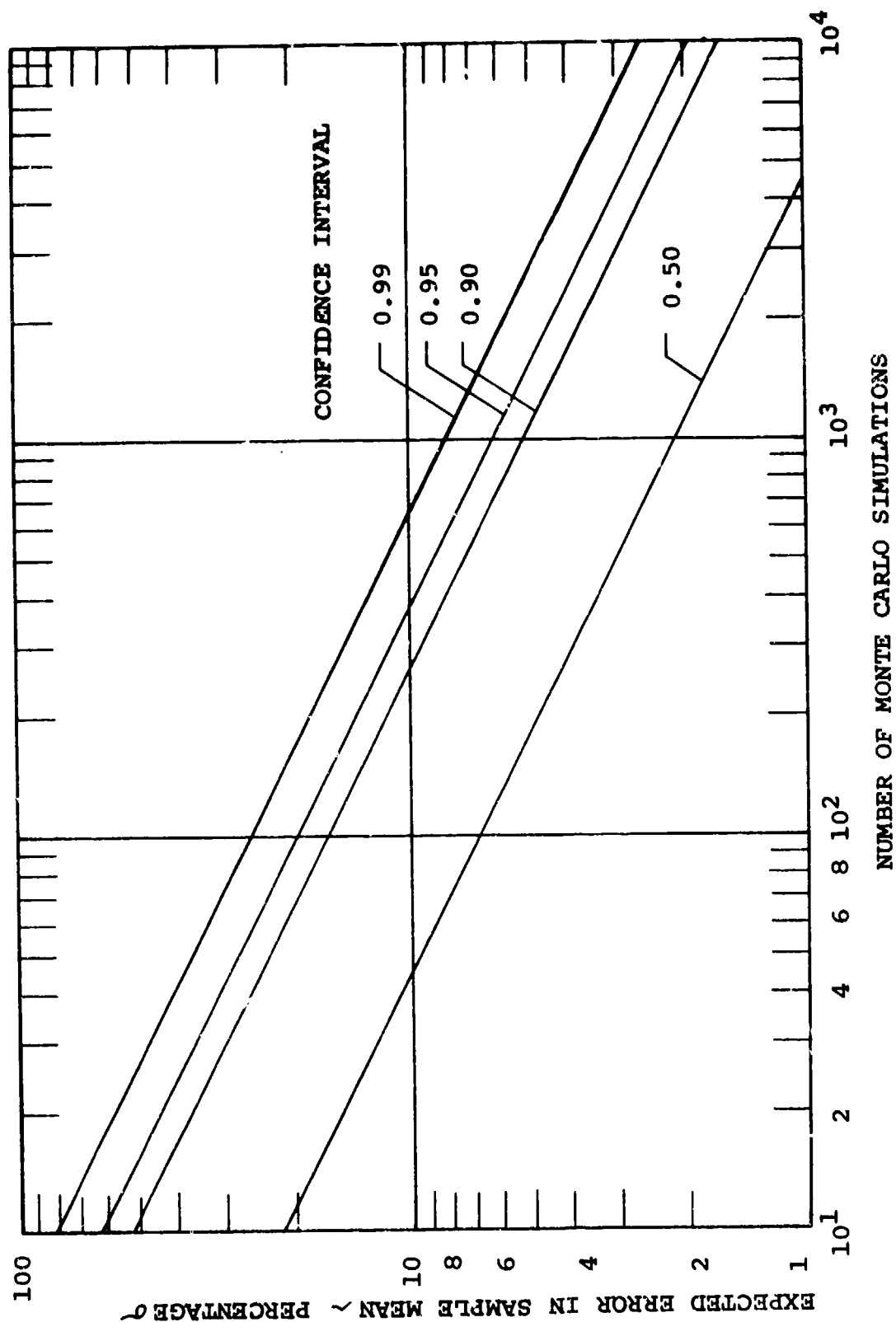


FIGURE A-8 EXPECTED ERROR IN SAMPLE MEAN IN PERCENTAGE OF STANDARD DEVIATION

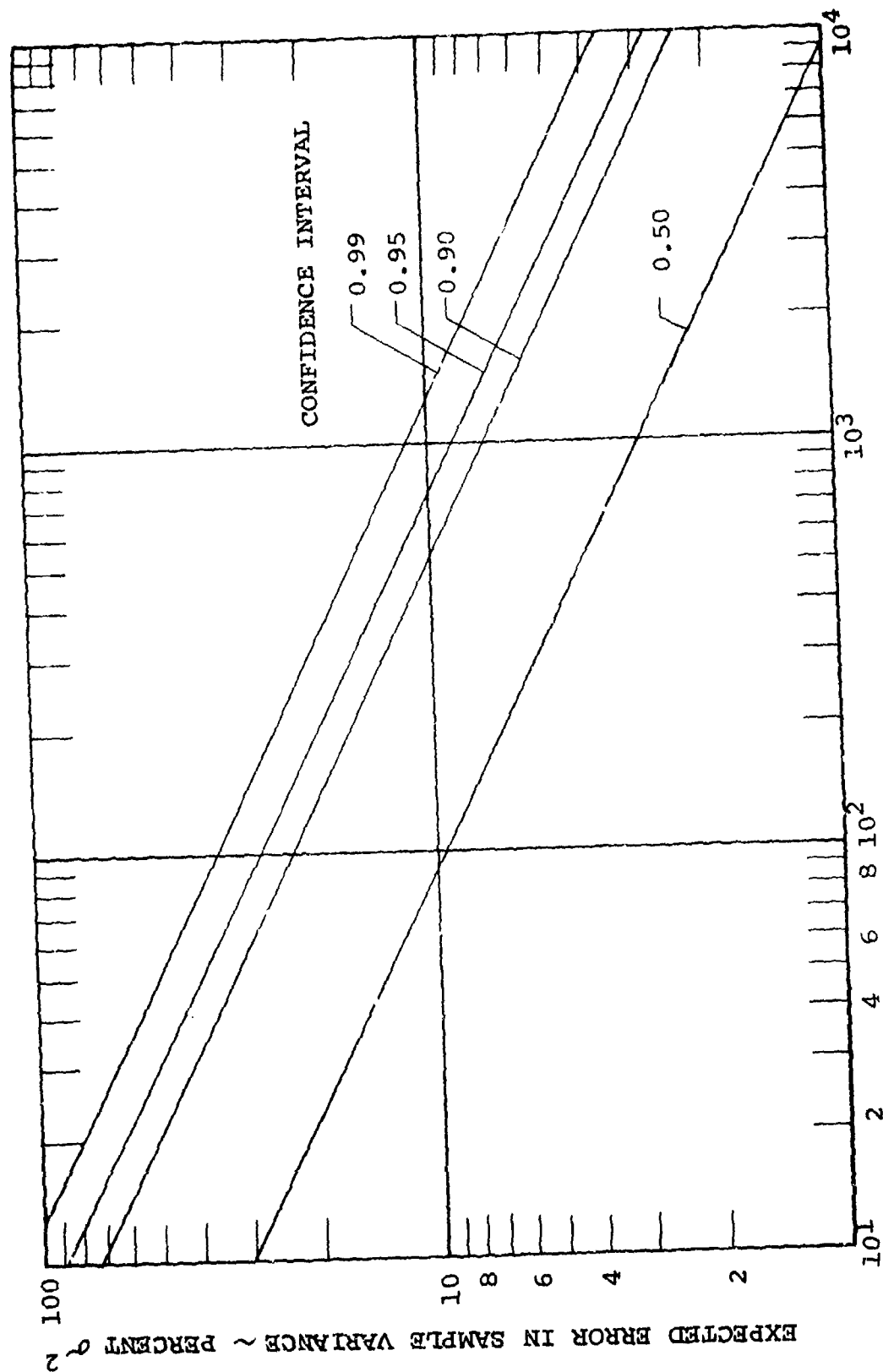


FIGURE A-9 EXPECTED ERROR IN SAMPLE VARIANCE VALUE

to determine
Processor
Section B.1.1, and
Section B.1.2.

B.1.1 Analytical Statistical Calculations

The theoretical dispersion assessment problem is to statistically determine the effect of independent variables on a dependent variable. For instance, the effect of the projectile error source model (independent variables) on the down-range dispersion (dependent variable). In general, an exact theoretical solution is not possible. However, the equations can be accurately approximated by a low order Taylor series for reasonable variations in the independent variables, and the mean value and variance of the dependent variable can be estimated. The Analytical Statistical mode mechanizes this concept.

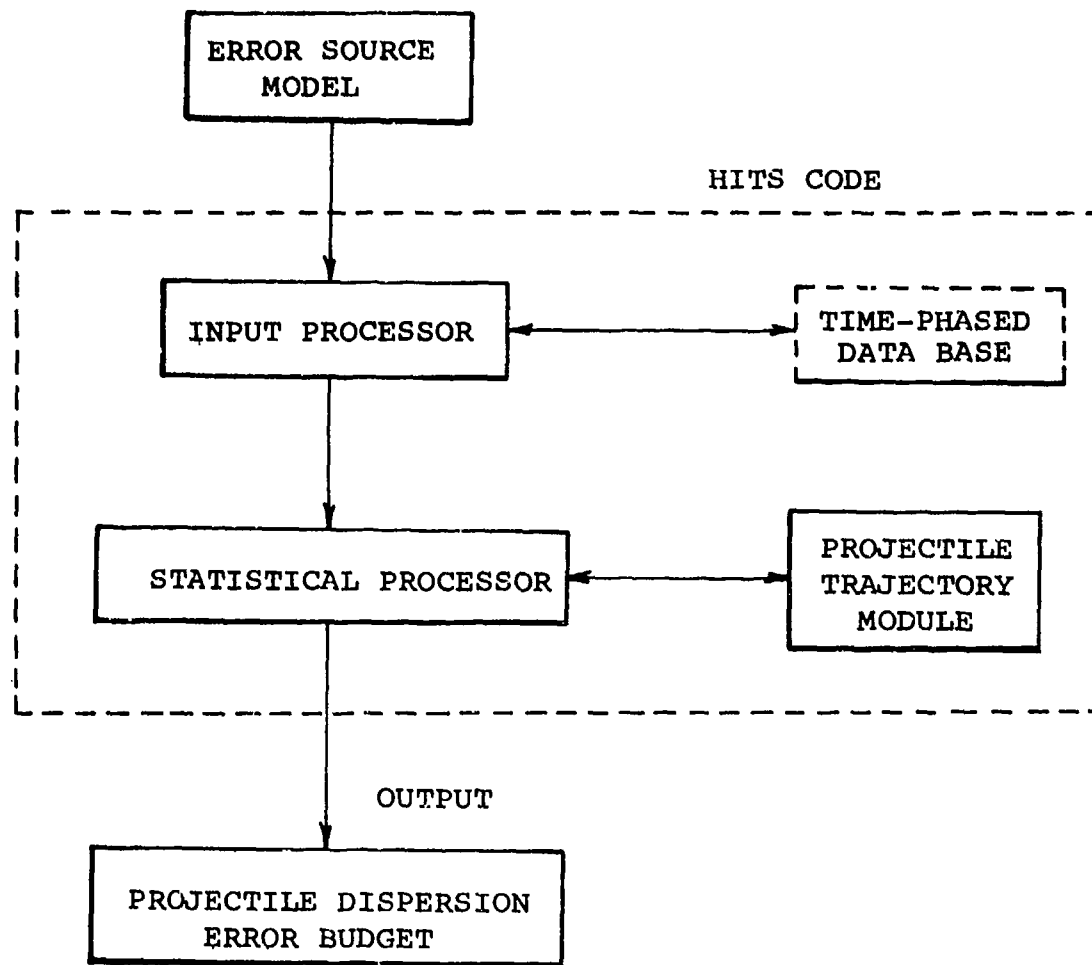


Figure B-1 HITS Flow Diagram

Taylor Series Approximation

The relationship between the dependent variable, y , and the independent variables, $x_1, x_2, \dots, x_n$, is representable as an algebraic equation

$$y(\underline{x}) = y(x_1, x_2, \dots, x_n) \quad (\text{B.1-1})$$

where $x_1, \dots, x_n$ are treated as the elements of a column vector $\underline{x}$. In general, the equation is non-linear. With the assumption that the function y is differentiable and reasonably well behaved, it can be accurately approximated by a second order Taylor series¹ about the mean value of the independent variables, $\bar{\underline{x}} = (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n)^T$.

$$\begin{aligned} y(\underline{x}) &= y(\bar{\underline{x}}) + \frac{\partial y}{\partial \underline{x}}(\bar{\underline{x}})(\underline{x} - \bar{\underline{x}}) + \\ &+ \frac{1}{2}(\underline{x} - \bar{\underline{x}})^T \frac{\partial}{\partial \underline{x}} \left(\frac{\partial y}{\partial \underline{x}} \right)^T(\bar{\underline{x}})(\underline{x} - \bar{\underline{x}}) \end{aligned} \quad (\text{B.1-2})$$

where T denotes the matrix transpose and

$$\frac{\partial y}{\partial \underline{x}}(\bar{\underline{x}}) = \left[\frac{\partial y}{\partial x_1}(\bar{\underline{x}}), \frac{\partial y}{\partial x_2}(\bar{\underline{x}}), \dots, \frac{\partial y}{\partial x_n}(\bar{\underline{x}}) \right] \quad (\text{B.1-3})$$

is the gradient row vector of first partial derivatives

¹Sokolnikoff, I. S., and Sokolnikoff, E. S., Higher Mathematics for Engineers and Physicists, McGraw-Hill, New York, 1941.

and

$$\frac{\partial}{\partial \underline{x}} \left(\frac{\partial y}{\partial \underline{x}} \right)^T (\bar{\underline{x}}) = \begin{bmatrix} \frac{\partial^2 y}{\partial x_1^2} (\bar{\underline{x}}) & \frac{\partial^2 y}{\partial x_2 \partial x_1} (\bar{\underline{x}}) & \dots & \frac{\partial^2 y}{\partial x_n \partial x_1} (\bar{\underline{x}}) \\ \frac{\partial^2 y}{\partial x_1 \partial x_2} (\bar{\underline{x}}) & \frac{\partial^2 y}{\partial x_2^2} (\bar{\underline{x}}) & \dots & \frac{\partial^2 y}{\partial x_n \partial x_2} (\bar{\underline{x}}) \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 y}{\partial x_1 \partial x_n} (\bar{\underline{x}}) & \frac{\partial^2 y}{\partial x_2 \partial x_n} (\bar{\underline{x}}) & \dots & \frac{\partial^2 y}{\partial x_n^2} (\bar{\underline{x}}) \end{bmatrix} \quad (\text{B.1-4})$$

is the Jacobian Matrix of second partial derivatives, both evaluated at the mean value of the independent variables, $\bar{\underline{x}}$.

Dependent Variable Mean Value

The mean value is found by taking the expectation of Eq. (B.1-2). After some manipulation, the mean value of the dependent variable can be shown to be

$$\begin{aligned} \bar{y} = y(\bar{\underline{x}}) + \frac{1}{2} \sum_{i=1}^n \frac{\partial^2 y}{\partial x_i^2} (\bar{\underline{x}}) \sigma_{x_i}^2 + \\ + \sum_{i=1}^{n-1} \sum_{j=i+1}^n \frac{\partial^2 y}{\partial x_i \partial x_j} (\bar{\underline{x}}) \text{Cov}(x_i, x_j) \end{aligned} \quad (\text{B.1-5})$$

This equation states that $\bar{y}$ is composed of the value of y at the mean of $\underline{x}$ plus second order corrections based on the variances and covariances of the independent variables. (Note: HITS includes the second term whenever the control variable IOPRNT ≥ 3 and the third whenever IOPRNT ≥ 5 .)

Dependent Variable Variance

The variance of the dependent variable is computed according to

$$\sigma_y^2 = E[y^2] - \bar{y}^2 \quad (\text{B.1-6})$$

Equations (B.1-2) and (B.1-5) are substituted and manipulated to obtain

$$\begin{aligned} \sigma_y^2 = & \sum_{i=1}^n \left[\frac{\partial y}{\partial x_i} (\bar{x}) \right]^2 \sigma_{x_i}^2 + \\ & + 2 \sum_{i=1}^{n-1} \sum_{j=i+1}^n \frac{\partial y}{\partial x_i} (\bar{x}) \frac{\partial y}{\partial x_j} (\bar{x}) \text{Cov}(x_i, x_j) + \\ & + \frac{1}{2} \sum_{i=1}^n \left[\frac{\partial^2 y}{\partial x_i^2} (\bar{x}) \right]^2 \sigma_{x_i}^4 + \sum_{i=1}^{n-1} \sum_{j=i+1}^n \left[\frac{\partial^2 y}{\partial x_i \partial x_j} (\bar{x}) \right]^2 \sigma_{x_i}^2 \sigma_{x_j}^2 + \\ & + 2 \sum_{j=1}^n \sum_{i=1}^{n-1} \sum_{k=i+1}^n \left[\frac{\partial^2 y}{\partial x_i \partial x_j} (\bar{x}) \frac{\partial^2 y}{\partial x_k \partial x_j} (\bar{x}) \right] \sigma_{x_j}^2 \text{Cov}(x_i, x_k) + \\ & + 2 \sum_{j=1}^{n-1} \sum_{m=j+1}^n \sum_{i=1}^{n-1} \sum_{k=i+1}^n \left[\frac{\partial^2 y}{\partial x_i \partial x_j} (\bar{x}) \frac{\partial^2 y}{\partial x_k \partial x_m} (\bar{x}) \right] \text{Cov}(x_i, x_k) \text{Cov}(x_j, x_m) \end{aligned} \quad (\text{B.1-7})$$

Equation (B.1-7) states that σ_y^2 is composed of linear terms based on the variance and covariances of the independent variables and a profuse number of second order corrections. (Note: HITS includes the first term whenever IOPRNT ≥ 3 and the second whenever IOPRNT ≥ 5 .) The second order corrections are exactly correct only when the independent variables are Gaussian, since the development of Eq. (B.1-7) assumed

$$E[(x_i - \bar{x}_i)(x_j - \bar{x}_j)(x_k - \bar{x}_k)(x_m - \bar{x}_m)] =$$

$$\text{Cov}(x_i, x_j) \text{Cov}(x_k, x_m) +$$

$$+ \text{Cov}(x_i, x_k) \text{Cov}(x_j, x_m) + \text{Cov}(x_i, x_m) \text{Cov}(x_j, x_k) \quad (\text{B.1-8})$$

(Note: HITS includes the third and fourth terms in Eq. (B.1-7) whenever IOPRNT ≥ 6 and the fifth and sixth whenever IOPRNT ≥ 7 .)

Calculation of Derivatives

In order to estimate the mean and variance via Eqs. (B.1-5) and (B.1-7), it is necessary to evaluate the dependent variable and its derivatives at the mean values of the independent variables. The evaluation of $y(\bar{x})$ requires one reference to the projectile trajectory module. HITS determines the derivatives by manipulating the inputs to the projectile trajectory module in a systematic fashion. A central difference scheme is used to numerically approximate the derivatives. The independent variables are incremented one at a time both positively and negatively about their mean values. Letting $\bar{x}_i$ represent the independent variable mean values and Δx_i a positive increment, two function evaluations, (i.e., calls to the projectile trajectory module), are performed to give

$$y^+ = y(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_i + \Delta x_i, \dots, x_n) \quad (\text{B.1-9})$$

$$y^- = y(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_i - \Delta x_i, \dots, x_n)$$

(Note: HITS sets Δx_i equal to the input tolerance value TOL for each independent variable.) The derivatives are then calculated according to

$$\frac{\partial y}{\partial x_i} = \frac{y^+ - y^-}{2 \Delta x_i} \quad (\text{B.1-10})$$

$$\frac{\partial^2 y}{\partial x_i^2}(\bar{x}) = \frac{y'' - 2y(\bar{x}) + y^-}{\Delta x_i^2} \quad (\text{B.1-11})$$

For the second order mixed derivatives, it is necessary to evaluate the function y four times for each pair of independent variables. The four function evaluations for the pair x_i and x_j are

$$\begin{aligned} y^{++} &= y\left(\bar{x}_1, \dots, \bar{x}_i + \frac{\Delta x_i}{2}, \dots, \bar{x}_j + \frac{\Delta x_j}{2}, \dots, \bar{x}_n\right) \\ y^{+-} &= y\left(\bar{x}_1, \dots, \bar{x}_i + \frac{\Delta x_i}{2}, \dots, \bar{x}_j - \frac{\Delta x_j}{2}, \dots, \bar{x}_n\right) \\ y^{-+} &= y\left(\bar{x}_1, \dots, \bar{x}_i - \frac{\Delta x_i}{2}, \dots, \bar{x}_j + \frac{\Delta x_j}{2}, \dots, \bar{x}_n\right) \\ y^{--} &= y\left(\bar{x}_1, \dots, \bar{x}_i - \frac{\Delta x_i}{2}, \dots, \bar{x}_j - \frac{\Delta x_j}{2}, \dots, \bar{x}_n\right) \end{aligned} \quad (\text{B.1-12})$$

The second mixed partial derivative is

$$\frac{\partial^2 y}{\partial x_i \partial x_j}(\bar{x}) = \frac{y^{++} + y^{--} - (y^{+-} + y^{-+})}{\Delta x_i \Delta x_j} \quad (\text{B.1-13})$$

Since the second order mixed partial derivatives are symmetric with respect to the independent variables, there are $n(n-1)$ distinct derivatives for any dependent variable, where n is the number of independent variables.

The increments used in the second order mixed partial derivative calculations are half the values used for the first order derivatives. The reason is illustrated in Figure B-2 which diagrams points in the $x_i - x_j$ plane. An ellipse can be drawn through the four points used for the first derivative calculation. With regards to the assumed Taylor series approximation, the ellipse represents the boundary of the region over which the linearization is valid, by virtue of the selection of the Δx 's. For consistency, the off-axis coordinates are halved to insure they will always be interior to the ellipse.

Summary

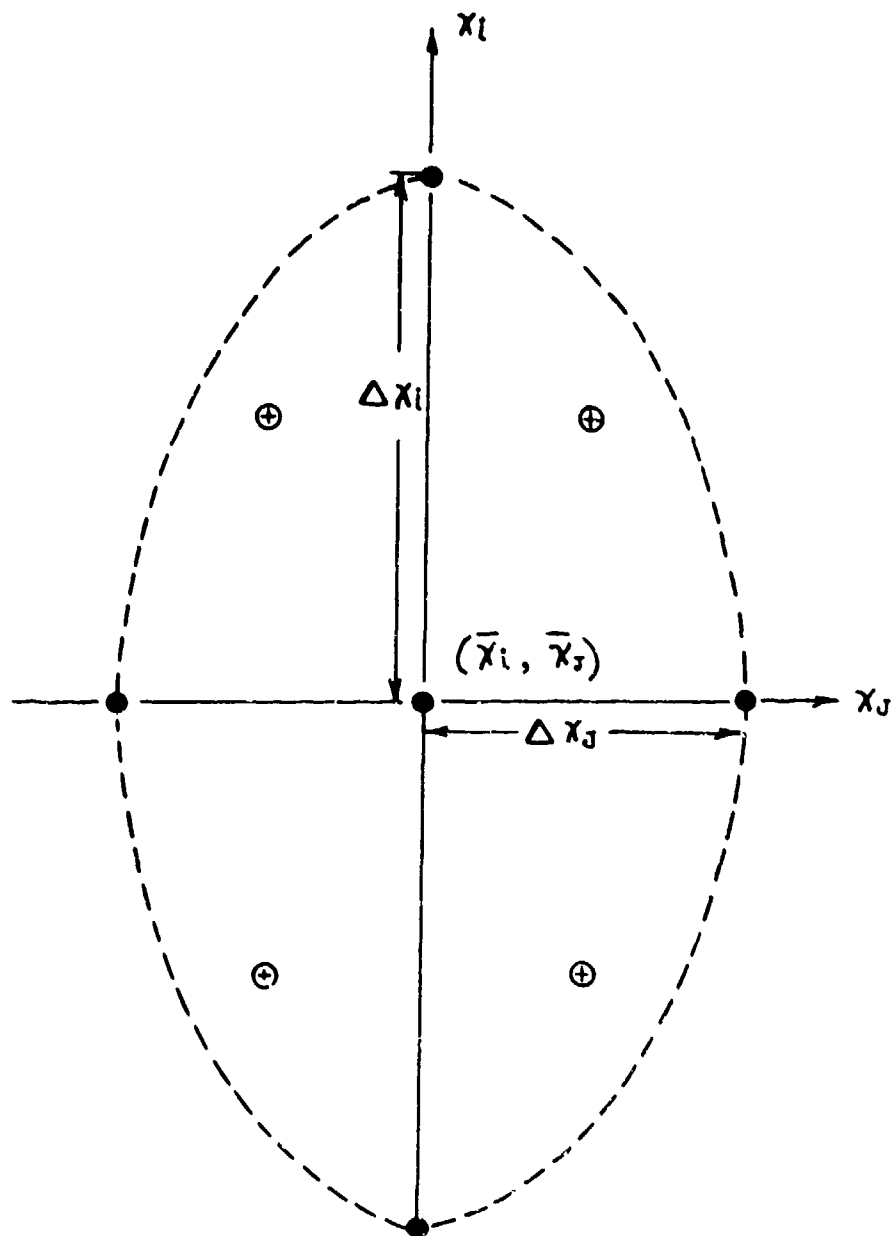
Equations (B.1-5) and (B.1-7) are the basic equations for the Analytical Statistical mode of the Statistical Processor. If there is more than one dependent variable the equations are applied to each in turn. The objective is to determine the mean and variance of all dependent variables from the input statistics of the independent variables. This forces the calculation of first and second order partial derivatives, which is a major undertaking. Since the Analytical Statistical mode is exercised prior to the Monte Carlo mode (to set up the histograms), these calculations are pertinent to the Monte Carlo simulations.

B.1.2 Monte Carlo Calculations

The Monte Carlo mode consists of three parts: (1) generation of representative independent variable values, (2) processing of these values to determine the corresponding dependent variable values, and (3) condensing the ensemble of solutions to a usable form. Representative sequences of the independent variables are obtained from random number generators, as discussed in Appendix A. The trajectory equations of Appendix C relate the dependent variables to the independent. This section discusses the third point: data condensation using histograms.

Cell Definition

In order to achieve substantial condensation, the histograms must be constructed while performing the Monte Carlo experiments. In order to do this, the boundaries of the cells must be determined prior to the experiments. They are determined by an equal division of the range of the variable as defined by the upper bound, x_u , and the lower bound, x_l . The i th cell is (x_{bi}, x_{bi+1}) , where the boundaries are given by



- POINTS FOR EVALUATION OF $\frac{\partial}{\partial x_i}, \frac{\partial}{\partial x_j}, \frac{\partial^2}{\partial x_i^2}, \frac{\partial^2}{\partial x_j^2}$
- ⊕ POINTS FOR EVALUATION OF $\frac{\partial^2}{\partial x_i \partial x_j}$

Figure B-2 Coordinates for Derivative Evaluations

$$x_{b_i} = x_\ell + (i-1) \frac{x_u - x_\ell}{N_c} \quad (\text{B.1-14})$$

where N_c is the number of cells. For independent variables, x_l and x_u are defined in Appendix A.5 for the various different types. For dependent variables, (TYPE = 7), x_l and x_u are determined by the preliminary Analytical Statistical mode calculations. They are

$$\begin{aligned} x_\ell &= \bar{x} - 3\sigma \\ x_u &= \bar{x} + 3\sigma \end{aligned} \quad (\text{B.1-15})$$

where $\bar{x}$ and σ are the calculated values of the mean value and standard deviation. An alternate set of values may be input (TYPE = 8). The cell number for a given value, x , is determined by evaluating

$$i = \text{Int} \left[\frac{x - x_\ell}{x_u - x_\ell} N_c \right] + 1 \quad (\text{B.1-16})$$

where Int is the largest integer less than the argument. The midpoint in each cell is termed the "class mark" and is denoted x_{m_i} :

$$x_{m_i} = \frac{1}{2} (x_{b_i} + x_{b_{i+1}}) = x_\ell + \left(i - \frac{1}{2}\right) \frac{x_u - x_\ell}{N_c} \quad (\text{B.1-17})$$

Density Function Estimate

The Monte Carlo mode counts the number of times each cell occurs during the experimental sequence. The probability density function is estimated according to

$$f(x) = \frac{N_i}{N_T} \frac{N_c}{x_u - x_\ell} \quad x_{b_i} \leq x \leq x_{b_{i+1}} \quad (\text{B.1-18})$$

where N_i is the number of times cell "i" occurred and

$$N_T^* = \sum_{i=1}^{N_c} N_i \quad (\text{B.1-19})$$

is the total of all cell counts for this variable. Since it is possible for dependent variables to fall outside the range covered by the histogram cells, (x_l, x_u) ; N_T^* is potentially less than the total number of experiments, N_T .

Moment Estimates

The sample mean and standard deviation are estimated from the histograms. The mean value is

$$\bar{x} = \int_{-\infty}^{\infty} x f(x) dx = \sum_{i=1}^{N_c} \int_{x_{b_i}}^{x_{t_{i+1}}} x f(x) dx \quad (\text{B.1-20})$$

which becomes the sample mean upon substitution of Eq. (B.1-18),

$$\mu = \sum_{i=1}^{N_c} \frac{N_i}{N_T^*} x_{m_i} \quad (\text{B.1-21})$$

Using Eq. (B.1-17) the sample mean is

$$\mu = x_\ell + \frac{x_u - x_\ell}{N_T^* N_c} \sum_{i=1}^{N_c} \left(i - \frac{1}{2} \right) N_i \quad (\text{B.1-22})$$

The variance is defined by

$$\sigma^2 = \int_{-\infty}^{\infty} (x - \bar{x})^2 f(x) dx \quad (\text{B.1-23})$$

Upon substitution of Eq. (B.1-18), and using the sample mean instead of the mean value, Eq. (B.1-23) becomes an expression for the sample variance:

$$s^2 = \left(\frac{x_u - x_\ell}{N_c} \right)^2 \left[\sum_{i=1}^{N_c} \left(i - \frac{1}{2} \right)^2 \frac{N_i}{N_T^*} - \left(\sum_{i=1}^{N_c} \left(i - \frac{1}{2} \right) \frac{N_i}{N_T^*} \right)^2 \right] \quad (\text{B.1-24})$$

B.2 Code Architecture

This section discusses the architecture of the HITS code. The architecture was designed to meet the requirements of the Statistical Processor computational procedures just described. The four storage arrays discussed in Section B.2.1 form the basic structure. The input processor of Figure B-1 plays the intimate role described in B.2.2. Statistical Processor functions are presented in B.2.3. Subroutine definitions and linking are detailed in B.2.4. The facility for computer generated histogram plots is discussed in B.2.5.

B.2.1 Data Storage

The entire code is designed around four arrays in common storage. These arrays are

- 1) "OE" array - This one-dimensional array is a common input-output storage area to be shared by the projectile trajectory module and the Statistical Processor. That is, all subroutines are written with all their variables equivalenced to the OE array. The variable "code numbers" are addresses in the OE array.
- 2) "IA" array - This bicolonn array is used to store address links between the "OE" array and the additional storage areas of the B and C arrays described next. The first 10 rows of the IA array are reserved for dependent variables.
- 3) "B" array - This one-dimensional array is used to store most input information and some additional values.
- 4) "C" array - This one-dimensional array is used to store some input information and intermediate output information. It is a general scratch pad storage for the Statistical Processor.

The data transfers between the arrays and the computational modules that effect these transfers are indicated in Figure B-3.

B.2.2 Input Processor Functions

The Input Processor is a collection of subroutines. It determines the kind of calculations to be performed, the subroutines to be used, and the data required. The Input Processor reads the data, checks it for completeness and takes appropriate action if errors are detected or data is found to be missing. If a complete set of data has been input, control is relinquished to the Statistical Processor. This section discusses the functions of the Input Processor in detail.

The Input Processor determines which trajectory module is going to be exercised (at present, there is only the one described in Appendix C, which is specified by $IY = 1$), clears the IA array, and stores a list of code numbers into the first column of the IA array starting at row 11. This is a complete list of all the addresses in the OE array for which data is required. The first 10 rows of IA are reserved for variables in the OE array which are defined by the input to be dependent variables. The OE array is initialized at preset (i.e., default) values.

The input cards are read and processed one at a time. Generally, the card contains a code number and five values as described in the text. The five values are placed in the B array with the aid of a counter. The first column of the IA array is searched to find the row containing the code number. When it has been located, the current value of the counter is stored in the second column; and the five values are placed in the corresponding location in the B array. Subsequently, the counter is advanced by six and the next data card is read and processed in a like fashion. The input proceeds until a card having a negative code number is encountered which signifies that all data has been read in. After all the data cards have been read, each row of the IA array has two addresses, the first is a location in the OE array, (i.e., the code number), and the second is a location in the B array where data is stored. Input data is extracted by searching the first column of the IA array, and then using the second column to vector into the B array. Thus, the order of the data cards is immaterial. Note: if any rows of the IA array are interchanged the information is not disturbed.

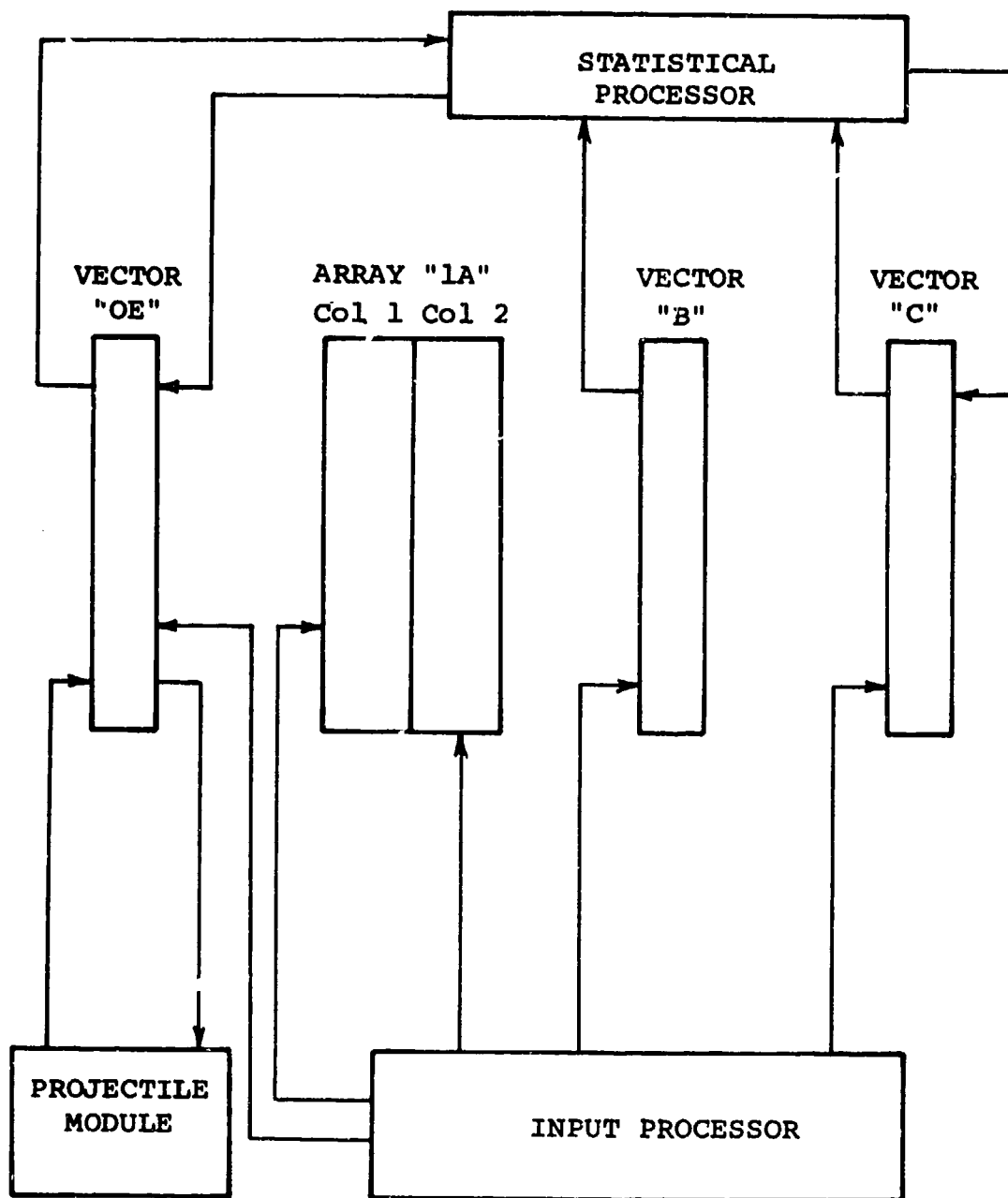


Figure B-3 Internal Data Transfers

If necessary data is missing, zeros will appear in the second column of the IA array. The second column of IA is checked for zeros with appropriate action taken whenever a zero is encountered. At present, a preset value is substituted. However, HITS has the facilities to access a "Time Phased Data Base." This option could be activated if real-time interactive terminals are available.

This description of the Input Processor is valid though over simplified. Several additions and/or modifications are necessary:

- 1) At the time each data card is read and processed, the variable value on the card is stored in both the OE array and B array. This saves a separate pass through the data to initialize the OE array.
- 2) For missing data, a "Time Phased Data Base" would be treated like a second source of data. That is, when the card reader is exhausted and data is found missing, the time phase data base is assumed to be implemented as a subroutine which can provide the missing data. This logic is present in the program, but for the present all missing data is filled in with preset values.
- 3) Dependent variable (TYPE = 7 or 8) code numbers are stored sequentially in the first 10 rows of the IA array. For a TYPE = 7, nothing is stored in the B array, and the second column of the IA array is not disturbed. For a TYPE = 8, the data is stored as previously described.
- 4) For a constant input value (TYPE = 5), a "-1" is inserted in the second column of the proper row of the IA array to signify that a value was read in. The value is then placed in the OE array.
- 5) A TYPE = 1 variable has a list of values associated with it. The list is stored in the C array. The TYPE, number of values, and the starting address for the list in the C array are stored sequentially in the B array. The appropriate index of the B array is stored in the IA array, in the appropriate row of the second column.

Since no nominal value is defined for this kind of variable, the first comment does not apply.

- 6) TYPE = 4 variables have a two-dimensional list associated with them. They are handled like a TYPE = 1 variable (see comment five). However, after read in, a nominal value, tolerance, variance, etc. are calculated and these values are stored in the B array as though they had been read in.
- 7) Inputs to the projectile trajectory module can be either floating point or fixed point. All input values are read in floating point format. To signify a value has to be stored in the OE array in an integer format, the negative of its code number is stored in the first column of the IA array. The code number is supplied by the Input Processor at initialization and therefore, there is no apparent difference to the user. Also note that integers and floating point numbers are stored in the B and OE arrays with mixed formats. The method is important to understanding the details of the code. Two equivalenced names are used, for instance, say U(5000) and IU(5000,2). The first is double precision floating and the second is single precision integer. When storing an integer value IU (*, 1) is referenced, and when storing a floating number U (*) is referenced.

B.2.3 Statistical Processor Functions

This section describes the control functions performed by the Statistical Processor. There are four modes of operation: Single Trajectory, Range Check, Analytical Statistical, and Monte Carlo.

Single Trajectory

After all data has been read by the Input Processor, the nominal case is executed and the results are stored in the C array. If the input stream contained only TYPE = 5 (constant) variables, the nominal case values are printed and a normal exit results.

Range Check

After all data has been read and all missing values filled in, the rows in the IA array are sorted so that TYPE = 1 (Range Check) variables occupy rows 11, 12 and 13. If there are TYPE = 1 variables, they are processed in a nested DO loop and the values

of the dependent variables for each combination are stored in the C array. After the DO loops have been satisfied, the C array information is printed and a normal exit results.

Analytical Statistical

If no TYPE = 1 variables are input, the IA array rows are sorted such that all TYPE = 2, 3 and 4 variables are situated at the top. After this sort, the nominal case is executed. The nominal case results are stored in the C array.

Following the nominal case, limit cases are run with the independent variables sequentially permuted. The resulting dependent variable values are stored in the C array behind the nominal case results. The derivatives are calculated sequentially. The resulting derivative values are stored in the C array by overwriting the previous function evaluations. Using the derivatives, the function mean values and variances are calculated and stored in the C array immediately after the derivatives. The derivatives, means and variances are then printed.

These processes can be time consuming and require large amounts of computer storage. For purposes of estimating the storage and time requirements, let N_D be the number of dependent variables (TYPE = 7 or 8), and N_I be the number of independent variables (TYPE = 2, 3 or 4). As a general rule, storage requirements vary with the product $N_D N_I$, whereas the time consumption varies with N_I . Table B-1 states time (passes through the projectile trajectory module) and storage requirements. The C array allocates a total of 5,000 locations to data storage of the outputs as well as tabular storage for TYPE = 4 variables. This should be more than adequate for most purposes. Should more storage be required, it is a straightforward process to increase the size of the C array in each subroutine in which it appears.

Table B-1 Time and Storage Requirements

Control Variable		Function Evaluations	Required Storage Locations (C Array)
$I\emptyset PRNT \geq 1$ $I\emptyset PRNT \geq 2$ $I\emptyset PRNT \geq 4$ $I\emptyset PRNT \geq 5$	Nominal Case	1	N_D
	Limit Cases	$2N_I$	$2N_D N_I$
	Derivatives	0	$2N_D$
	Mixed Derivatives	$4N_I(N_I + 1)$	$N_D N_I(N_I - 1)/2$
	Covariances	0	$N_I(N_I + 1)/2$

Monte Carlo

The Monte Carlo mode performs four functions: It reads additional input, verifies the input, conducts Monte Carlo experiments and constructs histograms, and outputs the results. The input step of the Monte Carlo mode is preceded by execution of the Analytical Statistical mode. The Monte Carlo mode acquires additional control parameters from the input which define the sample size, number of cells, calculational option, and allowable number of rejected trial solutions.

The second function, input verification, consists of re-arranging some of the values generated by the Analytical Statistical mode to be compatible with the Monte Carlo calculations. To efficiently describe the verification process, the range of all variables is assumed to be known in terms of

$$\bar{x} - T \leq x \leq \bar{x} + T \quad (\text{B.2-1})$$

where $\bar{x}$ is the mean value and T is the tolerance. The relationship of these variables to inputted and computed quantities has been stated elsewhere. These are summarized here along with some additions and/or modifications. In general for TYPE = 2, 3 or 8 variables the user specifies $\bar{x}$, T , and the variance σ^2 , by input. The Analytical Statistical mode calculates these same quantities for TYPE = 4 and 7 variables. To insure proper sampling and to detect possible user errors the input is verified according to the following rules.

- 1) For TYPE = 2 or 7 variables the tolerance T is taken to be the larger of the input specified value or 3σ and the input value of $\bar{x}$ is accepted.
- 2) For TYPE = 3 or 8 variables the given tolerance value and input value of $\bar{x}$ are accepted and the variance value is ignored.
- 3) For TYPE = 4 variables the midrange value and tolerance are calculated from the extremal values of the input list with the input values ignored. For example, if ten values are entered

$$\bar{x} = \frac{1}{2} (x_{10} + x_1) \quad (\text{B.2-2})$$

$$T = \frac{1}{2} (x_{10} - x_1) \quad (\text{B.2-3})$$

For TYPE = 4 variables the maximum value of the distribution function is also found and stored in the location reserved for the calculated variance value. Any quantities changed in the verification process are stored and are subsequently printed on the output. The input verification step also includes the initialization of the histogram counters as well as the counting of the number of TYPE = 8 variables to be considered.

The third function in the computational process is the conduct of the Monte Carlo experiment and the construction of histograms. This is the heart of the whole computation. The independent variable values are generated, processed through the projectile trajectory module, and analyzed to increment appropriate histogram cell counters. The process is then repeated over and over until one of the following terminating conditions occurs:

- 1) Sampling difficulties for a TYPE = 4 variable occur.
- 2) The number of trial solutions which are outside of the internally generated histogram ranges exceeds a user defined limit.
- 3) The number of trial solutions which are outside of the user supplied histogram ranges exceeds a user defined limit.
- 4) The specified number of experiments have been performed.

Histograms are constructed for both the dependent and independent variables. Reasonable estimates of the ranges of all variables are available from input or are internally computed and cell boundaries are established. During the execution of the Monte Carlo experiments it is possible for one or more dependent variables to fall outside the range covered by its histogram. The results from such an experiment should be rejected from all histograms. This requires the code to take the special precautions described in the next two paragraphs when incrementing the histogram counters.

To accommodate these problems the following scheme is employed. The basic histograms are stored in five integer arrays. Two of these arrays, ID1 and I11, are used to store the count for each cell belonging to the dependent and independent variables, respectively. For example, if each variable range

contains five cells, the counter for the second cell of the third dependent variable is the twelfth location of ID1. The arrays ID1 and I11 allow for a maximum of 20 cells with a maximum of 10 dependent variables and 20 independent variables. Two additional integer arrays, IDT(10) and IIT(20), are used for temporary storage of appropriate counter addresses in the ID1 and I11 arrays. As the independent variable values are generated for each trial solution, the appropriate counter addresses are stored in the IIT array. After the dependent variables have been evaluated, their counter addresses are stored in the IDT array. If all of dependent variable values are within the range of their respective histograms, the addresses in the IDT and IIT arrays are used to increment the appropriate counters in the ID1 and I11 arrays. If one or more of the dependent variable values is not within range, then this trial solution is discarded and none of the interval counters are incremented. In this fashion the histograms count only acceptable experiments. A fifth integer array, IR1, is used to count the frequency with which each of the dependent variables causes experiments to be discarded.

The user may desire histogram ranges for the dependent variables which differ from those internally computed. To accommodate this possibility three additional integer arrays, ID2, I12 and IR2, are included. These arrays are replicas of the basic arrays, ID1, I11 and IR1, and are manipulated in exactly the same manner described above with one exception. The exception is that user specifications in the form of TYPE = 8 variables are used instead of the internally generated ranges. Thus, if the user specifies a TYPE = 8 dependent variable, the routine provides two sets of histograms, one set using internal ranges and one set using the TYPE = 8 specifications.

The fourth and final function performed on the Monte Carlo mode consists of printing the Monte Carlo results. The histogram values are calculated and printed. The sample mean and variance are calculated and printed along with the corresponding values from the Analytical Statistical mode. The output is on a variable by variable basis. Occasionally the program hangs up in the output section with an indicated error of a divide fault. This usually occurs because none of the generated solutions were within the histogram ranges. This is usually due to either a gross input error, or an over restricted range for a TYPE = 8 variable.

B.2.4 HITS Subroutines

The HITS computer code consists of a MAIN program and sixty-two subroutines. This section briefly describes the subroutines and their functional interconnections.

Table B-2 lists and briefly describes each of the subroutines. Further details can be gained from the program listings of Appendix D.

In Figure B-4 is a diagram depicting the subroutine linking arrangement of the HITS code. Referring to Figure B-4, there are four decision blocks labeled D1 through D4, each of which has several branch paths. The paths from the decision blocks are mutually exclusive and whenever the code is exercised only one path is followed. Decision block D1 is used to select the projectile trajectory module. At present, there is only one option ($IY = 1$); the others are dummy facilities. The second decision block, D2, determines which of the four Statistical Processor modes are to be exercised: Single Trajectory, Range Check, Analytical Statistical, or Monte Carlo. Decision block D2 is controlled by the TYPE input data. The decision block D3 is under direct control of the user and affects the decision to use a Taylor series approximation of the Projectile Trajectory module for the Monte Carlo experiments. Decision block D4 determines whether separate histograms are to be generated for user specified histogram ranges. This decision is made on the basis of whether or not any $TYPE = 8$ variables were input.

Figure B-5 diagrams the subroutine linking arrangement for the Projectile Trajectory module. Note the two distinct trajectory calculations based on the real world and fire control projectile characteristics.

B.2.5 Histogram Computer Plots

The HITS computer code is designed to facilitate creation of computer drawn histogram plots. Since computer plot software is hardware specific, the present code doesn't do the plotting. Rather it gathers the data together into one subroutine (HYSPLT) so that automatic plotting can be easily implemented by users who desire it. This section defines the format of the data.

The two-dimensional array TITLE (I,J) contains the titles for the histograms. The second subscript, J, denotes the variable

Table B-2 HITS Subroutine List

SUBROUTINE	FUNCTION
MAIN	Print input data and sequence other subroutine calls.
ARTLU	One-dimensional table look up routine.
A2	Computes $\int_0^x a^2 dx$ using Real World parameters.
A2F	Computes $\int_0^x a^2 dx$ using Fire Control parameters.
NAMES	Block data routine to initialize output variable names.
CHCKIA	Perform check to see if any data is missing.
CHCKIN	Check the input variables: a) establish number of TYPE = 8 variables b) force tolerance = 3σ for TYPE = 2 and 7 variables c) establish tolerance for TYPE = 4 variables.
CNVRT	Convert uniform random numbers to Gaussian random numbers with the interval 0 to 1 as 3σ limits.
CRØSS	Computes the Real World projectile's oscillatory motion and crossrange (horizontal and vertical) velocity and position due to oscillatory motion.
CRØSSF	Computes the Fire Control projectile's oscillatory motion and crossrange (horizontal and vertical) velocity and position due to oscillatory motion.
CRVFT	Generate output variable values using 1st or 2nd order Taylor series expansions of projectile trajectory module.
DØABC	Provide automatic Range Check calculations.
DØ234	Provide Analytical Statistical mode calculations.
EXTRA	Output routine for Range Check variable input data.
FC2987	Fire Control trajectory module interface.

Table B-2 HITS Subroutine List (Cont'd)

SUBROUTINE	FUNCTION
FEXP	Computes double precision exponents with error checking.
FILLA	Calculate mean values and variances from histograms.
FILLIN	Dummy routine to provide interface with time phased data base for missing information.
GQCØR	Computes fifth and sixth terms in Eq. (B.1-7).
GQUC	Computes third and fourth terms in Eq. (B.1-7).
G1795	Dummy alternate trajectory module interface.
G2440	Dummy alternate trajectory module interface.
G2987	Trajectory module interface.
HSTG1	Increment histogram counters using internally estimated variable range values.
HSTG2	Increment histogram counters using user defined (TYPE = 8) variable range values.
HYSPLT	Interface for automatic histogram bar chart plots.
INCARD	Reads data cards and decodes the TYPE numbers
INCØN	Computes algebraic constants using Real World initial conditions.
INCØNF	Computes algebraic constants using Fire Control initial conditions.
INITIL	Selects projectile trajectory module, presets data values and initialize INCARD.
INLHST	Initialize all histogram and error counters to zero.
LØADER	Collects and stores data for automatic histogram bar chart plotter.

Table B-2 HITS Subroutine List (Cont'd)

SUBROUTINE	FUNCTION
MCRL	Sequence Monte Carlo operations and monitor number of trials and number of rejects.
MØVEUP	Internally sorts input data to move variables to top of storage arrays.
NCØV	Locates covariances of independent variables in the C array.
NMEAN	Locates mean values of dependent variables in the C array.
NVAR	Locates variances of dependent variables in the C array.
NVARX	Locates variances of independent variables in B array.
N1D	Locates first order derivatives in C array.
N2D	Locates second order derivatives in C array.
PICK1	Branch routine to select projectile trajectory module.
PRNTMC	Provides printed output of the Monte Carlo results.
QCØR	Computes second term of Eq. (B.1-5) and second term of Eq. (B.1-7).
RANDØM	Formats random numbers.
RANDU	Random number generator for supplying uniform random numbers on interval 0 to 1.
RW2987	Real World trajectory module interface.
SAMPLE	Generates random sample values for all TYPE = 2, 3 or 4 variables.
SPRNT	Formats and prints contents of OE array.
STDDEV	Prints standard deviations with greater precision.
STØREC	Data storage routine for efficient packing of the C array.

Table B-2 HITS Subroutine List (Concl'd)

SUBROUTINE	FUNCTION
TG1795	Dummy alternate trajectory module preset routine.
TG2440	Dummy alternate trajectory module preset routine.
TG2987	Initializes trajectory parameters at preset (default) values.
TRAJT	Computes Real World projectile velocity and range as a function of time, including the influence of oscillatory motion on drag.
TRAJX	Computes Real World projectile velocity and time as a function of range, including the influence of oscillatory motion on drag.
TRAJXF	Computes Fire Control projectile velocity and time as a function of range, including the influence of oscillatory motion on drag.
TVX	Computes projectile velocity and range as a function of time for a particle (no oscillatory drag) trajectory.
T4NTV	Computes mean value, variance and tolerance for arbitrarily distributed variables.
VXT	Computes projectile velocity and time as a function of range for a particle (no oscillatory drag) trajectory.
WIND	Computes projectile horizontal velocity and position brought about by crosswinds.
XXC	Computes the range at which the projectile's oscillatory motion has converged.
ZATAN2	Computes the arctangent of a function, placing the resulting angle in the proper quadrant. Includes provisions to avoid computational errors for the special case of non-oscillatory motion.
ZZ	Computes appropriate value of h as defined by Eq. (C.3-44)

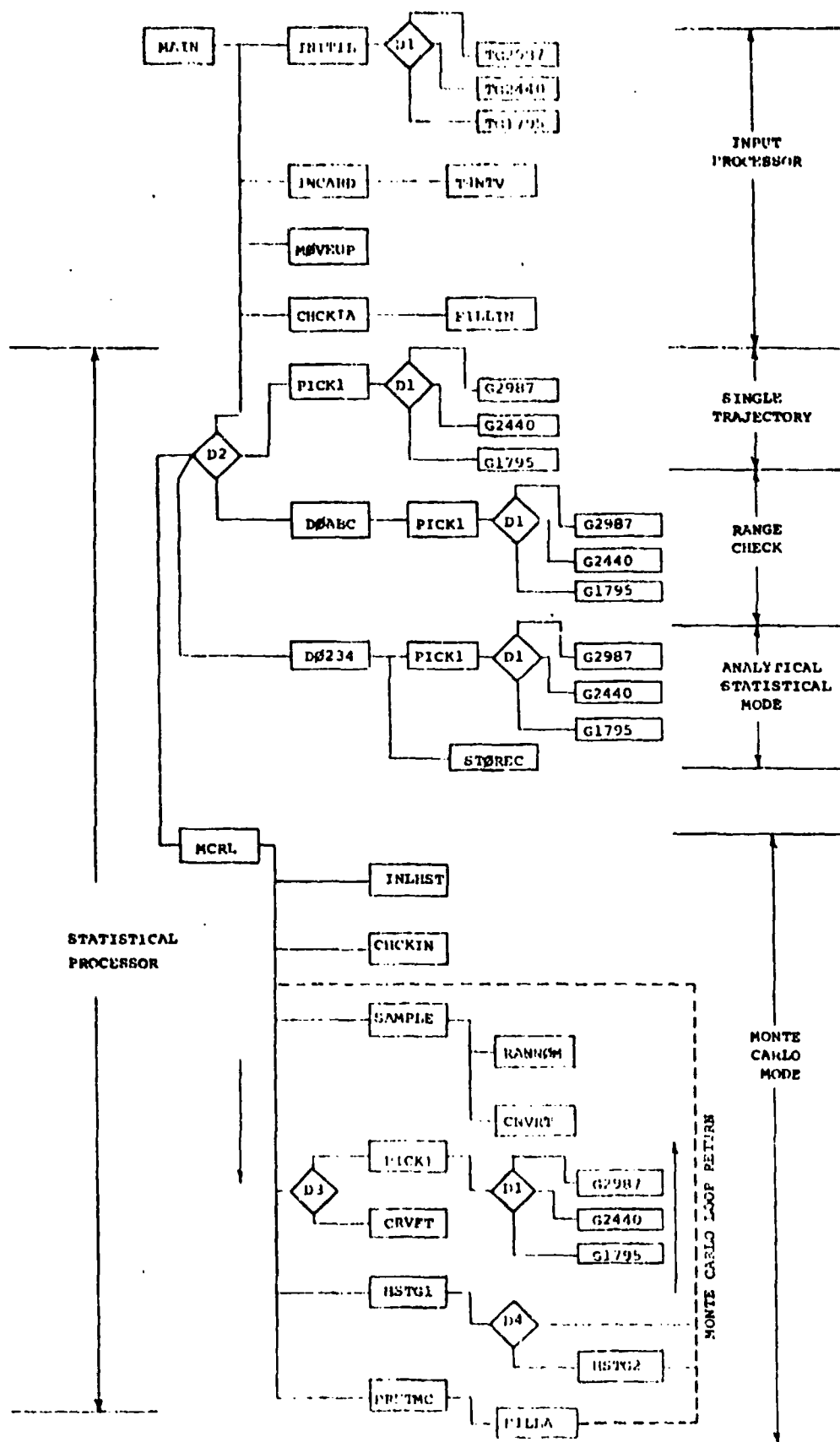


Figure B-4 HITS Subroutine Linking Diagram

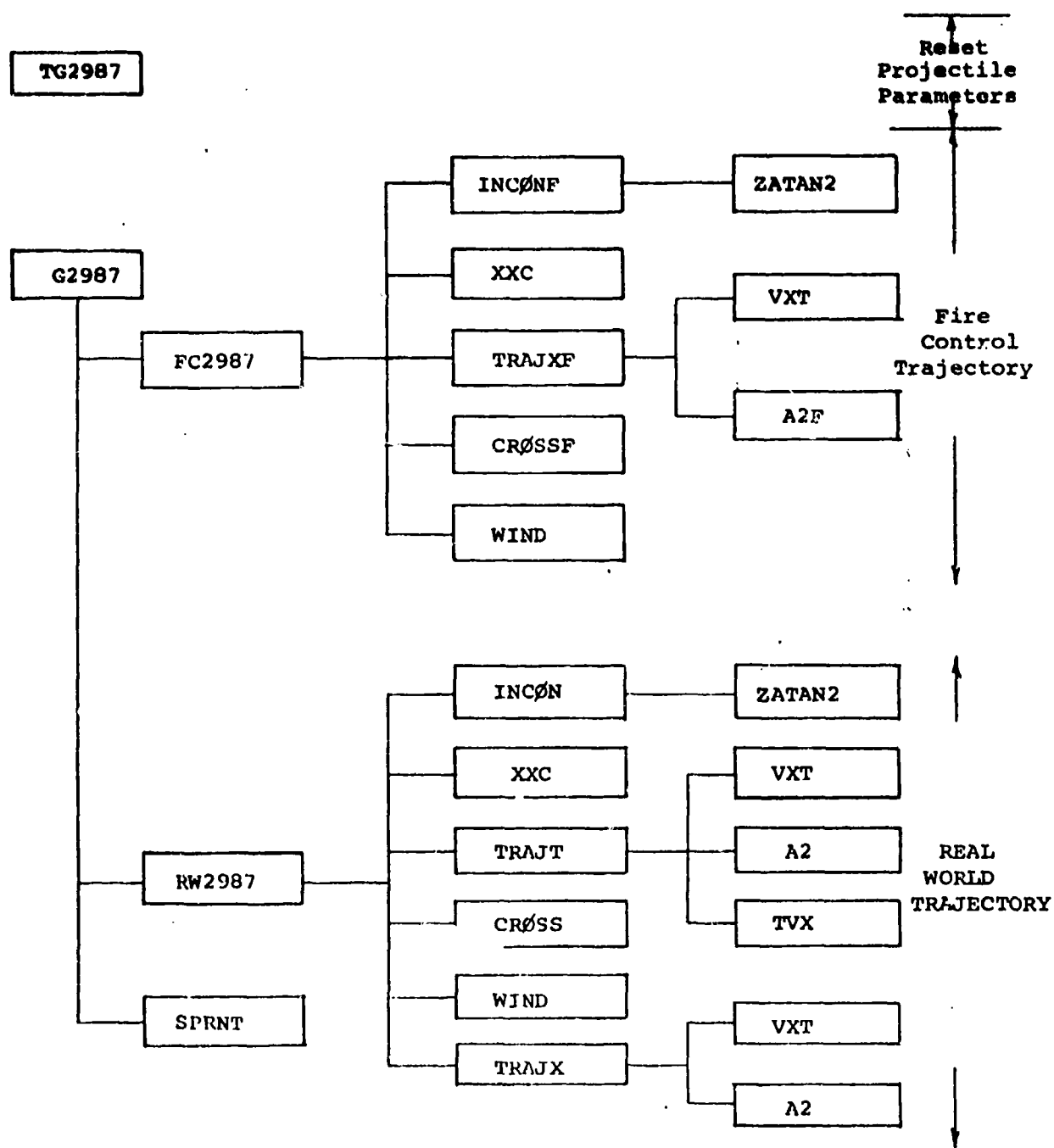


Figure B-5 Projectile Trajectory Module
Subroutine Linking Diagram

associated with the histogram. These are, at most, $KD + K234$ variables, where KD is the number of dependent variables (appearing first) and $K234$ is the number of independent variable (appearing last). The first subscript defines specific items relating to the variable as defined by Figure B-6. The actual histograms are stored in the three-dimensional array $ZDATAD(I,J,K)$. The index K denotes the $KD + K234$ variables with the dependent variables appearing first and the independent last. Figure B-7 illustrates the format of a two dimensional slice of the array, $ZDATAD(I,J,*)$. The upper portion applies to the dependent variables and the lower defines the array for independent variables.

TITLE (I,*)												
ROW I	INTERNAL NAME	DESCRIPTION	REMARKS									
1	IPASS	Histogram Indicator	<table><tr><th>Variable Range</th><th>Dependent</th><th>Independent</th></tr><tr><td>Internal</td><td>1</td><td>2</td></tr><tr><td>User</td><td>3</td><td>4</td></tr></table>	Variable Range	Dependent	Independent	Internal	1	2	User	3	4
Variable Range	Dependent	Independent										
Internal	1	2										
User	3	4										
2	IV	Code Number	X-Axis Label									
3	ITYPE	Variable Type	2, 3, 4, 7 or 8									
4	NRR	Number of Rejects	Rejects									
5	CM	Histogram Mean	MEAN									
6	CV	Histogram Variance	VAR									
7	PM	Estimated Mean	(MEAN)									
8	PV	Estimated Variance	(VAR)									
9	FCTR	Histogram Cell Size	Interval									
10	PT	Tolerance	TOL									
11	NRR	Number of Rejects										
12	CM	Histogram Mean	MEAN									
13	CV	Histogram Variance	VAR									
14	PM	Estimated Mean	(MEAN)									
15	PV	Estimated Variance	(VAR)									
16	FCTR	Histogram Range	Interval									
17	PT	Tolerance	TOL									
18	NCELL	Number of Cells										
19	SAVE	Maximum Relative Frequency										
20	ISM ~~OOD~~	Indicator for Spline Smoothing Option	1 for TYPE = 2, Zero									

Internal Ranges
(IPASS = 1 or 2)

User Ranges
(IPASS = 3 or 4)

Internal Ranges
(IPASS = 1 or 2)

 User Ranges
(IPASS = 3 or 4)

Figure B-6 Histogram Title Array

		Internal Ranges		User Ranges	
		X-Axis	Y-Axis	X-Axis	Y-Axis
I \ J		1	2	3	4
1		X_{b0}	$100 N_0/N_T^*$	X_{b0}	$100 N_0/N_T^*$
2		X_{b1}	$100 N_1/N_T^*$	X_{b1}	$100 N_1/N_T^*$

(a) Dependent Variables

		Internal Ranges		User Ranges	Theoretical
		X-Axis	Y-Axis	Y-Axis	Y-Axis
I \ J		1	2	3	4
1					
2					

(b) Independent Variables

Figure B-7 Histogram Storage Array ZDATAD (I,J,*)

APPENDIX C

CLOSED FORM TRAJECTORY EQUATIONS

This appendix derives closed form expressions for the trajectory of a hypervelocity projectile. The equations define the trajectory in time as well as space. The closed form expressions result from realistic simplifying approximations to the full six degree of freedom (6 DOF) equations of motion. All assumptions were verified by comparison to exact solutions as determined by numerical integration of the equations of motion. In all cases, the closed form equations agreed to within engineering tolerances. The equations are of general interest. Avco has employed them in: (1) the aerodynamic design of a hypervelocity projectile, (2) the interpretation of ballistic range test data, and (3) the evaluation of projectile dispersion. Potential applications include incorporation in operational fire control computers, computation of firing tables, and any other situation requiring rapid, accurate, low-cost trajectory determination. Section C.1 presents background material and an overview of the analytical development. Particle trajectory equations are presented in Section C.2. Sections C.3 and C.4 present cross-range and downrange perturbation equations, respectively, used to correct the basic particle trajectory.

C.1 Introduction

The complete coupled, nonlinear equations of motion of a projectile in free flight cannot be solved in a general closed form. To obtain such solutions, it is necessary to make simplifying approximations. Many such solutions have been developed in the past under various assumptions for application to a wide variety of problems. These include the gross evaluation of range-velocity-time histories, ballistic range data reduction, and the development of jump angle expressions for specific launch disturbances. One of the unique aspects of the trajectory equations presented here is that both downrange as well as crossrange dynamics are taken into account. The key has been to build upon the existing work and to piece together various solutions in such a manner as to accurately describe the downrange and cross-range dynamics of hypervelocity projectiles.

The trajectory model has three parts. The first part describes the basic particle trajectory including the effects of constant velocity winds in both the crossrange and downrange

directions. The second part deals with angle of attack oscillatory motion and subsequent crossrange deflection from the particle trajectory. The final part of the model makes use of the angle of attack history from the oscillatory motion solution to compute range, velocity, and flight time perturbations for correction of the basic particle trajectory. These values account for the increase in drag brought about by angle of attack oscillations. The three parts are then superimposed to describe the complete trajectory history in six degrees of freedom.

The coordinate system used in the trajectory model is shown in Figure C-1. A consistent set of symbols is used throughout this appendix and is presented in Table C-1 along with code numbers which relate the variables to the HITS Code.

C.2 Particle Trajectory

This section presents the basic particle trajectory closed form equations. The development of the equations is discussed in Section C.2.1. Section C.2.2 presents results which verify the equations.

C.2.1 Analytical Development

The particle trajectory solution is fundamentally the same as that developed and reported in an earlier study,¹ with the formulation extended to include the effects of winds in the crossrange and downrange directions. The primary assumptions involved in the solution are the following:

- The drag coefficient varies as $1/v^2$.
- The wind has constant speed and direction.
- The projectile weathercocks into the wind and flies a static zero angle of attack trajectory.
- A uniform density atmosphere.
- The absence of gravity.

The drag coefficient is allowed to vary with velocity according to

$$C_D = C_{D_\infty} + \frac{K_D}{v^2} \quad (C.2-1)$$

¹Avco Systems Division, "A Study of the Feasibility of the Two Stage Light Gas Gun as an Air Defense Weapon," AVSD-0197-73-CR, June 1973.

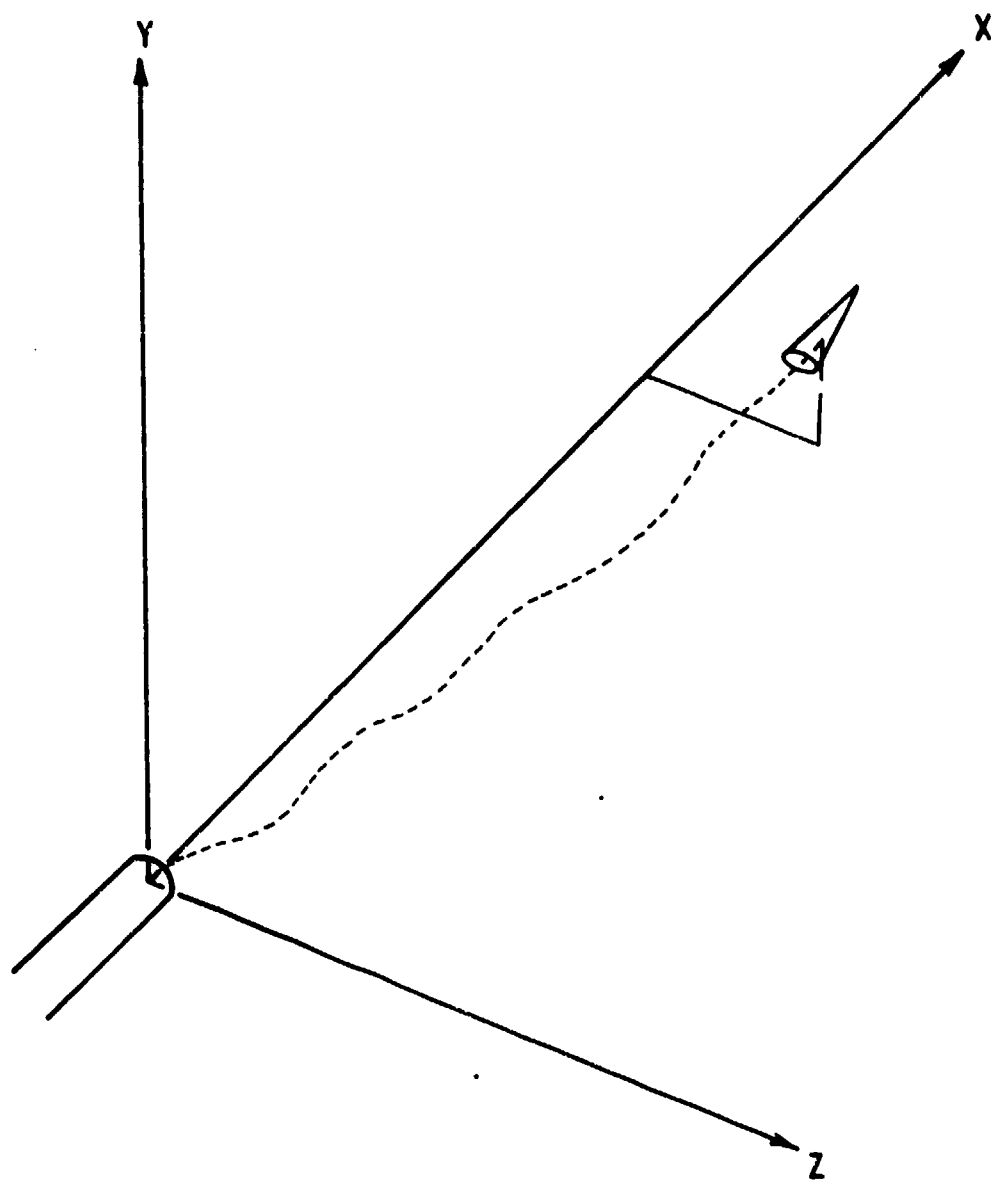


FIGURE C-1 TRAJECTORY COORDINATE SYSTEM

TABLE C-1

LIST OF SYMBOLS

HITS CODE NUMBERS			SYMBOL	DEFINITION AND COMMENTS
R.W. ¹	F.C. ²	C.O. ³		
122	22		A	reference area, ft ²
		404	B	$1 + \left(\frac{W_Z}{V_o - W_X} \right)^2$
			C _D	drag coefficient
		412	C _{D∞}	asymptotic limit of C _D as V → ∞
		419	$\overline{C_{D_a}}$	mean drag coefficient including oscillatory motion effects
		418	$\overline{C_{D_{a-\infty}}}$	mean drag coefficient excluding oscillatory motion effects
			C _{L_a}	$C_{N_a} - \left(C_{D_{a-\infty}} + \frac{K_D}{V_o^2} \right)$
113	13		C _M	pitching moment coefficient about center of gravity, $\frac{M}{\frac{1}{2} \rho V^2 A D}$
		413	C _{M_a}	pitching moment coefficient slope, $\frac{dC_M}{da}$, rad ⁻¹
			C _{M_{P_a}}	magnus moment coefficient, $\frac{\partial^2 C_M}{\partial \left(\frac{PD}{2V} \right) \partial a}$, rad ⁻¹
		415	C _{M_{φ_a}}	$\frac{\partial^2 C_M}{\partial P \partial a}$, sec
112	12		C _{M_q}	pitch damping derivative, $\frac{\partial C_M}{\partial \left(\frac{\phi D}{2V} \right)}$

- 1 Denotes Real World projectile parameter
2 Denotes Fire Control projectile parameter
3 Denotes Computed quantity

TABLE C-1

LIST OF SYMBOLS (Cont'd)

HITS CODE NUMBERS			SYMBOL	DEFINITION AND COMMENTS
R.W.	F.C.	C.Q.		
110	10	414	$C_{M\dot{\theta}}$	$\frac{\partial C_M}{\partial \dot{\theta}}$, sec
			C_M	normal force coefficient, $\frac{F}{\frac{1}{2} \rho V^2 A}$
			C_{N_α}	normal force coefficient slope, $\frac{\partial C_N}{\partial \alpha}$, rad ⁻¹
			C_x	axial force coefficient, $\frac{F}{\frac{1}{2} \rho V^2 A}$
124	24	434	C_{x_0}	axial force coefficient at zero angle of attack
			D	reference length, ft
			f	$\frac{\rho A g}{2W}$, ft ⁻¹
			g	gravitational acceleration, 32.174 ft/sec ²
206	206	435	h	$\frac{C_{N_\alpha} V_0}{m}$ (ICNCL = 0); $\frac{C_{L_\alpha} V_0}{m}$ (ICNCL = 1)
			ICNCL	Aerodynamic coordinate frame selector
			I_x	roll moment of inertia, slug-ft ²
			I_y	pitch moment of inertia, slug-ft ²
137	27	430	I'	$\frac{I_y}{\frac{1}{2} \rho V^2 A D}$, sec ²
			JA	jump angle, radians

TABLE C-1

LIST OF SYMBOLS (Cont'd)

HITS CODE NUMBERS			SYMBOL	DEFINITION AND COMMENTS
R.W.	F.C.	C.Q.		
130	30	471	JA_y	vertical component of jump angle, radians
		472	JA_z	horizontal component of jump angle, radians
			K_{C_x}	constant describing variation of C_x with angle of attack, rad^{-2}
		411	K_D	constant describing variation of C_x with velocity, sec^2/ft^2
123	23	416, 417	K_1, K_2	parameters defined by Eqs. (C.3-14) and (C.3-15)
			L	projectile length, ft
		431	m'	$\frac{2W}{\rho A_0 V_0}$, sec
128	28		P	projectile spin rate, rad/sec
			P_{cr}	projectile resonant spin rate, rad/sec
		441, 442, 443, 444	R_1, R_2 R_3, R_4	parameters defined by Eqs. (C.3-16) through (C.3-19)
		445	R_{trim}	static trim angle amplification factor due to spin
111	11		SM	projectile static margin, $\frac{x_{cp} - x_{cg}}{L}$
			t	flight time, sec
		450	t'	$\frac{\pi}{V_0}$, sec
		500	t_n	nominal flight time, sec

TABLE C-1

LIST OF SYMBOLS (Cont'd)

HITS CODE NUMBERS			SYMBOL	DEFINITION AND COMMENTS
R.W.	F.C.	C.Q.		
			Δt	perturbation to flight time arising from oscillatory motion, sec
			V	velocity, ft/sec
			V_x	downrange component of velocity, ft/sec
			V_y	vertical component of velocity, ft/sec
			V_z	horizontal component of velocity, ft/sec
			V_{zw}	projectile horizontal velocity due to crosswind, ft/sec
101	1		V_0	muzzle velocity, ft/sec
			ΔV	velocity perturbation due to oscillatory motion, ft/sec
125	25		W	projectile weight, lbs
102	2		W_x	downrange component of wind velocity, ft/sec
103	3		W_z	crossrange component of wind velocity, ft/sec
			X	range, ft
			X_c	range at angle of attack convergence, ft
			X_{cg}	projectile center of gravity location from nose, ft
			X_{cp}	projectile center of pressure location from nose, ft
			ΔX	downrange error, ft
			Y	vertical displacement, ft

TABLE C-1

LIST OF SYMBOLS (Cont'd)

HITS R.W	CODE NUMBERS		SYMBOL	DEFINITION AND COMMENTS
	F.C.	C.Q.		
114	14	426	$\dot{Y}_0$	component of initial velocity in the Y direction, ft/sec
			Z	horizontal displacement, ft
		427	$\dot{Z}_0$	component of initial velocity in the Z direction, ft/sec
			Z_w	projectile horizontal displacement due to cross-wind, ft
		400	α	pitch angle of attack in body fixed coordinate system, radians
			α_{ST}	static trim pitch angle of attack (zero spin rate), radians
		402	α_{trim}	trim pitch angle of attack when spinning, radians
		476	α_0	initial pitch angle of attack, radians
		403	$\dot{\alpha}_0$	time rate of change of α_0 , rad/sec
		401	$\bar{\alpha}$	total angle of attack, radians
115	15	409	$\bar{\alpha}_{upper}$	upper envelope of $\bar{\alpha}$, radians
		408	β	yaw angle of attack in body, fixed coordinate system, radians
			β_{ST}	static trim yaw angle of attack (zero spin rate), radians
		405	β_{trim}	trim yaw angle of attack when spinning, radians
		406	β_0	initial yaw angle of attack, radians
		407	$\dot{\beta}_0$	time rate of change of β_0 , rad/sec

TABLE C-1

LIST OF SYMBOLS (Concl'd)

HITS CODE NUMBERS			SYMBOL	DEFINITION AND COMMENTS
R.W.	F.C.	C.Q.		
116	16		θ_0	initial pitch orientation, radians
118	18		$\dot{\theta}_0$	initial pitching rate, rad/sec
		464,465, 466,429 467,463	$\lambda_0, \lambda_1, \lambda_2, \Delta\lambda$	components of damping rates, sec^{-1}
			ν_1, ν_2	phase angles defined by Eqs. (C.3-24) and (C.3-25), respectively, radians
104	4		ρ	atmospheric density, slugs/ft ³
		437,438, 439	ϕ_0, ϕ_1, ϕ_2	phase angles defined by Eqs. (C.4-15), (C.3-45), and (C.3-46), respectively, radians
			ϕ_{ST}	meridional orientation of static trim, radians
			$\Delta\phi$	meridional shift of static trim orientation due to spin, radians
117	17		ψ_0	initial yaw orientation, radians
119	19		$\dot{\psi}_0$	initial yaw rate, rad/sec
		460,461, 462,428	$\omega_0, \omega_1, \omega_2, \Delta\omega$	components of oscillation frequency defined by Eqs. (C.3-7), (C.3-3), (C.3-4), and (C.3-8), respectively; rad/sec

where $C_{D\infty}$ and K_D are constants evaluated for the specific projectile and velocity regime. Knowledge of the projectile drag coefficient at two velocities within the range of interest allows determination of the two constants $C_{D\infty}$ and K_D . The drag coefficient approximation may also be expressed in terms of the ballistic coefficient, β (i.e., $\beta = W/C_D A$), as indicated in Figure C-2 which shows the accuracy of a typical fit.

With the inclusion of constant velocity winds, the previous¹ particle trajectory solution takes the form

$$V_x = \left\{ (V_o - W_x)^2 e^{2f\sqrt{B} C_{D\infty} (W_x t - x)} + \frac{K_D}{B C_{D\infty}} \left(e^{2f\sqrt{B} C_{D\infty} (W_x t - x)} - 1 \right) \right\}^{1/2} + W_x \quad (C.2-2)$$

$$t = \frac{1}{f\sqrt{C_{D\infty} K_D}} \left\{ \tan^{-1} \left[\sqrt{\frac{B C_{D\infty}}{K_D}} (V_o - W_x) \right] - \tan^{-1} \left[\sqrt{\frac{B C_{D\infty}}{K_D}} (V_x - W_x) \right] \right\} \quad (C.2-3)$$

$$V_{zw} = W_z \left[1 - \frac{V_x - W_x}{V_o - W_x} \right] \quad (C.2-4)$$

$$Z_w = \frac{W_z}{V_o - W_x} (V_o t - x) \quad (C.2-5)$$

It is most important to notice that range is the independent variable in the particle trajectory equations. Subsequently, velocity and time-of-flight corrections are computed and applied as a function of range and not time. Equation (C.2-3) is indeterminate for the case of constant drag coefficient ($C_D = C_{D\infty}$ and $K_D = 0$). Numerical evaluation of this case can be made using any sufficiently small value for K_D , such that $K_D/V^2 \ll C_{D\infty}$. In the presence of a wind component in the downrange direction, W_x , an iterative procedure must be used to solve the equations, since the product of the unknown flight time, t , and W_x appears in the expression for velocity, i.e., Eq. (C.2-2).

¹Avco Systems Division, "A Study of the Feasibility of the Two Stage Light Gas Gun as an Air Defense Weapon," AVSD-0197-73-CR, June 1973.

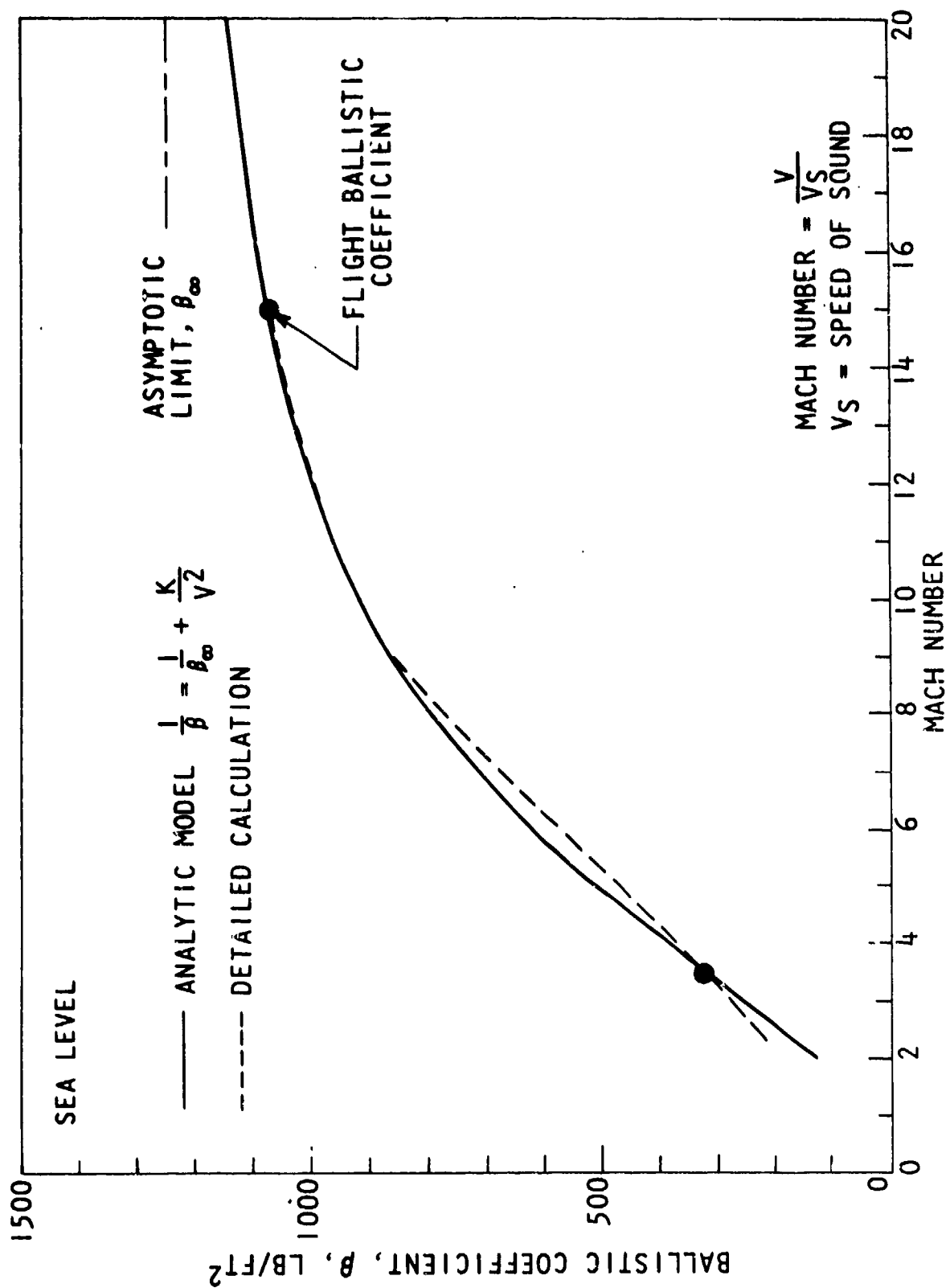


FIGURE C-2 TYPICAL BALLISTIC COEFFICIENT VARIATION WITH MACH NUMBER

C.2.2 Verification

The particle trajectory equations, including the drag coefficient approximation, were verified by comparing them to exact numerical solutions of the 6 DOF equations of motion. Figure C-3 summarizes one comparison. It shows the range-velocity-time histories from the analytic model and the 6 DOF simulation agree.

A separate comparison was made to evaluate the effects of winds on crossrange error. This comparison (as well as all subsequent check cases in this appendix) was made using a smaller projectile whose aerodynamic and mass properties are listed in Table C-2. With a muzzle velocity of 11,000 ft/sec and a constant crossrange wind velocity, W_z , of 15 ft/sec, the 6 DOF calculations showed an 11.2 ft crossrange drift after 2 seconds of flight. The analytic model agreed to within 2 percent.

C.3 Crossrange Perturbations

This section presents the crossrange perturbation equations that are used to correct the basic particle trajectory. The development of the equations is discussed in Section C.3.1. Section C.3.2 presents results which verify the equations.

C.3.1 Analytical Development

The crossrange perturbation model addresses the oscillatory motion and corresponding crossrange effects. A solution for the angle of attack history was obtained from a NACA report.¹ This model employs the following typical ballistic range assumptions:

- Constant velocity
- Constant spin rate
- Linear aerodynamics
- Small trim angles due to shape or mass asymmetries
- Small angles ($\cos \theta \approx 1$, $\tan \theta \approx \sin \theta \approx \theta$)
- Uniform density atmosphere
- The absence gravity

¹Nelson, Robert L., "The Motion of Rolling Symmetrical Missiles Referred to a Dody-Axis System," NACA Technical Note 3737, November 1956.

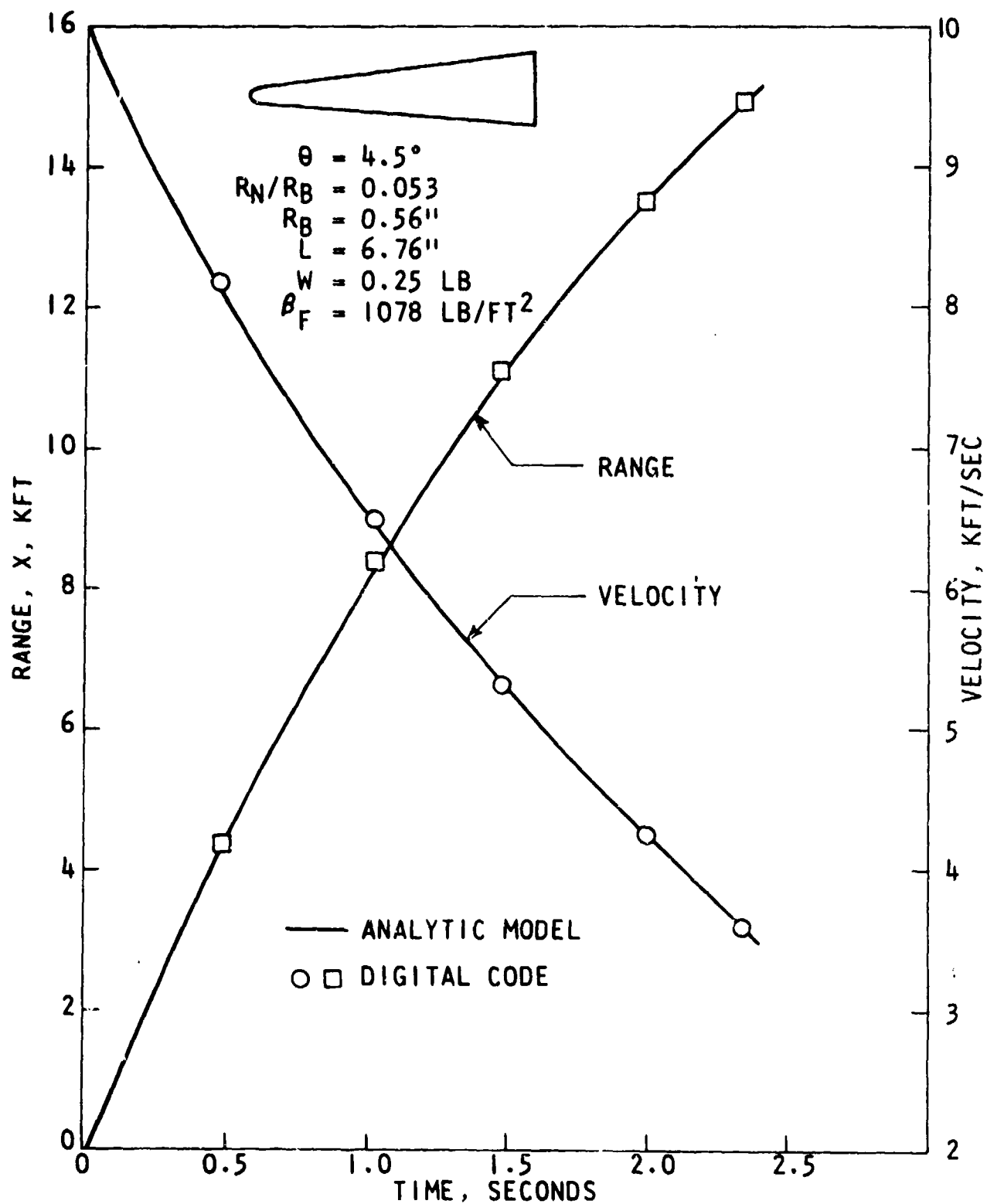


FIGURE C-3 TRAJECTORY COMPARISON

TABLE C-2

TEST CASE PROJECTILE PROPERTIES

CHARACTERISTIC	VALUE
Mass Properties: Cone Angle, θ_c Length, L Nose Radius, R_N Base Radius, R_B Weight, W Roll Moment of Inertia, I_x Pitch Moment of Inertia, I_y Aerodynamic Characteristics Flight Ballistic Coefficient ($M = 15$), δ_F Static Margin, SM	5.5° 3.7" .019" .375" .11 lbs. .538 x 10⁻⁶ slug-ft² .787 x 10⁻⁵ slug-ft² 1000 PSF 6.2% L

At first appearance, the constant velocity assumption may seem objectionable since significant velocity variations occur along the actual trajectory. The constant velocity assumption is required to achieve a closed form solution with a nonzero spin rate. In reality, the method of applying the constant velocity crossrange corrections to the particle trajectory minimizes the errors incurred in calculating the perturbation assuming a constant velocity. Velocity affects crossrange displacement through its influence on the aerodynamic force in the crossrange direction. This force is composed of two components: the drag force and the normal force. For low drag, slender projectiles, the normal force dominates and the drag force can be safely ignored. The normal force is proportional to the angle of attack. Thus, assuming small trim angles of attack, the normal force is only important while the projectile is oscillating in angle of attack. Furthermore, the first few oscillations are the most important because subsequent oscillations tend to average out. Since these occur near the muzzle when velocity is essentially constant, the constant velocity assumption is realistic. The crossrange perturbation is applied to the basic particle trajectory as a function of range. This scales the "constant velocity" crossrange perturbation in accord with the downrange velocity variation of the particle trajectory. Thus, the crossrange perturbation equations simulate the true velocity variation.

The solution given in the NACA report¹ is in terms of the body fixed coordinate system shown in Figure C-4. The form of the solution is

$$\alpha = K_1 e^{\lambda_1 t'} \sin(\omega_1 t' + \nu_1) - K_2 e^{\lambda_2 t'} \sin(\omega_2 t' - \nu_2) + \alpha_{trim} \quad (C.3-1)$$

$$\beta = K_1 e^{\lambda_1 t'} \cos(\omega_1 t' + \nu_1) + K_2 e^{\lambda_2 t'} \cos(\omega_2 t' - \nu_2) + \beta_{trim} \quad (C.3-2)$$

where α and β are, respectively, the pitch and yaw body angles of attack as illustrated in Figure C-4. These equations are written in terms of the range expressed as a pseudo flight time, $t' = X/V_0$. This effects the appropriate scaling between

¹Nelson, Robert L., "The Motion of Rolling Symmetrical Missiles Referred to a Body-Axis System," NACA Technical Note 3737, November 1956.

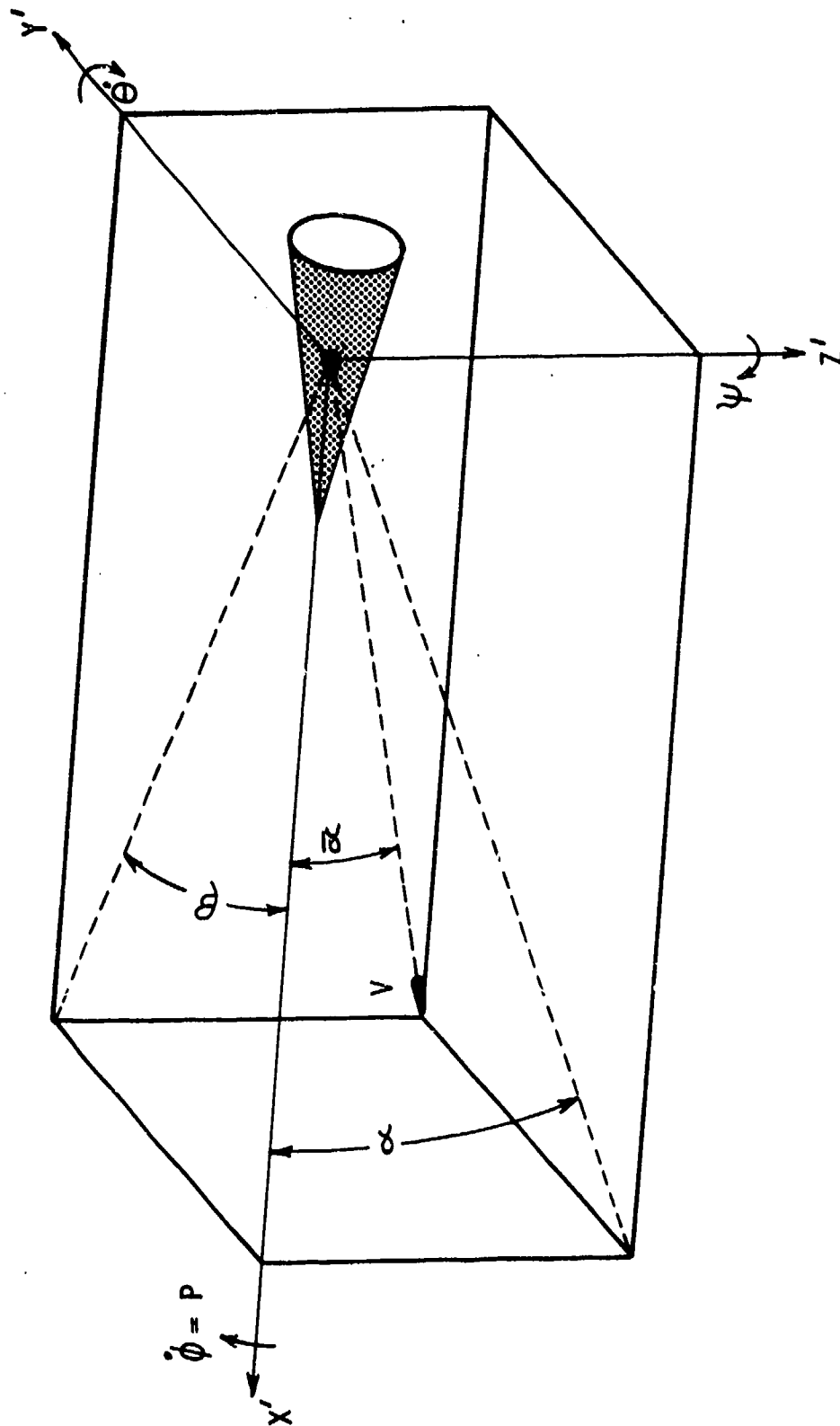


FIGURE C-4 PROJECTILE BODY FIXED COORDINATE SYSTEM

the crossrange perturbation solution assuming constant velocity and the realistic velocity variations contained in the particle trajectory. The frequency and damping constants are defined as follows:

$$\omega_1 = \omega_0 - \Delta\omega \quad (\text{C.3-3})$$

$$\omega_2 = \omega_0 + \Delta\omega \quad (\text{C.3-4})$$

$$\lambda_1 = \lambda_0 + \Delta\lambda \quad (\text{C.3-5})$$

$$\lambda_2 = \lambda_0 - \Delta\lambda \quad (\text{C.3-6})$$

where

$$\omega_0 = \frac{\sqrt{2}}{4} \left\{ -\frac{4C_{M_a}}{I'} + \left(P \frac{I_x}{I_y} \right)^2 - \left(\frac{C_{N_a}}{m'} + \frac{C_{M_{\dot{\theta}}}}{I'} \right)^2 + \left[\left\{ -\frac{4C_{M_a}}{I'} + \left(P \frac{I_x}{I_y} \right)^2 - \left(\frac{C_{N_a}}{m'} + \frac{C_{M_{\dot{\theta}}}}{I'} \right)^2 \right\}^2 + 4 \left[P \frac{I_x}{I_y} \left(\frac{C_{N_a}}{m'} + \frac{C_{M_{\dot{\theta}}}}{I'} \right) + 2P \frac{C_{M_{\dot{\phi}_a}}}{I'} \right]^2 \right]^{1/2} \right\}^{1/2} \quad (\text{C.3-7})$$

$$\Delta\omega = P \left(1 - \frac{I_x}{2I_y} \right) \quad (\text{C.3-8})$$

$$\lambda_0 = \frac{1}{2} \left(\frac{C_{M_{\dot{\theta}}}}{I'} - \frac{C_{N_a}}{m'} \right) \quad (\text{C.3-9})$$

$$\Delta\lambda = \frac{P I_x}{4\omega_0 I_y} \left(\frac{C_{N_a}}{m'} + \frac{C_{M_{\dot{\theta}}}}{I'} \right) + \frac{P C_{M_{\dot{\phi}_a}}}{2\omega_0 I'} \quad (\text{C.3-10})$$

where

$$C_{M_a} = -C_{N_a} (\text{SM}) \frac{L}{D} \quad (\text{C.3-11})$$

$$I' = \frac{2I_y}{\rho A D V_0^2} \quad (\text{C.3-12})$$

$$m' = \frac{2W}{\rho A_B V_o} \quad (C.3-13)$$

The terms K_1 and K_2 in Eqs. (C.3-1) and (C.3-2) are constants involving the initial conditions

$$K_1 = \sqrt{R_1^2 + R_2^2} \quad (C.3-14)$$

$$K_2 = \sqrt{R_3^2 + R_4^2} \quad (C.3-15)$$

where

$$R_1 = \frac{\omega_o [\dot{a}_o + \omega_2(\beta_o - \beta_{trim}) - \lambda_2(a_o - a_{trim})] + \Delta\lambda [\dot{\beta}_o - \omega_2(a_o - a_{trim}) - \lambda_2(\beta_o - \beta_{trim})]}{2(\omega_o^2 + \Delta\lambda^2)} \quad (C.3-16)$$

$$R_2 = \frac{-\omega_o [\dot{\beta}_o - \omega_2(a_o - a_{trim}) - \lambda_2(\beta_o - \beta_{trim})] + \Delta\lambda [\dot{a}_o + \omega_2(\beta_o - \beta_{trim}) - \lambda_2(a_o - a_{trim})]}{2(\omega_o^2 + \Delta\lambda^2)} \quad (C.3-17)$$

$$R_3 = \beta_o - \beta_{trim} - R_1 \quad (C.3-18)$$

$$R_4 = a_o - a_{trim} - R_2 \quad (C.3-19)$$

The initial conditions a_o , β_o , $\dot{a}_o$, and $\dot{\beta}_o$ in Eqs. (C.3-16) and (C.3-17) are related to the projectile inertial orientations and rates by

$$a_o = \theta_o - \frac{\dot{Y}_o}{V_o} \quad (C.3-20)$$

$$\beta_o = \frac{\dot{z}_o}{V_o} - \psi_o \quad (C.3-21)$$

$$\dot{a}_o = \dot{\theta}_o - \frac{\rho \Lambda g}{2w} C_{N_a} V_o a_o - P \beta_o \quad (C.3-22)$$

$$\dot{\beta}_o = -\dot{\psi}_o - \frac{\rho \Lambda g}{2w} C_{N_a} V_o \beta_o + P a_o \quad (C.3-23)$$

The phase angles ν_1 and ν_2 in Eqs. (C.3-1) and (C.3-2) are computed according to

$$\nu_1 = \sin^{-1} \left(\frac{R_2}{K_1} \right) = \cos^{-1} \left(\frac{R_1}{K_1} \right) \quad (C.3-24)$$

$$\nu_2 = \sin^{-1} \left(\frac{R_4}{K_2} \right) = \cos^{-1} \left(\frac{R_3}{K_2} \right) \quad (C.3-25)$$

Both the sine and cosine definitions are given here to establish the proper quadrants in which the phase angles ν_1 and ν_2 lie.

The terms a_{trim} and β_{trim} in Eqs. (C.3-1), (C.3-2), (C.3-16), and (C.3-17) are body fixed trim values for a spinning body (i.e., the "rolling trim") and are related to the body fixed static trim values, a_{ST} and β_{ST} , by

$$a_{trim} = R_{trim} \sqrt{a_{ST}^2 + \beta_{ST}^2} \cos(\phi_{ST} + \Delta\phi) \quad (C.3-26)$$

$$\beta_{trim} = R_{trim} \sqrt{a_{ST}^2 + \beta_{ST}^2} \sin(\phi_{ST} + \Delta\phi) \quad (C.3-27)$$

where

$$\phi_{ST} = \sin^{-1} \frac{\beta_{ST}}{\sqrt{a_{ST}^2 + \beta_{ST}^2}} = \cos^{-1} \frac{a_{ST}}{\sqrt{a_{ST}^2 + \beta_{ST}^2}} \quad (C.3-28)$$

From the work presented in an AIAA journal,¹ R_{trim} and $\Delta\phi$ are given by

$$P_{trim} = \left[\left(1 - \left\{ \frac{P}{P_{CR}} \right\}^2 \right)^2 + \left(2\mu \frac{P}{P_{CR}} \right)^2 \right]^{-1/2} \quad (C.3-29)$$

$$\Delta\phi = \begin{cases} \tan^{-1} \left[\frac{2\mu \left(\frac{P}{P_{CR}} \right)}{1 - \left(\frac{P}{P_{CR}} \right)^2} \right] & \text{for } \left[1 - \left(\frac{P}{P_{CR}} \right)^2 \right] > 0 \\ \pi + \tan^{-1} \left[\frac{2\mu \left(\frac{P}{P_{CR}} \right)}{1 - \left(\frac{P}{P_{CR}} \right)^2} \right] & \text{for } \left[1 - \left(\frac{P}{P_{CR}} \right)^2 \right] < 0 \end{cases} \quad (C.3-30)$$

where

$$P_{CR} = \left[\frac{-C_{M_a} \rho V_0^2 AD}{2(I_y - I_x)} \right]^{1/2} \quad (C.3-31)$$

$$\mu = \frac{\frac{C_{N_a} \rho A b}{4w} - \frac{C_{M_q} D^2 \rho A}{8(I_y - I_x)}}{\left[\frac{-C_{M_a} \rho AD}{2(I_y - I_x)} \right]^{1/2}} \quad (C.3-32)$$

¹Pettus, Joseph J., "Persistent Reentry Vehicle Roll Resonance," American Institute of Aeronautics and Astronautics Paper No. 66-49, January 1966.

The term R_{trim} represents the amplification factor for a spin rate, P , close to the critical (or resonant) frequency, P_{CR} . The term $\Delta\phi$ represents the meridional shift of the plane of the rolling trim depending on the value of the spin rate with respect to P_{CR} . At resonance conditions ($P = P_{CR}$), the amplification is maximum and the rolling trim plane is shifted by 90° with respect to the plane of the static trim.

With the angle of attack history established, the lift force is known. The crossrange displacement is found by integration of the equations of motion normal to the flight path. The remainder of this section is devoted to this integration.

Because of the spinning motion of the projectile, the body fixed coordinate system in which α and β are described, see Figure C-4, rotates with respect to the inertial frame in which the trajectory is described, see Figure C-1. At exit from the muzzle ($t = 0$) the two coordinate systems are aligned. At this instant, positive α is nose up and creates a lift force in the vertical direction which is the positive y direction in the trajectory frame. Owing to the nature of the body fixed coordinate system, positive β at $t = 0$ is nose left and produces a horizontal yaw force in the negative z direction. With positive spin being "right wing down," the body fixed lift and yaw forces rotate with respect to the trajectory coordinate frame as illustrated in Figure C-5. The force components in the inertial trajectory coordinate system are written as follows.¹

$$L_\alpha = \frac{C_{N_\alpha} \rho A V_0^2}{2} \alpha \quad (C.3-33)$$

$$L_\beta = \frac{C_{N_\beta} \rho A V_0^2}{2} \beta \quad (C.3-34)$$

$$\phi = P t' \quad (C.3-35)$$

$$F_y = L_\alpha \cos \phi + L_\beta \sin \phi \quad (C.3-36)$$

$$F_z = L_\alpha \sin \phi - L_\beta \cos \phi \quad (C.3-37)$$

¹ In most applications, it is more appropriate to use C_{L_α} instead of C_{N_α} in the ensuing development. This affects only the definition of h , as noted in Eq. (C.3-44).

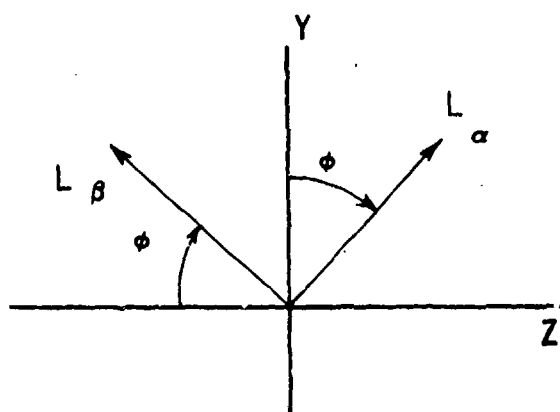


FIGURE C-5 LIFT FORCE COMPONENTS IN THE TRAJECTORY FRAME

Thus the equations of motion are

$$\frac{w}{g} \frac{d^2 Y}{dt'^2} = \frac{C_{N_a} \rho A V_o^2}{2} [\alpha \cos P t' + \beta \sin P t'] \quad (C.3-38)$$

$$\frac{w}{g} \frac{d^2 Z}{dt'^2} = \frac{C_{N_a} \rho A V_o^2}{2} [\alpha \sin P t' - \beta \cos P t'] \quad (C.3-39)$$

where α and β are given by Eqs. (C.3-1) and (C.3-2). The notation t' is maintained in Eqs. (C.3-38) and (C.3-39) since it is not the true flight time.

Integrating Eqs. (C.3-38) and (C.3-39) gives the solutions for crossrange velocity and displacement brought about by oscillatory motion

$$\begin{aligned} V_y = h \frac{V}{V_o} & \left\{ \frac{K_1}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} \left[e^{\lambda_1 t'} \sin [(\omega_1 + P)t' + \nu_1 + \phi_1] - \sin (\nu_1 + \phi_1) \right] \right. \\ & - \frac{K_2}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} \left[e^{\lambda_2 t'} \sin [(\omega_2 - P)t' - \nu_2 + \phi_2] - \sin (\phi_2 - \nu_2) \right] \\ & \left. + \frac{a_{trim}}{P} \sin P t' - \frac{\beta_{trim}}{P} (\cos P t' - 1) + \frac{1}{h} \frac{dy}{dt} \right\}_o \end{aligned} \quad (C.3-40)$$

$$\begin{aligned}
V_z = h \frac{v}{v_o} & \left\{ \frac{-K_1}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} \left[e^{\lambda_1 \tau'} \cos [(\omega_1 + P) \tau' + \nu_1 + \phi_1] - \cos (\nu_1 + \phi_1) \right] \right. \\
& - \frac{K_2}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} \left[e^{\lambda_2 \tau'} \cos [(\omega_2 - P) \tau' - \nu_2 + \phi_2] - \cos (\phi_2 - \nu_2) \right] \\
& \left. - \frac{\beta_{trim}}{P} \sin P \tau' - \frac{a_{trim}}{P} (\cos P \tau' - 1) + \frac{1}{h} \frac{dz}{dt} \right\}_o
\end{aligned}$$

(C.3-41)

$$\begin{aligned}
Y = h & \left\{ \frac{K_1}{\lambda_1^2 + (\omega_1 + P)^2} e^{\lambda_1 \tau'} \sin [(\omega_1 + P) \tau' + \nu_1 + 2\phi_1] \right. \\
& - \frac{K_2}{\lambda_2^2 + (\omega_2 - P)^2} e^{\lambda_2 \tau'} \sin [(\omega_2 - P) \tau' - \nu_2 + 2\phi_2] \\
& - \left[\frac{K_1 \sin (\nu_1 + \phi_1)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} - \frac{K_2 \sin (\phi_2 - \nu_2)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} - \frac{1}{h} \frac{dy}{dt} \right]_o \tau' \\
& - \frac{K_1 \sin (\nu_1 + 2\phi_1)}{\lambda_1^2 + (\omega_1 + P)^2} + \frac{K_2 \sin (2\phi_2 - \nu_2)}{\lambda_2^2 + (\omega_2 - P)^2} \\
& \left. + a_{trim} \left[\frac{1 - \cos P \tau'}{P^2} \right] + \beta_{trim} \left[\frac{P \tau' - \sin P \tau'}{P^2} \right] \right\}
\end{aligned}$$

(C.3-42)

$$\begin{aligned}
Z = h & \left\{ \frac{-K_1}{\lambda_1^2 + (\omega_1 + P)^2} e^{\lambda_1 t'} \cos [(\omega_1 + P)t' + \nu_1 + 2\phi_1] \right. \\
& - \frac{K_2}{\lambda_2^2 + (\omega_2 - P)^2} e^{\lambda_2 t'} \cos [(\omega_2 - P)t' - \nu_2 + 2\phi_2] \\
& + \left[\frac{K_1 \cos (\nu_1 + \phi_1)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} + \frac{K_2 \cos (\phi_2 - \nu_2)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} + \frac{1}{h} \frac{dz}{dt} \right]_0 t' \\
& + \frac{K_1 \cos (\nu_1 + 2\phi_1)}{\lambda_1^2 + (\omega_1 + P)^2} + \frac{K_2 \cos (2\phi_2 - \nu_2)}{\lambda_2^2 + (\omega_2 - P)^2} \\
& \left. - \beta_{\text{trim}} \left[\frac{1 - \cos P t'}{P^2} \right] + a_{\text{trim}} \left[\frac{P t' - \sin P t'}{P^2} \right] \right\}
\end{aligned}
\tag{C.3-43}$$

where¹

$$h = \begin{cases} \frac{C_{N_a} \rho A g V_o^2}{2 W} & \text{ICNCL} = 0 \\ \frac{C_{L_a} \rho A g V_o^2}{2 W} & \text{ICNCL} = 1 \end{cases}
\tag{C.3-44}$$

$$\phi_1 = \sin^{-1} \frac{-(\omega_1 + P)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} = \cos^{-1} \frac{\lambda_1}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}}
\tag{C.3-45}$$

$$\phi_2 = \sin^{-1} \frac{-(\omega_2 - P)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} = \cos^{-1} \frac{\lambda_2}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}}
\tag{C.3-46}$$

I. All numerical verification results presented in this appendix use C_{N_a} . This is a worst case assumption for the reference projectile.

again take note of the respective quadrants for ϕ_1 and ϕ_2 .

The "jump angle" is defined as the root sum of the squares of the coefficients of the linear term in Eqs. (C.3-42) and (C.3-43) divided by the velocity. It is

$$J A = \sqrt{J A_y^2 + J A_z^2} \quad (C.3-47)$$

where

$$J A_y = \frac{h}{V_o} \left[\frac{K_2 \sin(\phi_2 - \nu_2)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} - \frac{K_1 \sin(\nu_1 + \phi_1)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} \right] \quad (C.3-48)$$

$$J A_z = \frac{h}{V_o} \left[\frac{K_1 \cos(\nu_1 + \phi_1)}{\sqrt{\lambda_1^2 + (\omega_1 + P)^2}} + \frac{K_2 \cos(\phi_2 - \nu_2)}{\sqrt{\lambda_2^2 + (\omega_2 - P)^2}} \right] \quad (C.3-49)$$

C.3.2 Verification

Verification of the crossrange perturbation equations was established by comparison to four 6 DOF simulations. The results are summarized in Table C-3. In all cases, the analytic solution agrees to within 6%. An additional comparison is shown in Figure C-6, in which crossrange error due to trim angle of attack is shown as a function of spin rate. The analytic model is seen to be in excellent agreement with the 6 DOF calculations.

C.4 Downrange Perturbations

This section presents the downrange perturbation equations that are used to correct the basic particle trajectory. The development of the equations is discussed in Section C.4.1. Section C.4.2 discusses results which verify the equations.

C.4.1 Analytical Development

The downrange perturbation equations model the downrange effects of angle of attack oscillations. In addition to the assumptions stated in Section C.3.1, the following assumptions are required to achieve an accurate closed form solution:

TABLE C-3

TRAJECTORY MODEL VERIFICATION $V_0 = 11000 \text{ ft/sec}$

Flight Time = 0.2 sec

CASE	θ_0	ϕ	$\dot{\theta}_0$ (RAD/SEC)	$\dot{\phi}$ (RAD/SEC)	θ	t	P (RAD/SEC)		X (FT)	Y (FT)	Z (FT)	V (FT/SEC)	JUMP ANGLE (DEGS)
1	0	0	0	0	0	0	0	6 DOF MODEL	2091.4 2091.4	0.0 0.0	0.0 0.0	9937.0 9937.0	0.0 0.0
2	10°	0	0	0	0	0	400	6 DOF MODEL	2084.4 2085.2	.37 .39	-.13 -.13	9903.5 9902.4	.19 .20
3	0	0	100	0	0	0	400	6 DOF MODEL	2090.1 2090.1	2.75 2.81	0 0	9931.8 9930.5	1.32 1.35
4	2	3	-50	70	0.2	0	200	6 DOF MODEL	2089.7 2089.8	-1.29 -1.35	2.59 2.68	9931.8 9928.5	1.39 1.44

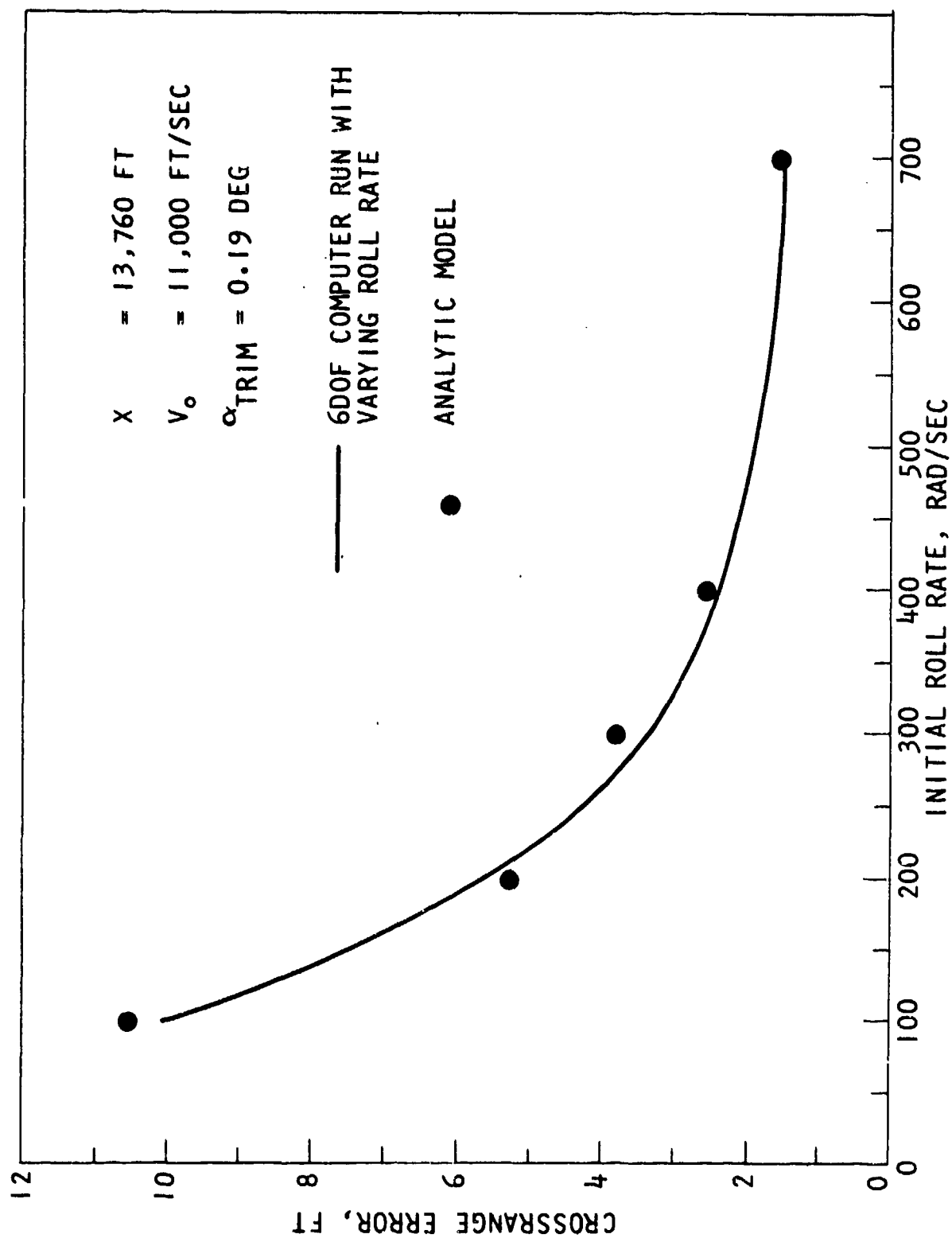


FIGURE C-6 LIFT INDUCED DISPERSION AS A FUNCTION OF ROLL RATE

- Trim angles of attack cause negligible drag.
- Downrange effects can be captured by mean drag coefficients.
- Pitch oscillations are lightly damped.

In addition to the crossrange perturbation, oscillatory motion has a downrange effect. Angle of attack oscillations cause the projectile to experience a net increase in drag. The exact mechanism is illustrated in Figure C-7. As shown in Figure C-7(a), drag is defined as the force acting in line with the velocity vector. There are two contributions when the body is at angle of attack, a component of the normal force, C_N , acts in the drag direction, increasing the drag force. The second component is the axial force coefficient, C_x , which in general is a function of angle of attack, as illustrated in Figure C-7(b). The drag coefficient is

$$C_D = C_x \cos \alpha + C_N \sin \alpha \quad (C.4-1)$$

which, for small angles, becomes

$$C_D \approx C_x + C_N \alpha \quad (C.4-2)$$

Modeling the axial force as parabolic in angle of attack, $C_x = C_{x_0} + K_{C_x} \alpha^2$, and the normal force as linear in angle of attack (i.e., $C_N = C_{N_a} \alpha$) the drag coefficient becomes

$$C_D = C_{x_0} + (C_{N_a} + K_{C_x}) \alpha^2 \quad (C.4-3)$$

The term C_{x_0} is the value of the drag coefficient at zero angle of attack. It is given by the drag model of Section C.2.1, Eq. (C.2-1). For the general case in which oscillations occur in both the pitch and yaw planes, the angle of attack in these expressions becomes the total angle of attack (i.e., the angle between the body axis and the velocity vector), denoted hereafter as $\bar{\alpha}$. For small angles

$$\bar{\alpha}^2 = \alpha^2 + \beta^2 \quad (C.4-4)$$

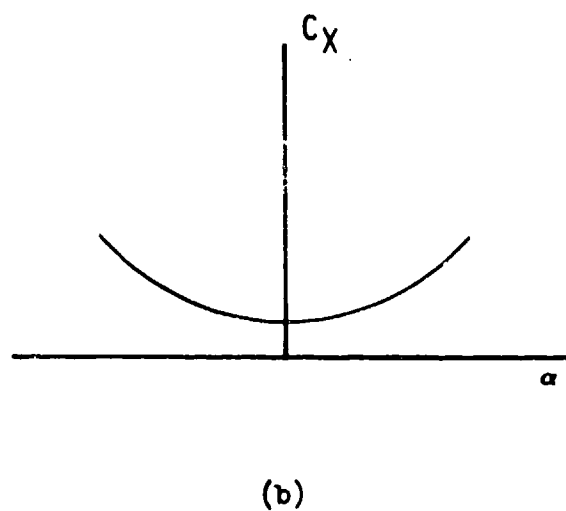
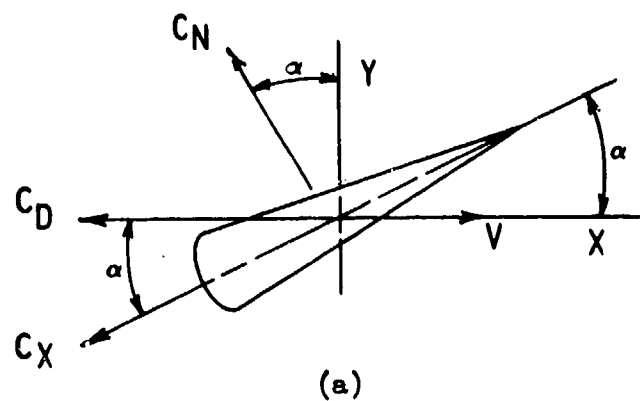


FIGURE C - 7 DOWNRANGE FORCES DUE TO ANGLE OF ATTACK

where α and β are, respectively, the pitch and yaw angles of attack in the body fixed coordinate system. Upon substitution of Eqs. (C.3-1) and (C.3-2) into Eq. (C.4-4), the total attack (with no trim) becomes

$$\bar{\alpha}^2 = K_1^2 e^{\frac{2\lambda_1 x}{V_0}} + K_2^2 e^{\frac{2\lambda_2 x}{V_0}} + 2K_1 K_2 e^{\frac{2\lambda_0 x}{V_0}} \cos\left(\frac{2\omega_0 x}{V_0} + \nu_1 - \nu_2\right) \quad (C.4-5)$$

where t' has been replaced by X/V_0 . The trim angle of attack is neglected so $\bar{\alpha}^2$ may be expressed in a concise, workable form. This is a reasonable approximation since in reality the trim angles are small and do not have an appreciable effect on the drag.

Substitution of Eq. (C.4-5) into Eq. (C.4-3), yields the drag coefficient

$$C_D = C_{D_\infty} + \frac{K_D}{V^2} + (C_{N_\alpha} + K_{C_x}) \left[K_1^2 e^{\frac{2\lambda_1 x}{V_0}} + K_2^2 e^{\frac{2\lambda_2 x}{V_0}} + 2K_1 K_2 e^{\frac{2\lambda_0 x}{V_0}} \cos\left(\frac{2\omega_0 x}{V_0} + \nu_1 - \nu_2\right) \right] \quad (C.4-6)$$

The downrange equation of motion is

$$\frac{w}{g} \frac{d^2 x}{dt^2} = - \frac{\rho A V^2}{2} \left\{ C_{D_\infty} + \frac{K_D}{V^2} + (C_{N_\alpha} + K_{C_x}) \left[K_1^2 e^{\frac{2\lambda_1 x}{V_0}} + K_2^2 e^{\frac{2\lambda_2 x}{V_0}} + 2K_1 K_2 e^{\frac{2\lambda_0 x}{V_0}} \cos\left(\frac{2\omega_0 x}{V_0} + \nu_1 - \nu_2\right) \right] \right\} \quad (C.4-7)$$

which cannot be solved in closed form. As a result, perturbations are calculated for use as corrections. The perturbations are based on mean values of the drag coefficient with and without oscillations, where the mean drag coefficient is defined as

$$\bar{C}_D = \frac{\int_0^x C_D dx}{\int_0^x dx} \quad (C.4-8)$$

For the case without oscillations (zero angle of attack)

$$\bar{C}_{D_{\alpha=0}} = \frac{\int_0^x \left(C_{D_{\infty}} + \frac{K_D}{V^2} \right) dx}{\int_0^x dx} \quad (C.4-9)$$

The equation of motion becomes

$$\frac{w}{g} \frac{dV}{dt} = \frac{w}{g} V \frac{dV}{dx} = - \frac{\rho A}{2} \left(C_{D_{\infty}} + \frac{K_D}{V^2} \right) V^2 \quad (C.4-10)$$

or equivalently

$$\left(C_{D_{\infty}} + \frac{K_D}{V^2} \right) dx = - \frac{2w}{\rho A g} \frac{dV}{V} \quad (C.4-11)$$

Substituting Eq. (C.4-11) into the Eq. (C.4-9) and integrating yields

$$\bar{C}_{D_{\alpha=0}} = \frac{1}{x} \frac{2w}{\rho A g} \ln \frac{V_0}{V} \quad (C.4-12)$$

For the case with oscillatory motion, the mean drag coefficient is

$$\bar{C}_{D_{\alpha}} = \bar{C}_{D_{\alpha=0}} + \frac{C_{N_{\alpha}} + K_{C_x}}{x} \int_0^x \bar{a}^2 dx \quad (C.4-13)$$

Upon substituting the expression for $\bar{a}^2$, Eq. (C.4-7), and integrating

$$\begin{aligned} \overline{C}_{D_a} = \overline{C}_{D_{a=0}} + \frac{C_{N_a} + K_{C_x}}{x} & \left\{ \frac{K_1^2 V_o}{2\lambda_1} \left[e^{\frac{2\lambda_1 x}{V_o}} - 1 \right] + \frac{K_2^2 V_o}{2\lambda_2} \left[e^{\frac{2\lambda_2 x}{V_o}} - 1 \right] \right. \\ & \left. + \frac{V_o K_1 K_2}{\sqrt{\lambda_o^2 + \omega_o^2}} \left[e^{\frac{2\lambda_o x}{V_o}} \cos \left(\frac{2\omega_o x}{V_o} + \nu_1 - \nu_2 + \phi_o \right) - \cos(\nu_1 - \nu_2 + \phi_o) \right] \right\} \end{aligned} \quad (C.4-14)$$

where

$$\phi_o = \sin^{-1} \frac{-\omega_o}{\sqrt{\lambda_o^2 + \omega_o^2}} = \cos^{-1} \frac{\lambda_o}{\sqrt{\lambda_o^2 + \omega_o^2}} \quad (C.4-15)$$

again taking note of the quadrant for ϕ_o .

The mean drag coefficients defined by Eqs. (C.4-12) and (C.4-14) are then used in constant drag coefficient equations for the motion:

$$V = V_o e^{-\frac{\rho A g}{2w} C_D x} \quad (C.4-16)$$

$$t = \frac{2w}{\rho A g} \frac{1}{V_o} \frac{1}{C_D} \left(e^{\frac{\rho A g}{2w} C_D x} - 1 \right) \quad (C.4-17)$$

The two results are subtracted to compute the perturbation quantities ΔV and Δt :

$$\Delta V = V_a - V_{a=0} = V_o \left[e^{-\frac{\rho A g}{2w} \overline{C}_{D_a} x} - e^{-\frac{\rho A g}{2w} \overline{C}_{D_{a=0}} x} \right] \quad (C.4-18)$$

$$\Delta t = t_a - t_{a=0} = \frac{2w}{\rho A g} \frac{1}{V_o} \left[\frac{e^{-\frac{\rho A g}{2w} \overline{C}_{D_a} x} - 1}{\overline{C}_{D_a}} - \frac{e^{-\frac{\rho A g}{2w} \overline{C}_{D_{a=0}} x} - 1}{\overline{C}_{D_{a=0}}} \right] \quad (C.4-19)$$

For values of range, X , prior to angle of attack convergence, these perturbations are applied directly to the velocity and flight time computed from the particle trajectory solution given in Section C.2, Eqs. (C.2-2) and (C.2-3). At values of range beyond the point of convergence, the particle trajectory solution from Eqs. (C.2-2) and (C.2-3) is restarted at the range of convergence, X_C . The initial conditions corresponding to the velocity and time at the convergence range, X_C , are the values for the particle trajectory equations (without angle of attack oscillations) perturbed by ΔV and Δt .

The range at convergence, X_C , is found by determining the point at which the angle of attack oscillations decay to a point where the drag is no longer appreciably affected by angle of attack. From Eq. (C.4-5), the upper envelope of the oscillations is given approximately by

$$\bar{a}_{UPPER} \approx e^{\frac{\lambda_0 x}{V_0}} \left[K_1 e^{\frac{\Delta \lambda}{V_0} x} + K_2 e^{-\frac{\Delta \lambda}{V_0} x} \right] \quad (C.4-20)$$

The damping is assumed to be small compared to the natural frequency, so that many oscillations are required to reach half amplitude. For the purposes of numerical calculations, experience has shown pitch oscillations beyond the point where $\bar{a}_{UPPER} = 0.5^\circ$ have little effect on the trajectory. Thus, the range at convergence is defined as the point at which this occurs. (This is a computer code input parameter and can be easily changed to any other value.)

The downrange error, ΔX , at nominal time, t_n , is computed by iterating the solution in range until the computed flight time matches t_n . If desired, the iteration can be avoided by an additional approximation. The downrange error is closely approximated by

$$\Delta x \approx (t_n - t) V \quad (C.4-21)$$

C.4.2 Verification

Verification of the downrange perturbation equations was established by comparison to four numerical solutions of the

full equations of motion given in Table C-3. The downrange position agrees to within 0.05%. The downrange perturbation, ΔX (the difference between the downrange position with oscillatory motion and the downrange position without oscillatory motion at the same flight time) agrees to approximately 10%.

APPENDIX D

HITS PROGRAM LISTING

This appendix presents the HITS¹ program listing. The code was designed to facilitate conversion to real-time operation from remote key board terminals. The program is written in FORTRAN IV and requires less than 175 K bytes of computer memory on an IBM 360.

The code generates seven (7) warning level (i.e., Level 4) diagnostics when compiled on an IBM-360/75: four (4) in Subroutine DOABC and three (3) in Subroutine DO234. These are to be expected and ignored.

¹ HITS is maintained in the Avco engineering computer code library as Production Code 5127.

WITS0001
WITS0002
WITS0003
WITS0004
WITS0005
WITS0006
WITS0007
WITS0008
WITS0009
WITS0010
WITS0011
WITS0012
WITS0013
WITS0014
WITS0015
WITS0016
WITS0017
WITS0018
WITS0019
WITS0020
WITS0021
WITS0022
WITS0023
WITS0024
WITS0025
WITS0026
WITS0027
WITS0028
WITS0029
WITS0030
WITS0031
WITS0032
WITS0033
WITS0034
WITS0035
WITS0036
WITS0037
WITS0038
WITS0039
WITS0040
WITS0041
WITS0042
WITS0043
WITS0044
WITS0045
WITS0046
WITS0047
WITS0048
WITS0049
WITS0050

[illegible]

- 250 -

D.1 MAIN Program (continued)

```

IMPLICIT REAL*8 (A-H,O-Y)
LOGICAL*1 CARD(80)
COMMON /CARDCM/ B(200), C(5000), IA(200,2), ID(10,2), INC,
1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KTL, K234, LABC(3)
COMMON /CINOM/ INOM
COMMON /CISPRN/ ISPRNT
COMMON /IMCH/ ID1(200), ID2(200), I11(400), I12(400), IDT(10), I1T(20),
1 IRI(10), IR2(10), NTRIAL, NCELL, MCALC, MCOPT, IRJ1, IRJ2, IN8, MCC, NR1, NR2
COMMON /HISPLT/ IOPLT, ISPLT, ZTITLE(20,30), ZDATA(21,4,30)
IOUT = 6
INC = 5
INC = 10
REVIND INC
WRITE (IOUT,9001)
C9001 FORMAT(1 THE INPUT CARDS ... // )
C1 READ (5,9002,END=2) CARD
C9002 FORMAT(80A1) CARD
C WRITE (INC,9002) CARD
C WRITE (IOUT,9003) CARD
C9003 FORMAT(1X, 80A1)
C GO TO 1
C 2 REVIND INC
C 8 CONTINUE
9005 FORMAT(1H1,///)
WRITE(IOUT,9005)
A 3(T9, 4(.H.),T30, 4(.H.),T40,25(.I.),T71,25(.T.),T102,25(.S.),/),HITS0077
B10(T9, 4(.H.),T30, 4(.H.),T50, 4(.I.),T81, 4(.T.),T102, 4(.S.),/),HITS0078
C 3(T9,25(.H.),T50, 4(.I.),T81, 4(.T.),T102,25(.S.),/),HITS0079
D11(T9, 4(.H.),T30, 4(.H.),T50, 4(.I.),T81, 4(.T.),T123, 4(.S.),/),HITS0080
E 3(T9, 4(.H.),T30, 4(.H.),T40,25(.I.),T81, 4(.T.),T102,25(.S.),/),HITS0081
F1H0,T15, HYPERVELOCITY,T48, INFLIGHT,T78, TRAJECTORY,T110, SCAT,HITS0082
GTER,/////, EEEEE,/,
H1H0, T55,CCCCC 0000 DDDD EEEEE,/,
I1H, T55,C 0 0 D D E ,/,
J1H, T55,C 0 0 D D EEE ,/,
K1H, T55,C 0 0 D D E ,/,
L1H, T55,CCCCC 0000 DDDD EEEEE,/,
M,///,T15, PREPARED BY,T85, PREPARED FOR,/,
N21, AVCO SYSTEMS DIVISION,T91, RESEARCH DIRECTORATE,/,
O21, 201 LOWELL STREET,T91, GENERAL THOMAS J. RODMAN LABORATORY,
P,T21, WILMINGTON, MASSACHUSETTS 01887,T91, ROCK ISLAND ARSENAL,
Q,T91, ROCK ISLAND, ILLINOIS 61201,/,/,/,
RT53, EFFECTIVE DATE - JANUARY 1976,
DO 6 I=1,10
6 IO(1,2) = 0
C INOM USED IN S2987.
INOM = 1
C READ THE FIRST CARD AND PLACE THE PROPER DE SUBSCRIPTS INTO THE 1A
C ARRAY.
C CALL INITIL

```

D.1 MAIN Program (continued)

```

      IF ( MCOPT.EQ. 0 ) GO TO 9
      IF ( IOPRNT.LI.3 ) IOPRNT=3
      IF ( MCALC.EQ. 3 ) IOPRNT = 4
      9 CONTINUE
      C READ THE OE NAME, TYPE, NOMINAL VALUE, TOLERANCE, VARIANCE AND TABLE
      C LENGTH AND PLACE INTO PROPER ARRAYS.
      CALL INCARD
      KCSAVE = KC + 1
      C MOVE EITHER THE TYPE 1 OR TYPES 2, 3, AND 4 UP TO THE START OF THE
      C IA ARRAY.
      CALL MOVEUP
      C PRINT NAMES OF VARIABLES
      C
      520 FORMAT(1X)
      IF (K234.EQ.0) GO TO 505
      WRITE(IOUT,501)
      FORMAT(1H1.39X,*** STOCHASTIC INDEPENDENT VARIABLES ***.//)
      501 DO 500 IK = 1,K234
      IOES = IA(IK,1)
      WRITE(IOUT,520)
      CALL FILLIN(IOES,IOUT)
      500 CONTINUE
      505 IF (KT1.EQ.0) GO TO 506
      WRITE(IOUT,507)
      FORMAT(1H1.39X,*** RANGE-CHECK INDEPENDENT VARIABLES ***.//)
      507 DO 508 IK = 1,KT1
      IOES = IA(IK,1)
      WRITE(IOUT,520)
      CALL FILLIN(IOES,IOUT)
      508 CONTINUE
      506 WRITE(IOUT,502)
      FORMAT(1H1.47X,*** DEPENDENT VARIABLES ***.//)
      502 DO 503 IK = 1,KD
      IOES = IO(IK,1)
      WRITE(IOUT,520)
      CALL FILLIN(IOES,IOUT)
      503 WRITE(IOUT,509)
      *** VARIABLES RESET BY INPUT SEQUENCE ***.//)
      509 FORMAT(1H1.39X,***
      DO 510 IK = 1,200
      IF (IA(IK,2).NE.-1) GO TO 510
      IOES = IA(IK,1)
      WRITE(IOUT,520)
      CALL FILLIN(IOES,IOUT)
      510 CONTINUE
      C CHECK COLUMN 2 OF IA FOR MISSING DATA.
      CALL CHCKIA
      C ARE WE RUNNING THE 3-DIM. MATRIX OR NON-TOL-VAR OR JUST NOMINAL VALUES
      IF ((KT1.NE.0).OR.(K234.NE.0)) CALL EXTRA
      IF ( KT1 .EQ. 0 ) GO TO 10

```

HITS0101
 HITS0102
 HITS0103
 HITS0104
 HITS0105
 HITS0106
 HITS0107
 HITS0108
 HITS0109
 HITS0110
 HITS0111
 HITS0112
 HITS0113
 HITS0114
 HITS0115
 HITS0116
 HITS0117
 HITS0118
 HITS0119
 HITS0120
 HITS0121
 HITS0122
 HITS0123
 HITS0124
 HITS0125
 HITS0126
 HITS0127
 HITS0128
 HITS0129
 HITS0130
 HITS0131
 HITS0132
 HITS0133
 HITS0134
 HITS0135
 HITS0136
 HITS0137
 HITS0138
 HITS0139
 HITS0140
 HITS0141
 HITS0142
 HITS0143
 HITS0144
 HITS0145
 HITS0146
 HITS0147
 HITS0148
 HITS0149
 HITS0150

D.1 MAIN Program (concluded)

```

INOM = 0
CALL DOABC
GO TO 30
10 IF ( K234 .EQ. 0 ) GO TO 20
   ISAVIS = ISPRNT
   ISPRNT = 0
   CALL D0234
   ISPRNT = ISAVIS
   GO TO 30
20 CALL PICK1
30 CONTINUE
   IF ( MCOPT .EQ. 0 ) GO TO 8
   KC = KCSAVE
   CALL MCRL
   IF ( IOPL0T .EQ. 0 ) GO TO 8
   CALL HYSPLT
   GO TO 8
END

```

```

HITS0151
HITS0152
HITS0153
HITS0154
HITS0155
HITS0156
HITS0157
HITS0158
HITS0159
HITS0160
HITS0161
HITS0162
HITS0163
HITS0164
HITS0165
HITS0166
HITS0167
HITS0168

```


D.3 Function A2

```

FUNCTION A2 ( X )
  IMPLICIT REAL * 8 ( A-H , O -Z )
  COMMON/OE/DEC(600)
  EQUIVALENCE
    A(PHIWIN ,OE( 103)), (V0 ,OE( 101)), (VWIND ,OE( 102)),
    B(V1 ,OE( 106)), (RHO ,OE( 104)), (CD1 ,OE( 105)),
    C(XN ,OE( 109)), (CD2 ,OE( 107)), (V2 ,OE( 108)),
    D(CMO ,OE( 112)), (CNA ,OE( 110)), (SM ,OE( 111)),
    E(PNITRI ,OE( 115)), (CMPO ,OE( 113)), (TRIM ,OE( 114)),
    F(RATE ,OE( 118)), (ANGO ,OE( 116)), (PHIANG ,OE( 117)),
    G(PHIGAM ,OE( 121)), (PHIRAT ,OE( 119)), (GAMO ,OE( 120)),
    H(D ,OE( 124)), (A ,OE( 122)), (ELL ,OE( 123)),
    I(AIY ,OE( 127)), (W ,OE( 125)), (AIX ,OE( 126)),
    J(CAA ,OE( 130)), (P ,OE( 128)), (ALPCON ,OE( 129)),
    EQUIVALENCE
    (ALPHA ,OE( 400)), (ATOT ,OE( 401)),
    (ALPTRM ,OE( 402)), (ALPHD0 ,OE( 403)), (B ,OE( 404)),
    (BETTRM ,OE( 405)), (BETA0 ,OE( 406)), (BETADO ,OE( 407)),
    C(BETA ,OE( 408)), (ALPMAX ,OE( 409)), (ALPMIN ,OE( 410)),
    D(CAYD ,OE( 411)), (CD8 ,OE( 412)), (CMA ,OE( 413)),
    E(CMHTD ,OE( 414)), (CMHTA ,OE( 415)), (CAY1 ,OE( 416)),
    F(CAY2 ,OE( 417)), (CDA0B ,OE( 418)), (CDAB ,OE( 419)),
    G(DELTX ,OE( 423)), (DELY ,OE( 421)), (DELT ,OE( 422)),
    H(DYDTC ,OE( 426)), (DYDX0 ,OE( 424)), (DZDX0 ,OE( 425)),
    I(DYDTC ,OE( 429)), (DZDTC ,OE( 427)), (DELTW ,OE( 428)),
    J(DELLEAM ,OE( 432)), (EYEP ,OE( 430)), (EMP ,OE( 431)),
    K(ELT ,OE( 435)), (EDLT ,OE( 433)), (F ,OE( 434)),
    L(HPHI1 ,OE( 438)), (PSID0 ,OE( 436)), (PHI0 ,OE( 437)),
    M(R1 ,OE( 441)), (R2 ,OE( 439)), (PSI0 ,OE( 440)),
    N(R4 ,OE( 444)), (R3 ,OE( 442)), (OE( 443)),
    P(TC0 ,OE( 447)), (RTRIM ,OE( 445)), (R3 ,OE( 446)),
    Q(TP ,OE( 450)), (TS ,OE( 448)), (TC ,OE( 449)),
    R(TOL ,OE( 453)), (THETA0 ,OE( 451)), (TGT0 ,OE( 452)),
    S(VYA ,OE( 456)), (VC ,OE( 454)), (VCO ,OE( 455)),
    EQUIVALENCE
    (W1 ,OE( 461)), (W2 ,OE( 459)), (WX ,OE( 458)),
    B(XLAM0 ,OE( 464)), (WZ ,OE( 462)), (W0 ,OE( 460)),
    C(XNU1 ,OE( 467)), (XLAM1 ,OE( 465)), (WL2 ,OE( 463)),
    D(XMU ,OE( 470)), (XNU2 ,OE( 468)), (XLAM2 ,OE( 466)),
    E(XJA ,OE( 473)), (XJAY ,OE( 469)), (XG ,OE( 472)),
    F(ALPH ,OE( 476)), (YA ,OE( 471)), (XJAZ ,OE( 474)),
    XNNP = XNU1 -XNU2 + PHI0
    A2 = CAY1 * V0 * (CAY1 /2.00 /XLAM1 * ( FEXP( 2.00 * XLAM1
    1 * X / V0 ) -1.00 ) + CAY2/ DSQRT( XLAM0 ** 2 + W0 * W0 ) *
    2( DEXP( 2.00 * XLAM0/V0 * X ) * DCOS( 2.00 * W0 * X /V0 + XNNP )
    3 - DCOS( XNNP ) ) ) + CAY2 * CAY2 * V0 / XLAM2 * ( FEXP( 2.00
    4 * XLAM2 /V0 * X ) -1.00 )
    5 /2.00
  RETURN
END

```

HITS0193
 HITS0194
 HITS0195
 HITS0196
 HITS0197
 HITS0198
 HITS0199
 HITS0200
 HITS0201
 HITS0202
 HITS0203
 HITS0204
 HITS0205
 HITS0206
 HITS0207
 HITS0208
 HITS0209
 HITS0210
 HITS0211
 HITS0212
 HITS0213
 HITS0214
 HITS0215
 HITS0216
 HITS0217
 HITS0218
 HITS0219
 HITS0220
 HITS0221
 HITS0222
 HITS0223
 HITS0224
 HITS0225
 HITS0226
 HITS0227
 HITS0228
 HITS0229
 HITS0230
 HITS0231
 HITS0232
 HITS0233
 HITS0234
 HITS0235
 HITS0236
 HITS0237
 HITS0238
 HITS0239
 HITS0240
 HITS0241
 HITS0242

[illegible]

Block Data	Implicit Real *8 (A-H,Q-Z)	Common/Names/Head(180)	Dimension Head(132).HEAD2(48)	Equivalence {HEAD1(1), HEAD(1), HEAD2(1), HEAD(133)}	Data Head1	Data Head2	Data Tol	W0	XNU1	Y0	ZN	XT	ZR	END
A	VI	CM2	ATMS	VO	W	WZ								
B	CM2	ATMS	8TRMS	V2	WZ	WZ								
C	GAM0	AZ0	AZ0	A	WZ	WZ								
D	AIY	P	ALPCON	CAA	WZ	WZ								
E	XBART	YBART	ZBART	YBART	WZ	WZ								
F	IFCRW	IFCRC	IFC	IDUMP	WZ	WZ								
G	XBART	YBART	ZBART	YBART	WZ	WZ								
H	DZT	DYX	DZX	DZT	WZ	WZ								
I	DVYX	DVZX	DI	DVYX	WZ	WZ								
J	DXTI	DYTI	DZTI	DXTI	WZ	WZ								
K	DZXI	DYXE	DZXE	DZXI	WZ	WZ								
L	DXDZTI	DYDZTI	DYDZTI	DXDZTI	WZ	WZ								
M	DYDZXE	ALPHA	ATOT	ALPHA	WZ	WZ								
N	BETA0	BETA	BETA	ALPHA	WZ	WZ								
O	CMAT	CMATHTD	CMATHTD	CMATHTD	WZ	WZ								
P	DELTT	DELTV	DELTV	DELTV	WZ	WZ								
Q	DELW	DELLW	DELLW	DELLW	WZ	WZ								
R	F	H	PSIDO	PHIO	WZ	WZ								
S	RI	R2	R3	R4	WZ	WZ								
T	DATA HEAD2	TS	TS	TG0	WZ	WZ								
A	VC	VC0	VC0	VYA	WZ	WZ								
B	W0	W1	W2	W3	WZ	WZ								
C	XNU1	XNU2	XNU3	XNU4	WZ	WZ								
D	YA	ZA	ALPH0	ALPH1	WZ	WZ								
E	TN	XTFC	YTFC	ZTFC	WZ	WZ								
F	XT	YT	ZT	VXT	WZ	WZ								
G	ZR	VXR	VYR	VZR	WZ	WZ								

D.6 Subroutine CHCKIA

```

SUBROUTINE CHCKIA
  IMPLICIT REAL*8 (A-H,O-Z)
  COMMON /CARDCM/ R(200), C(5000), IA(200,2), ID(10,2), INC,
1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K214, LABC(3)
 8000 WRITE(IOUT,8000) CH*** VARIABLES ASSIGNED PRESET VALUES ***.//)
  JKOUT = IOUT
  C IF COLUMN 2 OF IA HAS A ZERO, WE MUST FILL IN SOME DATA.
  DO 10 I=1,KA
    IAT = IA(I,1)
    IF ( IA(I,2) .EQ. 0 ) CALL FILLIN(IAT,JKOUT)
  10 CONTINUE
 8001 WRITE(IOUT,8001)
  FORMAT(1H1)
  RETURN
END

```

HITS0327
 HITS0328
 HITS0329
 HITS0330
 HITS0331
 HITS0332
 HITS0333
 HITS0334
 HITS0335
 HITS0336
 HITS0337
 HITS0338
 HITS0339
 HITS0340
 HITS0341
 HITS0342

D.7 Subroutine CHCKIN

```

SUBROUTINE CHCKIN
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/CARDCM/ 8(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT,
IIV,KARSI(3),KA,KB,KC,KD,KT1,K234,LABC(3)
COMMON/IMCH/IO1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),
IIR1(1),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NRI,NR2
DIMENSION IB(2,200)
EQUIVALENCE (IB(1,1),B(1))
THIS SUBROUTINE CHECKS THE INPUTS AND
..TYPE 8 COUNTS THE VARIABLES
..TYPE 2 FORCES THE TOLERANCE TO BE A 3 SIGMA MINIMUM
..TYPE 4 DEFINES TOLERANCE=.5(XU-XL), MEAN=.5(XU+XL) AND PUTS
THE MAXIMUM VALUE OF THE TAB. DIST. FUNCT. IN THE VARINB=0
DO 2 I=1,KD
IF (ID(I,2))2,2,1
IN8=IN8+1
CONTINUE
DO 9 I=1,K234
IADD=IA(I,2)
ITYP=IB(I,IADD)
IF (ITYP-3)3,9,5
FOR ITP=2 IT IS GAUSSIAN IT,IV=B ADDRESSES OF TOL AND VAR
IT=IADD+2
IV=IT+1
TOL=3.*DSQRT(B(IV))
IF (TOL-B(IT))9,9,4
B(IT)=TOL
GO TO 9
TYPE 4 DISTRIBUTED VARIABLE ENTRY
VAR=.
IT=IADD+4
ILT=IB(I,IT)
IT=IT+1
IS=IB(I,IT)
IV=ILT+IS-1
IT IS THE ADDRESS IN B OF THE 1ST C ADDRESS FOR THE TAB. DIST. FC
IT IS THE NUMBER OF ENTRIES IN C
IV IS THE LAST ADDRESS IN C OF THE DIST. FUNCT.
DO 6 J=IS,IV
IF (VAR-C(J))7,6,6
VAR=C(J)
CONTINUE
VAR IS THE MAX. VALUE STORE IN VARIANCE POSITION OF 8 ENTRIES
CALCULATE 1ST AND LAST ADDRESSES OF VARIABLE TABLE IN C
IT=IV+1
IV=IT+ILT-1
IADD=IADD+1
B(IADD)=.5*(C(IT)+C(IV))
IADD=IADD+1

```

HITS0343
HITS0344
HITS0345
HITS0346
HITS0347
HITS0348
HITS0349
HITS0350
HITS0351
HITS0352
HITS0353
HITS0354
HITS0355
HITS0356
HITS0357
HITS0358
HITS0359
HITS0360
HITS0361
HITS0362
HITS0363
HITS0364
HITS0365
HITS0366
HITS0367
HITS0368
HITS0369
HITS0370
HITS0371
HITS0372
HITS0373
HITS0374
HITS0375
HITS0376
HITS0377
HITS0378
HITS0379
HITS0380
HITS0381
HITS0382
HITS0383
HITS0384
HITS0385
HITS0386
HITS0387
HITS0388
HITS0389
HITS0390
HITS0391
HITS0392

D.7 Subroutine CHCKIN (concluded)

```
B(IADD)=.5*(C(IV)-C(IT))  
IADD=IADD+1  
B(IADD)=VAR  
CONTINUE  
RETURN  
END
```

9

```
HITS0393  
HITS0394  
HITS0395  
HITS0396  
HITS0397  
HITS0398
```

D.8 Subroutine CNVRT

```

SUBROUTINE CNVRT(RN)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION R(11)
DATA R/0.0,0.28,0.36,0.41,0.46,0.5,0.54,0.59,0.64,0.72,1.0/
THIS PROGRAM CONVERTS RECT. DIST. NOS. TO GAUSSIAN DIST. NOS.
AJ=10.*RN
J=AJ
BJ=J
RN=(AJ-BJ)*(R(J+2)-R(J+1))+R(J+1)
RETURN
END

```

C

HITS0399
HITS0400
HITS0401
HITS0402
HITS0403
HITS0404
HITS0405
HITS0406
HITS0407
HITS0408
HITS0409

[illegible]

D.9 Subroutine CROSS (continued)

```

1000      W2TV = W2T      - XNU2
1001      FORMAT( ' VCONZ TP ALPHA ZCONA BETA VYA ATOT VZA' )
1002      1VY VCONZ TP ALPHA ZCONA BETA VYA ATOT VZA
2/1X, IPI0E12.4 // )
1      ALPHA = ELT1 * CAY1 * DSIN( W1TV ) - ELT2 * CAY2
1      BETA = ELT1 * CAY1 * DCOS( W1TV ) + ELT2 * CAY2
1      TP2 = TP * TP
1      A2 = 0.00
1      B2 = 0.00
1      IF( DABS( ALPHA ) .GT. 1.0-38 )
1      A2 = ALPHA * ALPHA
1      IF( DABS( BETA ) .GT. 1.0-38 ) B2 = BETA * BETA
1      ATOT = DSORT( A2+B2 )
1      P2 = P * P
1      PTP = P * TP
1      IF( P .EQ. 0. ) GO TO 1
1      SINPT = DSIN( PTP )
1      COSPT = DCOS( PTP )
1      VCONZ = (ALPTRM * SINPT - BETTRM * (COSPT - 1.00)) / P
1      VCONZ = -(BETTRM * SINPT + ALPTRM * (COSPT - 1.00)) / P
1      VCONA = (ALPTRM * (1.00 - COSPT) + BETTRM * (PTP - SINPT)) / P2
1      ZCONA = (-BETTRM * (1.00 - COSPT) + ALPTRM * (PTP - SINPT)) / P2
1      GO TO 2
1      CONTINUE
1      VCONY = TP * ALPTRM
1      VCONZ = -BETTRM * TP
1      VCONA = TP2 * ALPTRM / 2.00
1      ZCONA = -TP2 * BETTRM / 2.00
2      CONTINUE
2      HV = H
2      HVV = HV * V
2      ELWP1 = XLAM1 ** 2 + (W1 + P) ** 2
2      ELWP2 = XLAM2 ** 2 + (W2 - P) ** 2
2      CAY1LW = CAY1 / DSORT( ELWP1 )
2      CAY2LW = CAY2 / DSORT( ELWP2 )
2      XNUPH1 = XNU1 + PHI1
2      XNUPH2 = -XNU2 + PHI2
2      WPTVP1 = (W1 + P) * TP + XNUPH1
2      WPTVP2 = (W2 - P) * TP + XNUPH2
2      COSNP1 = DCOS( XNUPH1 )
2      COSNP2 = DCOS( XNUPH2 )
2      SINNP1 = DSIN( XNUPH1 )
2      SINNP2 = DSIN( XNUPH2 )
2      VYA = HVV * ( CAY1LW * (ELT1 * DSIN( WPTVP1 ) - SINNP1
2      - CAY2LW * (ELT2 * DSIN( WPTVP2 ) - SINNP2
2      + VCONY + DYDTOT / H )
2      VZA = HVV * (-CAY1LW * (ELT1 * DCOS( WPTVP1 ) - COSNP1
2      - CAY2LW * (ELT2 * DCOS( WPTVP2 ) - COSNP2
2      )

```

D.9 Subroutine CROSS (concluded)

```

2 + VCONZ + DZDTG/ H )
WCON1 = WPTVP1 + PHI2
WCON2 = WPTVP2 + PHI2
IF( IDUMP .EQ. 1 ) WRITE(6,1000) TP , ALPHA , BETA , ATOT
1.VCONY , VCONZ , YCONA , ZCONA , VYA , VZA
YA = H * ( CAY1/ELWP1 * ELT1 * DSIN(WCON1 ) - CAY2/ELWP2 * ELT2
1 * DSIN(WCON2) - (CAY1LW * DSIN( XNUPH1 ) - CAY2LW * DSIN(XNUPH2) - DYDT
20 /H ) * TP - CAY1/ELWP1 * DSIN(XNUPH1+PHI1) + CAY2/ELWP2 * DSIN
3(XNUPH2 +PHI2) + YCONA )
ZA = H * ( -CAY1/ELWP1 * ELT1 * DCOS(WCON1 ) - CAY2/ELWP2 * ELT2
1 * DCOS(WCON2) + (CAY1LW * DCOS(XNUPH1 ) + CAY2LW * DCOS(XNUPH2) + DZDT
20 /H ) * TP + CAY1/ELWP1 * DCOS(XNUPH1+PHI1) + CAY2/ELWP2 * DCOS
3(XNUPH2 +PHI2) + ZCONA )
EOLT = FEXP( XLAMO * TP)
EDLT = DEXP( DELLAM* TP)
ALPMAZ = EOLT * ( CAY1 * EDLT + CAY2 / EDLT )
ALPMAZ = EOLT * ( CAY1 * EDLT - CAY2 / EDLT )
XJAY = -HV * ( CAY1LW * SINNP1 - CAY2LW * SINNP2 ) * 1.D3
XJAZ = 4V * ( CAY1LW * COSNP1 + CAY2LW * COSNP2 ) * 1.D3
XJA = DSQRT( XJAY ** 2 + XJAZ ** 2 )
IF( IDUMP , EQ. 1 ) WRITE(6,1001) YA , ZA , EOLT , EDLT
1001 FORMAT( , ALPMAZ , ALPMAZ , XJAY , XJAZ , XJA , EDLT , EDLT
24 // )
RETURN
END

```

HITS0510
 HITS0511
 HITS0512
 HITS0513
 HITS0514
 HITS0515
 HITS0516
 HITS0517
 HITS0518
 HITS0519
 HITS0520
 HITS0521
 HITS0522
 HITS0523
 HITS0524
 HITS0525
 HITS0526
 HITS0527
 HITS0528
 HITS0529
 HITS0530
 HITS0531
 HITS0532
 HITS0533
 HITS0534
 HITS0535
 HITS0536

D.10 Subroutine CROSSE

```

SUBROUTINE CROSSE( X, V )
  IMPLICIT REAL *8 (A-H, J-Z)
  COMMON/OE/OE(600)
  EQUIVALENCE
    A(PHIWIN),OE( 3)),(V0),OE( 1)),(VMIND),OE( 2)),
    B(V1),OE( 6)),(RHO),OE( 4)),(CD1),OE( 5)),
    C(XN),OE( 9)),(CD2),OE( 7)),(V2),OE( 8)),
    D(CMQ),OE(12)),(CNA),OE(10)),(SM),OE(11)),
    E(PNITPI),OE(15)),(CMXA),OE(13)),(TRIM),OE(14)),
    F(RATE),OE(18)),(ANGO),OE(16)),(PHIANG),OE(17)),
    G(PHIGAM),OE(21)),(PHIRAT),OE(19)),(GAMO),OE(20)),
    H(D),OE(24)),(A),OE(22)),(ELL),OE(23)),
    I(ATY),OE(27)),(W),OE(25)),(AIX),OE(26)),
    J(CAA),OE(30)),(P),OE(28)),(ALPCON),OE(29))
  EQUIVALENCE
    A(ALPTRM),OE( 42)),(IDUMP),OE(204)),
    B(BETTRM),OE( 45)),(ALPHA),OE(400)),
    C(BETA),OE( 48)),(ALPHD0),OE(403)),
    D(CAYD),OE( 41)),(BETAD0),OE(404)),
    E(CMTHTD),OE( 44)),(ALPMAX),OE(406)),
    F(CAY2),OE( 47)),(CD8),OE(412)),
    G(DELTT),OE( 42)),(CMTHTA),OE(415)),
    H(DELX),OE( 43)),(COA08),OE(418)),
    I(DYDT),OE( 46)),(DELY),OE(421)),
    J(DELLAM),OE( 42)),(DYDX),OE(424)),
    K(FCLT),OE( 43)),(EYEP),OE(427)),
    L(H),OE( 45)),(EDLT),OE(433)),
    M(PHI1),OE( 48)),(PSID),OE(436)),
    N(R1),OE( 41)),(PHI2),OE(439)),
    O(R4),OE( 44)),(R2),OE(442)),
    P(TCO),OE( 47)),(RTRIM),OE(445)),
    Q(TP),OE( 50)),(ITS),OE(448)),
    R(TOL),OE( 53)),(THETA0),OE(451)),
    S(VYA),OE( 56)),(VC),OE(454)),
    EQUIVALENCE
      A(W1),OE( 61)),(WZ),OE(457)),
      B(XLAM),OE( 64)),(W2),OE(459)),
      C(XNU1),OE( 67)),(XLAM1),OE(462)),
      D(XMU),OE( 70)),(XNU2),OE(465)),
      E(XJA),OE( 73)),(XJAY),OE(468)),
      F(ALPH),OE( 76)),(YA),OE(471)),
      TP = X/V0,
      TL1 = XLAM1 * TP,
      TL2 = XLAM2 * TP,
      ELT1 = FEXP( TL1 ),
      ELT2 = FEXP( TL2 ),
      W1T = W1 * TP,
      W1TV = W1T + XNU1,
      W2T = W2 * TP
  
```

HITS0537
 HITS0538
 HITS0539
 HITS0540
 HITS0541
 HITS0542
 HITS0543
 HITS0544
 HITS0545
 HITS0546
 HITS0547
 HITS0548
 HITS0549
 HITS0550
 HITS0551
 HITS0552
 HITS0553
 HITS0554
 HITS0555
 HITS0556
 HITS0557
 HITS0558
 HITS0559
 HITS0560
 HITS0561
 HITS0562
 HITS0563
 HITS0564
 HITS0565
 HITS0566
 HITS0567
 HITS0568
 HITS0569
 HITS0570
 HITS0571
 HITS0572
 HITS0573
 HITS0574
 HITS0575
 HITS0576
 HITS0577
 HITS0578
 HITS0579
 HITS0580
 HITS0581
 HITS0582
 HITS0583
 HITS0584
 HITS0585
 HITS0586

D.10 Subroutine CROSSF (concluded)

```

2 + VCONZ + DZDT0/ H )
  WCON1 = WPTVP1 + PHI1
  WCON2 = WPTVP2 + PHI2
  IF( IDUMP .EQ. 1 ) WRITE(6,1000) TP , ALPHA , BETA , ATOT
1. VCONY , VCONZ , YCONA , ZCONA , VYA , VZA
  YA = H * ( CAY1/ELWP1 * ELT1 * DSIN(WCON1) - CAY2/ELWP2 * ELT2
1 * DSIN(WCON2) - (CAY1LW * SIN( XNUPH1) - CAY2LW * DSIN(XNUPH2) - DYDT
20 /H ) * TP - CAY1/ELWP1 * DSIN(XNUPH1+PHI1) + CAY2/ELWP2 * DSIN
3(XNUPH2 +PHI2) + YCONA )
  ZA = H * ( -CAY1/ELWP1 * ELT1 * DCOS(WCON1) - CAY2/ELWP2 * ELT2
1 * DCOS(WCON2) / ( CAY1LW * DCOS(XNUPH1) + CAY2LW * DCOS(XNUPH2) + DZDT
20 /H ) * TP + CAY1/ELWP1 * DCOS(XNUPH1+PHI1) + CAY2/ELWP2 * DCOS
3(XNUPH2 +PHI2) + ZCONA )
  EOLT = FEXP( XLAMO * TP)
  EOLT = DEXP( OELLAM* TP)
  ALPMAX = EOLT * ( CAY1 * EOLT + CAY2 / EOLT )
  ALPMIN = EOLT * ( CAY1 * EOLT - CAY2 / EOLT )
  XJAY = -HV * ( CAY1LW * SINNP1 - CAY2LW * SINNP2 ) * 1.03
  XJAZ = HV * ( CAY1LW * COSNP1 + CAY2LW * COSNP2 ) * 1.03
  XJA = DSQRT( XJAY ** 2 + XJAZ ** 2 )
  IF( IDUMP .EQ. 1 ) WRITE(6,1001) YA , ZA , EOLT , EOLT
1001 FORMAT( ' YA XJAY XJAZ XJA EDLT XJA EDLT ALPHITS0637
  IMAX ALPMIN XJAY XJAZ EDLT XJA EDLT ALPHITS0638
24 // ) RETURN
END
ALPHITS0639
ALPHITS0640
ALPHITS0641
ALPHITS0642
ALPHITS0643
ALPHITS0644
ALPHITS0645
ALPHITS0646
ALPHITS0647
ALPHITS0648
ALPHITS0649
ALPHITS0650
ALPHITS0651
ALPHITS0652
ALPHITS0653
ALPHITS0654
ALPHITS0655
ALPHITS0656
ALPHITS0657
ALPHITS0658
ALPHITS0659
ALPHITS0660
ALPHITS0661
ALPHITS0662
ALPHITS0663

```

D.11 Subroutine CRVFT

```

SUBROUTINE CRVFT(KOPT)
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/CARDCM/ 8(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT,
IIV,KAVSI(3),KA,KB,KC,KD,KT1,K234,LABC(3)
COMMON/OEC/OE(600)
DIMENSION DELX(20),FUN(10)
THIS ROUTINE USES THE T.S. EXPANSIONS OF YP-58.YP3
USES 1ST,2ND UNMIXED OR FULL 2ND ORDER FOR KOPT =1,2,3 RESP.
CREATE DELX SAMPLE TABLE
DO 1 I=1,K234
JB=IA(I,2)+1
IOE=IA(I,1)
DELX(I) = OE(IOE) - B(JB)
INITIALIZE EXPANSIONS SUMS TO NOMINAL VALUES
DO 2 I=1,KD
FUN(I)=C(KC+I-1)
START LOOP ON INDEPENDENT UNMIXED DERIVATIVE SUMS
DO 4 I=1,K234
I1 = KC + KD * (2*I-1) - 1
I2=I1+KD
GO TO(12,11,11),KOPT
DO 13 J=1,KD
FUN(J)=FUN(J)+DELX(I)*C(I1+J)
GO TO 4
DO 3 J=1,KD
FUN(J)=FUN(J)+DELX(I)*C(I1+J)+.5*DELX(I)*C(I2+J))
CONTINUE
END OF LOOP --CHECK FOR INCLUSION OF MIXED DERIVATIVES
IF(KOPT-2)9,9,5
I1=I2+3*KD
I2=K234-1
DO 8 I=1,I2
JB=I+1
DO 7 J=JB,K234
DO 6 L=1,KD
FUN(L)=FUN(L)+DELX(I)*DELX(J)*C(I1+L)
I1=I1+KD
CONTINUE
STORE RESULTS IN DE ARRAY
DO 10 I=1,KD
IOE=ID(I,1)
OE(IOE)=FUN(I)
RETURN
END

```

HITS0664
HITS0665
HITS0666
HITS0667
HITS0668
HITS0669
HITS0670
HITS0671
HITS0672
HITS0673
HITS0674
HITS0675
HITS0676
HITS0677
HITS0678
HITS0679
HITS0680
HITS0681
HITS0682
HITS0683
HITS0684
HITS0685
HITS0686
HITS0687
HITS0688
HITS0689
HITS0690
HITS0691
HITS0692
HITS0693
HITS0694
HITS0695
HITS0696
HITS0697
HITS0698
HITS0699
HITS0700
HITS0701
HITS0702
HITS0703
HITS0704
HITS0705
HITS0706
HITS0707

D.12 Subroutine DOABC

```

SUBROUTINE DOABC
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL*1 BL,*,*
LOGICAL*1 OTT(10) /10H1234567890/
LOGICAL*1 FMT1(46)/46H(' IND. VAR.', (A1,' OE('I4,' )=,1PE12.4),./
A//
LOGICAL*1 FMT2(29) /29H(5X, (A1,' OE('I4,' )=,1PE12.4),./
LOGICAL*1 FMT3(18) /18H(5X,1P E12.4,///)
COMMON /CARDCM/ B(200), C(5000), IA(200,2), IO(10,2), INC,
1 IOPRNT, IOUT, IY, KARSI(3), KA, KB, KC, KD, KTL, K234, LABC(3)
COMMON/OEC/OE(600)
COMMON/CISPRN/ ISPRNT
DIMENSION IOES(3), JC(3), IOE(2, 600), IB(2,200),
1 ABCOE(3), OUTOE(10)
EQUIVALENCE (IOES(1), IAOE), (IOES(2), IBOE), (IOES(3), ICOE),
1 (JC(1), JCA), (JC(2), JCB), (JC(3), JCC), (OE(1), IOE(1,1)),
2 (B(1), IB(1,1))
2 IF (ISPRNT.EQ.0) WRITE(IOUT,500)
500 FORMAT(1H1,39X,*** RANGE CHECK INDEPENDENT VARIABLE COMBINATIONS
A***,//)
501 FORMAT(1H1,39X,*** RANGE CHECK INDEPENDENT VARIABLE COMBINATION
A***,//)
C SET UP THE FORMATS.
FMT1(15) = OTT(KT1)
FMT2(8) = OTT(KD)
FMT3(8) = OTT(KO)
IF ( KD .LT. 10 ) GO TO 2
FMT2(5) = OTT(1)
FMT3(7) = OTT(1)
2 CONTINUE
C IOES(1) IS THE OE SUBSCRIPT FOR THE 1,TH TYPE 1 ARRAY.
C JC(1) IS ONE LESS THAN THE ADDRESS IN C FOR THE 1,TH TYPE 1 ARRAY.
C THE OE SUBSCRIPTS FOR THE TYPE 1 ARRAYS HAVE BEEN MOVED UP TO THE
C START OF THE IA ARRAY.
C GET THE OE SUBSCRIPTS AND THEIR C ADDRESS.
DO 5 I=1,KT1
IOES(I) = IA(I,1)
JB = IA(I,2)
5 JC(I) = IB(1,JB+2) - 1
IAR = 0
10 IAR = IAR + 1
IF ( IAOE .GT. 0 ) GO TO 12
L = -IAOE
IOE(1,L) = C(JCA+IAR)
GO TO 15
12 OE(IAOE) = C(JCA+IAR)
15 ABCOE(1) = C(JCA+IAR)
C IF THERE IS ONLY ONE TYPE 1 ARRAY, IBR=1 TO GET PAST 3 IF'S AT END.
IAR = 1
IF ( KTL .LT. 2 ) GO TO 40

```

HITS0708
HITS0709
HITS0710
HITS0711
HITS0712
HITS0713
HITS0714
HITS0715
HITS0716
HITS0717
HITS0718
HITS0719
HITS0720
HITS0721
HITS0722
HITS0723
HITS0724
HITS0725
HITS0726
HITS0727
HITS0728
HITS0729
HITS0730
HITS0731
HITS0732
HITS0733
HITS0734
HITS0735
HITS0736
HITS0737
HITS0738
HITS0739
HITS0740
HITS0741
HITS0742
HITS0743
HITS0744
HITS0745
HITS0746
HITS0747
HITS0748
HITS0749
HITS0750
HITS0751
HITS0752
HITS0753
HITS0754
HITS0755
HITS0756
HITS0757

D.12 Subroutine DOABC (concluded)

```

      GO TO 21
20 IAR = IBR + 1
21 IF ( IAOE .GT. 0 ) GO TO 22
   L = -IAOE
   IOE(1,L) = C(JCB+IBR)
   GO TO 25
22 OE(IAOE) = C(JCB+IBR)
25 ABCOE(2) = C(JCB+IBR)
   C IN THE EVENT THERE IS NO THIRD TYPE 1 ARRAY, ICR=1 WILL GET US PAST
   C THE 3 IF,S AT THE END.
   ICR = 1
   IF ( KT1 LT 3 ) GO TO 40
   GO TO 31
30 ICR = ICR + 1
31 IF ( IOOE .GT. 0 ) GO TO 32
   L = -IOOE
   IOE(1,L) = C(JCC+ICR)
   GO TO 35
32 OE(ICOE) = C(JCC+ICR)
35 ABCOE(3) = C(JCC+ICR)
40 CALL PICK1
   IF(IISPRNT.EQ.1) WRITE(IOUT,501)
   WRITE (IOUT,FMT1) (BL, IA(I,1), ABCOE(I), I=1,KT1)
   DO 50 I=1,KD
   JOE = ID(I,1)
50 OUTOE(I) = OE(JOE) (BL, ID(I,1), I=1,KD)
   WRITE (IOUT,FMT2) (OUTOE(I), I=1,KD)
   IF ( ICR .LT. LABC(3) ) GO TO 30
   IF ( IBR .LT. LABC(2) ) GO TO 20
   IF ( IAR .LT. LABC(1) ) GO TO 10
   RETURN
END

```

HITS0758
 HITS0759
 HITS0760
 HITS0761
 HITS0762
 HITS0763
 HITS0764
 HITS0765
 HITS0766
 HITS0767
 HITS0768
 HITS0769
 HITS0770
 HITS0771
 HITS0772
 HITS0773
 HITS0774
 HITS0775
 HITS0776
 HITS0777
 HITS0778
 HITS0779
 HITS0780
 HITS0781
 HITS0782
 HITS0783
 HITS0784
 HITS0785
 HITS0786
 HITS0787
 HITS0788
 HITS0789
 HITS0790

D.13 Subroutine DO234

```

SUBROUTINE DO234
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL*1 BL,/,
LOGICAL*1 OTT(10)/10H1234567890/
LOGICAL*1 FMT2(29)/29H(5X,(A1.,OE(.,14.,.) ),.)/
LOGICAL*1 FMT3(14)/14H(5X,1P E12.4)/
DIMENSION IBB(2,200)
EQUIVALENCE (IBB(1,1),B(1))
COMMON /CARDCM/ B(200), C(5000), IA(200,2), ID(10,2), INC,
1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KY1, K234, LABC(3)
COMMON/OEC/OE(600)
EQUIVALENCE (IDUMP ,OE( 204))
DIMENSION IOES(2), LN(2)
EQUIVALENCE (IOES(1), IOE1), (IOES(2), IOE2), (LN(1), L1),
1 (LN(2), L2)
COMMON/INDEX/ JD, I234, ICRYBAR, ICSUMS, ICDQUB, KCON
IJB1 = 0
IJB2 = 0
IJB3 = 0
IJB4 = 0
DO 20 IBOB = 1, KD
IJB = ID(IBOB,1)
IF(IJB.EQ.320) IJB1 = IBOB
IF(IJB.EQ.321) IJB2 = IBOB
IF(IJB.EQ.318) IJB3 = IBOB
IF(IJB.EQ.319) IJB4 = IBOB
200 CONTINUE
IJB = IJB1 + IJB2 + IJB3 + IJB4
C SET UP THE FORMATS.
FMT2(6) = OTT(KD)
FMT3(8) = OTT(KD)
IF ( KD .LT. 10 ) GO TO 2
FMT2(5) = OTT(1)
FMT3(7) = OTT(1)
2 CONTINUE
C KCRYBAR IS ONE LESS THAN THE START OF THE YBAR OUTPUT IN THE C ARRAY.
KCRYBAR = KC
IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 201
WRITE (IOUT,9003) (BL, IA(I,1), OE(IA(I,1)), I=1,K234)
9003 FORMAT('1NOMINAL CASE, INDEPENDENT VARIABLES ...',/, (5X, 5(A1,
1 , OE(., 14., .))=, IPE12.4))
201 CONTINUE
C THE NOMINAL VALUES.
CALL PICK1
CALL STOREC
K = KC - KD
IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 801
WRITE (IOUT,9005)
9005 FORMAT('1NOMINAL CASE, DEPENDENT VARIABLES ...',/, (
1 WRITE (IOUT,FMT2) (BL, ID(I,1), I=1,KD)

```

HITS0791
HITS0792
HITS0793
HITS0794
HITS0795
HITS0796
HITS0797
HITS0798
HITS0799
HITS0800
HITS0801
HITS0802
HITS0803
HITS0804
HITS0805
HITS0806
HITS0807
HITS0808
HITS0809
HITS0810
HITS0811
HITS0812
HITS0813
HITS0814
HITS0815
HITS0816
HITS0817
HITS0818
HITS0819
HITS0820
HITS0821
HITS0822
HITS0823
HITS0824
HITS0825
HITS0826
HITS0827
HITS0828
HITS0829
HITS0830
HITS0831
HITS0832
HITS0833
HITS0834
HITS0835
HITS0836
HITS0837
HITS0838
HITS0839
HITS0840

[illegible]

[illegible]

D.13 Subroutine D0234 (continued)

```

      SYJ = SYJ + C(KRMJ) * VAR
      SYJS = SYJS + C(KRP+J)**2 * VAR
40  CONTINUE
      C(KCSMJ) = C(KCYBJ) + 0.500 * SYJ
      C(KCSMJ+KD) = SYJS
50  CONTINUE
      IF((IJB.GT.3).AND.(IDUMP.EQ.0)) GO TO 206
      C PRINT E(Y(J)) AND E(Y(J)**2).
      WRITE(IOUT,6969)
      WRITE(IOUT,9012) (ID(I,1), I=1,KD)
      9012 FORMAT('  2ND ORDER MEANS AND 1ST ORDER VARIANCES'/// 62X.
      A'DEPENDENT VARIABLES'/// 8X, 10I12)
      WRITE(IOUT,9013) (C(KCSUMS+I), I=1,KD)
      9013 FORMAT('1X,4X,5HMEANS,1X,1P10E12.4)
      WRITE(IOUT,9023) (C(KCSUMS+KD+I), I=1,KD)
      9023 FORMAT('1X,9HVARIANCES,1X,1P10E12.4)
      DO 7500 I = 1,KD
      7500 CALL STDDEV(C(KCSUMS+KD+I),I,KD,IOUT)
      206 CONTINUE
      JB0B = KCSUMS
      KM0B = KCYBAR
      IF ( IOPRNT.EQ. 3 ) GO TO 920
      C UPDATE KC DUE TO E(Y(J)) AND E(Y(J)**2).
      KC = KC + 2 * KD
      KCDOUB = KC
      C VARY THE INDEPENDENT VARIABLES TWO AT A TIME.
      IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 207
      WRITE(IOUT,9014) (ID(I,1), I=1,KD)
      9014 FORMAT('1SECOND ORDER MIXED PARTIALS'/// 8X, 10I12)
      1'DEPENDENT VARIABLES'/// (8X, 10I12)
      207 CONTINUE
      C KC DR WILL STORE THE SECOND DERIVATIVES.
      KC DR = KCDOUB - KD
      KM1 = K234 - 1
      KD3 = 3 * KD
      DO 90 I=1,KM1
      IOE = IA(I,1)
      IR = IA(I,2)
      IPI = I + 1
      DO 90 J=IPI,K234
      JOE = IA(J,1)
      JR = IA(J,2)
      BB = B(1B+2) * B(JB+2)
      C THE NEXT DOUBLE DO LOOP WILL CREATE THE FOUR COMBINATIONS OF NOMINAL
      C VALUE PLUS AND MINUS TOLERANCE.
      DO 60 I2=1,2
      DE(IOE) = B(1B+1) + (-1.000)**(3-I2) * B(1B+2) / 2.000
      DO 60 J2 = 1,2
      DE(JOE) = B(JB+1) + (-1.000)**(3-J2) * B(JB+2) / 2.000
      CALL PICK1

```

HITS0941
 HITS0942
 HITS0943
 HITS0944
 HITS0945
 HITS0946
 HITS0947
 HITS0948
 HITS0949
 HITS0950
 HITS0951
 HITS0952
 HITS0953
 HITS0954
 HITS0955
 HITS0956
 HITS0957
 HITS0958
 HITS0959
 HITS0960
 HITS0961
 HITS0962
 HITS0963
 HITS0964
 HITS0965
 HITS0966
 HITS0967
 HITS0968
 HITS0969
 HITS0970
 HITS0971
 HITS0972
 HITS0973
 HITS0974
 HITS0975
 HITS0976
 HITS0977
 HITS0978
 HITS0979
 HITS0980
 HITS0981
 HITS0982
 HITS0983
 HITS0984
 HITS0985
 HITS0986
 HITS0987
 HITS0988
 HITS0989
 HITS0990

D.13 Subroutine D0234 (continued)

```

        CALL STOREC
60  CONTINUE
    KCDR = KCDR + KD
    C THE COMBINATIONS OF PLUS AND MINUS ARE KD APART.
    DO 70 L=1,KD
        KCDRJ = KCDR + L
        C(KCDRJ) = ( C(KCDRJ) + C(KCDRJ+KD3) - C(KCDRJ+KD) - C(KCDRJ+
            1 KD2) ) / BB
70  CONTINUE
    IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 208
    WRITE(IOUT,9015) IOE, JOE, (C(KCDR+L), L=1,KD)
9015 FORMAT('0',2I5, 2X, 1P10E12.4)
208  CONTINUE
    C RESTORE THE SECOND OE OF THE PAIR TO ITS NOMINAL VALUE.
    OE(JOF) = B(JB+1)
    C BRING KC UP TO DATE.
    KC = KCDR + KD
80  CONTINUE
    C RESTORE THE FIRST OE OF THE PAIR TO ITS NOMINAL VALUE.
    OE(IOE) = B(IB+1)
90  CONTINUE
    JB08 = KCSUMS
    KMB08 = KCYBAR
    IF ( IOPRNT.EQ. 4 ) GO TO 900
50  GAUSSIAN CORRECTION FOR VARIANCES
    I234 = K234
    ICYBAR = KCYBAR
    ICSUMS = KCSUMS
    ICDOUB = KCDOUB
    JD = KD
    KCOV = KC
    IF(IOPRNT.EQ.5) GO TO 803
    IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 805
    WRITE(IOUT,6969)
    WRITE(IOUT,9050)(ID(I,1),I=1,KD)
9050  FORMAT(' 2ND ORDER MEANS AND VARIANCES ASSUMING GAUSSIAN INDEPENDENT
805  ANT VARIABLES'///62X, 'DEPENDENT VARIABLES' // 8X, 10I12)
    CONTINUE
    CALL GQUC
    IF((IJB.GT.0).AND.(IDUMP.EQ.0)) GO TO 806
    WRITE(IOUT,9013)(C(KCSUMS+I),I = 1,KD)
    WRITE(IOUT,9023)(C(KCSUMS+KD+I),I = 1,KD)
    DO 7501 I = 1,KD
7501  CALL STDDEV(C(KCSUMS+KD+I),I,KD,IOUT)
806  CONTINUE
    IF(IOPRNT.EQ.6) GO TO 900
803  CONTINUE
    KDAVE = (K234 * (K234 + 1))/2

```

HITS0991
 HITS0992
 HITS0993
 HITS0994
 HITS0995
 HITS0996
 HITS0997
 HITS0998
 HITS0999
 HITS1000
 HITS1001
 HITS1002
 HITS1003
 HITS1004
 HITS1005
 HITS1006
 HITS1007
 HITS1008
 HITS1009
 HITS1010
 HITS1011
 HITS1012
 HITS1013
 HITS1014
 HITS1015
 HITS1016
 HITS1017
 HITS1018
 HITS1019
 HITS1020
 HITS1021
 HITS1022
 HITS1023
 HITS1024
 HITS1025
 HITS1026
 HITS1027
 HITS1028
 HITS1029
 HITS1030
 HITS1031
 HITS1032
 HITS1033
 HITS1034
 HITS1035
 HITS1036
 HITS1037
 HITS1038
 HITS1039
 HITS1040

D.13 Subroutine D0234 (continued)

```

      DO 8570 IDAV = 1,KDAVE
      IDA = KCOV + IDAV
      IF (IDA.GT.5000) WRITE(6,8500) IDA
      8600 FORMAT(10X,'IDA = ',15,1X,'IN D0234 WHICH EXCEEDS THE C ARRAY')
      8500 C(IDA) = 0.000
      C THE CARD FOR THIS READ IS DIRECTLY BEHIND THE NAME-TYPE-NOM-TGL-VAR-
      C TABLE LENGTH CARDS.
      5000 WRITE(IOUT,5000)
      442X,6X,'CODE',6X,6X,'CODE',6X,2X,'CORRELATION',/
      842X,5X,'NUMBER',5X,5X,'NUMBER',5X,2X,'COEFFICIENT',/
      120 READ (INC,9001) IOE1, IOE2, RHO
      9001 FORMAT(15,15,F13.1)
      IF (IOE1.LT.0) GO TO 802
      WRITE(IOUT,5001) IOE1,IOE2,RHO
      5001 FORMAT(40X,5X,15,6X,5X,15,6X,2X,1PE11.0,/)
      C RUN THRU IA AND GET SUBSCRIPTS (OF THE IA ARRAY) WHERE IOE1 AND
      C IOE2 ARE LOCATED.
      DO 130 J=1,2
      DO 110 I=1,K234
      ISAV = I
      IF (IA(I,1).EQ. IOES(J)) GO TO 120
      110 CONTINUE
      9002 WRITE (IOUT,9002)
      9002 FORMAT(39H ILLEGAL OE SUBSCRIPT READ IN WITH RHO.)
      CALL EXIT
      120 LN(J) = ISAV
      130 CONTINUE
      C GET THE SMALLEST IA SUBSCRIPT INTO LN(1).
      C GET IF ( LN(1) .LT. LN(2) ) GO TO 140
      LS = LN(1)
      LN(1) = LN(2)
      LN(2) = LS
      C GET THE SUBSCRIPTS TO PICK UP THE VARIANCES IN THE B ARRAY.
      140 IB1 = IA(L1,2)
      IB2 = IA(L2,2)
      COV = RHO * DSORT( B(IB1+3) * B(IB2+3) )
      C STORE COVARIANCES IN THE C ARRAY
      C
      MCOV = VCOV(L1,L2)
      C(MCOV) = COV
      GO TO 130
      802 CONTINUE
      C
      C COMPUTE 2ND ORDER MEANS AND 1ST ORDER VARIANCES WITH CORRELATIONS
      CALL OCOR
      IF (IOPRNT.EQ.7) GO TO 804
      IF ((IJ3.GT.0).AND.(IDUMP.EQ.0)) GO TO 807
      WRITE(IOUT,6950)

```

HITS1041
HITS1042
HITS1043
HITS1044
HITS1045
HITS1046
HITS1047
HITS1048
HITS1049
HITS1050
HITS1051
HITS1052
HITS1053
HITS1054
HITS1055
HITS1056
HITS1057
HITS1058
HITS1059
HITS1060
HITS1061
HITS1062
HITS1063
HITS1064
HITS1065
HITS1066
HITS1067
HITS1068
HITS1069
HITS1070
HITS1071
HITS1072
HITS1073
HITS1074
HITS1075
HITS1076
HITS1077
HITS1078
HITS1079
HITS1080
HITS1081
HITS1082
HITS1083
HITS1084
HITS1085
HITS1086
HITS1087
HITS1088
HITS1089
HITS1090

[illegible]

D.13 Subroutine D0234 (continued)

```

DO 503 I808 = 1,KD
IF(ID(I808,1).EQ.306) GO TO 504
IF(ID(I808,1).EQ.307) GO TO 505
IF(ID(I808,1).EQ.308) GO TO 506
GO TO 503
504 OXT0 = C(KM808 + I808)
M808 = M808 + 1
IF(IOPRNT.LT.3) GO TO 503
OE(301) = C(J808 + I808)
BVAR1 = C(J808 + KD + I808)
GO TO 503
505 OYT0 = C(KM808 + I808)
M808 = M808 + 1
IF(IOPRNT.LT.3) GO TO 503
OE(302) = C(J808 + I808)
BVAR2 = C(J808 + KD + I808)
GO TO 503
506 OZT0 = C(KM808 + I808)
M808 = M808 + 1
IF(IOPRNT.LT.3) GO TO 503
OE(303) = C(J808 + I808)
BVAR3 = C(J808 + KD + I808)
CONTINUE
507 IF(M508.EQ.3) GO TO 507
WRITE(IOUT,601)
601 FORMAT(1H1,/,80HVARIABLES OXT, OYT, OZT (IE 306, 307, 308) MUST
      ABE DEFINED AS EITHER TYPE 7 OR 8,/,1H1)
CALL EXIT
507 C(KM808+I808)=DSQRT((OXT0-OE(301))*2+(OYT0-OE(302))*2+
      A(OZT0-OE(303))*2)
DO 221 I808 = 1,K234
C(KM803+I808+KD+2*(I808-1)*KD) = 0.000
C(KM808+I808+2*I808*KD)=0.000
221 IF(IOPRNT.LT.3) GO TO 220
508 AVAR = (DSQRT(BVAR1) + DSQRT(BVAR2) + DSQRT(BVAR3))/3.
C(J803 + I808) = 1.59576900 * AVAR
C(J808 + I808 + KD) = 0.453520900 * AVAR * AVAR
IF(IOPRNT.LT.4) GO TO 220
KAR = (K234-1) * K234/2
DO 222 I808 = 1,KAR
C(KC008+I808*(I808-1)*KD) = 0.000
222 CONTINUE
223 IJ8 = IJ8 - IJ81
IF(IJ8.EQ.0) GO TO 299
225 CONTINUE
IF(IJ82.EQ.0) GO TO 226
M808 = 0
IJ808 = IJ82
IF(IJ81.EQ.0) WRITE(IOUT,9000)
DO 703 I808 = 1,KD

```

HITS1141
HITS1142
HITS1143
HITS1144
HITS1145
HITS1146
HITS1147
HITS1148
HITS1149
HITS1150
HITS1151
HITS1152
HITS1153
HITS1154
HITS1155
HITS1156
HITS1157
HITS1158
HITS1159
HITS1160
HITS1161
HITS1162
HITS1163
HITS1164
HITS1165
HITS1166
HITS1167
HITS1168
HITS1169
HITS1170
HITS1171
HITS1172
HITS1173
HITS1174
HITS1175
HITS1176
HITS1177
HITS1178
HITS1179
HITS1180
HITS1181
HITS1182
HITS1183
HITS1184
HITS1185
HITS1186
HITS1187
HITS1188
HITS1189
HITS1190

D.13 Subroutine D0234 (continued)

```

IF(ID(I808,1).EQ.309) GO TO 704
IF(ID(I808,1).EQ.310) GO TO 705
GO TO 703
704 DYO = C(KM808+I808)
      MB08 = MB08 + 1
      IF(IOPRNT,LT.3) GO TO 703
      OE(304) = C(J808 + I808)
      BVAR1 = C(J808 + I808 + KD)
      GO TO 703
705 DZX = C(KM808+I808)
      MB08 = MB08 + 1
      IF(IOPRNT,LT.3) GO TO 703
      OE(305) = C(J808 + I808)
      BVAR2 = C(J808 + I808 + KD)
      CONTINUE
703 IF(MB08.EQ.2) GO TO 707
      WRITE(IOUT,610)
      FORMAT(1H1,////.74H VARIABLES DYX AND DZX (IE 309, 310) MUST BE DEFINED AS EITHER TYPES 7 OR 8.////.1H1)
      CALL EXIT
707 C(KM808+I808)=DSQRT((DYO-OE(304))*2+(DZX-OE(305))*2)
      DO 23 I808 = 1, K234
      C(KM808+I808+KD+2*(I808-1)*KD) = 0.000
230 C(KM808+I808+2*I808*KD)=0.000
      IF(IOPRNT,LT.3) GO TO 240
708 AVAR = (DSQRT(BVAR1) + DSQRT(BVAR2))/2.
      C(J808 + I808) = 1.253314D0 * AVAR
      C(J808 + I808 + KD) = 0.42920367D0 * AVAR * AVAR
      IF(IOPRNT,LT.4) GO TO 240
      KAR = (K234-1) * K234/2
      DO 232 I808 = 1,KAR
232 C(KC008+I808+(I808-1)*KD) = 0.000
240 CONTINUE
      IJB = IJB - IJB2
      IF(IJB.EQ.0) GO TO _99
226 CONTINUE
      WRITE(IOUT,8001)
8001 FORMAT(1H1,50X,14H*** NOTICE ***.//.
      A30X, THE SPECIFICATION OF EITHER RADT OR RADX (IE .//.
      B30X, VARIABLES 318 OR 319) OR BOTH AS OUTPUT VARIABLES .//.
      C30X, NECESSITATES UP-DATING THE DERIVATIVES. MEANS. .//.
      D30X, AND VARIANCES. PARTIAL DERIVATIVES OF RADT AND RADX ARE .//.
      E30X, MEANINGLESS. THUS THE FULL PROJECTILE DISPERSION CODE .//.
      F30X, SHOULD BE UTILIZED IN MONTE CARLO SIMULATIONS (MCALC = 0) .//.
      G30X, TO OBTAIN MEANINGFUL VALUES OF RADT AND/OR RADX. IF .//.
      H30X, A TAYLOR SERIES APPROXIMATION OF THE PROJECTILE .//.
      I30X, DISPERSION CODE (MCALC = 1, 2, OR 3) IS USED IN MONTE .//.
      J30X, CARLU SIMULATIONS THE RADT AND RADX RESULTS ARE .//.
      K30X, MEANINGLESS. RADT AND/OR RADX MEANS AND VARIANCES ARE .//.
      L30X, TAKEN FROM THE TYPE 8 SPECIFICATION.))

```

HITS11191
 HITS11192
 HITS11193
 HITS11194
 HITS11195
 HITS11196
 HITS11197
 HITS11198
 HITS11199
 HITS1200
 HITS1201
 HITS1202
 HITS1203
 HITS1204
 HITS1205
 HITS1206
 HITS1207
 HITS1208
 HITS1209
 HITS1210
 HITS1211
 HITS1212
 HITS1213
 HITS1214
 HITS1215
 HITS1216
 HITS1217
 HITS1218
 HITS1219
 HITS1220
 HITS1221
 HITS1222
 HITS1223
 HITS1224
 HITS1225
 HITS1226
 HITS1227
 HITS1228
 HITS1229
 HITS1230
 HITS1231
 HITS1232
 HITS1233
 HITS1234
 HITS1235
 HITS1236
 HITS1237
 HITS1238
 HITS1239
 HITS1240

D.13 Subroutine D0234 (continued)

```

IF (IJ83.EQ.0) GO TO 300
IJ808 = IJ83
IF ( IOPRNT.EQ.1) GO TO 350
DO 351 I808 = 1,K234
C(KM803+IJ808+KD+2*(I808-1)*KD) = 0.000
C(KM808+IJ808+2*I808*KD) = 0.000
IF ( IOPRNT.EQ.2) GO TO 350
IF ( ID(IJ808.2) .NE.0) GO TO 301
IF ( IOUT.8002)
WRITE(IOUT,8002)
FORMAT(1H1,///.38H VARIABLE RADT (IE 318) MUST BE TYPE 8.///.1H1)
8002 CALL EXIT
C(J808+IJ808) = 8(ID(IJ808.2)+1)
C(J808+IJ808+KD) = 8(ID(IJ808.2)+3)
IF ( IOPRNT.EQ.3) GO TO 350
KAR = (K234-1)*K234/2
DO 302 I808 = 1,KAR
C(KC0008+IJ808+(I808-1)*KD) = 0.000
CONTINUE
IF ( IJ84.EQ.0) GO TO 299
CONTINUE
IJ808 = IJ84
IF ( IOPRNT.EQ.1) GO TO 299
DO 361 I808 = 1,K234
C(KM808+IJ808+KD+2*(I808-1)*KD) = 0.000
C(KM808+IJ808+2*I808*KD) = 0.000
IF ( IOPRNT.EQ.2) GO TO 299
IF ( ID(IJ808.2) .NE.0) GO TO 362
IF ( IOUT.8003)
WRITE(IOUT,8003)
FORMAT(1H1,///.38H VARIABLE RADX (IE 319) MUST BE TYPE 8.///.1H1)
8003 CALL EXIT
C(J808+IJ808) = 8(ID(IJ808.2)+1)
C(J808+IJ808+KD) = 8(ID(IJ808.2)+3)
IF ( IOPRNT.EQ.3) GO TO 299
KAR = (K234-1)*K234/2
DO 369 I808 = 1,KAR
C(KC0008+IJ808+(I808-1)*KD) = 0.000
CONTINUE
WRITE(IOUT,9003) (BL,IA(I,1),8(IA(I,2)+1),I = 1,K234)
WRITE(IOUT,9005) (BL,ID(I,1), I = 1,KD)
WRITE(IOUT,FMT2) (C(KM808+I), I = 1,KD)
WRITE(IOUT,FMT3) IF ( IOPRNT.GT.1) GO TO 380
WRITE(IOUT,9006) (BL,ID(I,1), I = 1,KD)
DO 381 I = 1,K234
WRITE(IOUT,9007) IA(I,1),8(IA(I,2)+1),8(IA(I,2)+2),
WRITE(IOUT,9007) (C(KM808+2*KD*(I-1)+KD+L), L = 1,KD)
A(C(KM808+2*KD*(I-1)+KD+L), L = 1,KD)
381 WRITE(IOUT,9008) (C(KM808+2*KD*(I-1)+KD+L), L = 1,KD)
380 RETURN
WRITE(IOUT,9009) (ID(I,1), I = 1,KD)
DO 382 I = 1,K234

```

HITS1241
 HITS1242
 HITS1243
 HITS1244
 HITS1245
 HITS1246
 HITS1247
 HITS1248
 HITS1249
 HITS1250
 HITS1251
 HITS1252
 HITS1253
 HITS1254
 HITS1255
 HITS1256
 HITS1257
 HITS1258
 HITS1259
 HITS1260
 HITS1261
 HITS1262
 HITS1263
 HITS1264
 HITS1265
 HITS1266
 HITS1267
 HITS1268
 HITS1269
 HITS1270
 HITS1271
 HITS1272
 HITS1273
 HITS1274
 HITS1275
 HITS1276
 HITS1277
 HITS1278
 HITS1279
 HITS1280
 HITS1281
 HITS1282
 HITS1283
 HITS1284
 HITS1285
 HITS1286
 HITS1287
 HITS1288
 HITS1289
 HITS1290

[illegible]

D.14 Subroutine EXTRA

```

SUBROUTINE EXTRA (A-H,O-Z)
IMPLICIT REAL*8 (A-H,O-Z)
COMMON /CARDCM/ B(200), C(5000), IA(200,2), ID(10,2), INC,
1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KTI, K234, LABC(3)
DIMENSION IB(2,200)
EQUIVALENCE (IB(1,1),B(1))
IF(KTI.EQ.0) GO TO 100

C PRINT RANGE CHECK VALUES
WRITE(IOUT,5000)
DO 1 I = 1,KTI
  IOE = IA(I,1)
  KB = IA(I,2) + 1
  ITL = IB(1,KB)
  KC = IB(1,KB+1)
1 WRITE(IOUT,5001) IOE,(C(KC-1+J),J=1,ITL)
5000 FORMAT(1H1,49X,*** RANGE CHECK VALUES ***.//)
A22X,6X,'CODE',6X,7X,'1',8X,7X,'2',8X,7X,'3',8X,7X,'4',8X,7X,'5',/,
B22X,5X,'NUMBER',5X,5(5X,'VALUE',6X),//)
5001 FORMAT(22X,4X,15,6X,5(3X,1PE11.4,2X),/.5(26X,5(3X,1PE11.4,2X),/),/
A/)

100 CONTINUE
IF(K234.EQ.0) RETURN
DO 2 I = 1, K234
  IOE = IA(I,1)
  KB = IA(I,2)
  IF(IB(1,KB).NE.4) GO TO 2
  WRITE(IOUT,5003) IOE
5003 FORMAT(1H1, 38X,*** CODE NUMBER, 15,1X,
1 PROBABILITY DENSITY FUNCTION
20N ***
A ,//,41X,5X,'POINT',6X,3X,'COORDINATE',3X,2X,'PROBABILITY',3X,/,
B,41X,5X,'NUMBER',5X,5X,'VALUE',6X,4X,'DENSITY',//)
  ITL = IB(1,KB+4)
  KC = IB(1,KB+5)
  DO 3 J = 1,ITL
3 WRITE(IOUT,5004) J,C(KC-1+ITL+J),C(KC-1+J)
2 CONTINUE
5004 FORMAT(40X,6X,14,6X,2(3X,1PE11.4,2X))
  RETURN
END

```

HITS1328
HITS1329
HITS1330
HITS1331
HITS1332
HITS1333
HITS1334
HITS1335
HITS1336
HITS1337
HITS1338
HITS1339
HITS1340
HITS1341
HITS1342
HITS1343
HITS1344
HITS1345
HITS1346
HITS1347
HITS1348
HITS1349
HITS1350
HITS1351
HITS1352
HITS1353
HITS1354
HITS1355
HITS1356
HITS1357
HITS1358
HITS1359
HITS1360
HITS1361
HITS1362
HITS1363
HITS1364
HITS1365
HITS1366
HITS1367
HITS1368

D.15 Subroutine FC2987

```

SUBROUTINE FC2987
  IMPLICIT REAL*8 (A-H,O-Z)
  COMMON/OEC/OE(600)
  EQUIVALENCE
    A(APHMIN),OE( 3)),(RHO
    B(VI),OE( 6)),(CD2
    C(XN),OE( 9)),(CNA
    D(CMO),OE(12)),(CMPA
    E(PNITRI),OE(15)),(ANGO
    F(RATE),OE(18)),(PHIRAT
    G(PHIGAM),OE(21)),(A
    H(D),OE(24)),(W
    I(AIY),OE(27)),(P
    J(CAA),OE(30))
  EQUIVALENCE
    A(ALPTRM),OE( 42)),(ALPHA
    B(BETRM),OE( 45)),(BETAO
    C(BETA),OE( 48)),(ALPMAX
    D(CAYD),OE( 41)),(CD8
    E(CMTHTD),OE( 44)),(CMTHTA
    F(CAY2),OE( 47)),(CDAOB
    G(DELT),OE( 42)),(DELV
    H(DELX),OE( 43)),(DYDXO
    I(DYDT),OE( 46)),(DZDTO
    J(DELLA),OE( 49)),(EYEP
    K(EOLT),OE( 43)),(EDLT
    L(H),OE( 45)),(PSIDJ
    M(PHI1),OE( 48)),(PHI2
    N(RI),OE( 44)),(R2
    O(R4),OE( 44)),(RTRIM
    P(TCO),OE( 47)),(TS
    Q(TP),OE( 50)),(THETAO
    R(TOL),OE( 53)),(VC
    S(VYA),OE( 56)),(VZA
    EQUIVALENCE
    A(WI),OE( 46)),(W2
    B(XLAM),OE( 46)),(XLAM1
    C(XNU1),OE( 47)),(XNU2
    D(XMU),OE( 47)),(XJAY
    E(XJA),OE( 47)),(YA
    F(ALPH),OE( 47))
  EQUIVALENCE
    A(YTFC),OE( 52)),(ZTFC
    B(VYTFC),OE( 55)),(VZTFC
    CALL INCONF
    A57 = ALPCON/57.29577951D0
    CALL XJC(XLAM0,XC,V0,DELLAM,CAY1,CAY2,TOL,A57)
    CALL TRAJXF(XN,XC,VX,TN)
    CALL CROSSF(XN,VX)
    CALL WIND(VX,TN,XN,WZ,WX,V0,VZW,ZW)

```

HITS1369
HITS1370
HITS1371
HITS1372
HITS1373
HITS1374
HITS1375
HITS1376
HITS1377
HITS1378
HITS1379
HITS1380
HITS1381
HITS1382
HITS1383
HITS1384
HITS1385
HITS1386
HITS1387
HITS1388
HITS1389
HITS1390
HITS1391
HITS1392
HITS1393
HITS1394
HITS1395
HITS1396
HITS1397
HITS1398
HITS1399
HITS1400
HITS1401
HITS1402
HITS1403
HITS1404
HITS1405
HITS1406
HITS1407
HITS1408
HITS1409
HITS1410
HITS1411
HITS1412
HITS1413
HITS1414
HITS1415
HITS1416
HITS1417
HITS1418

.OE(1)),(VWIND
.OE(4)),(CD1
.OE(7)),(V2
.OE(10)),(SM
.OE(13)),(TRIM
.OE(16)),(PHIANG
.OE(19)),(GAMO
.OE(22)),(ELL
.OE(25)),(AIX
.OE(28)),(ALPCON
.OE(40)),(ATOT
.OE(43)),(B
.OE(46)),(BETADO
.OE(49)),(ALPMIN
.OE(41)),(CMA
.OE(46)),(CAY1
.OE(49)),(CDAB
.OE(42)),(DELT
.OE(45)),(DZDXO
.OE(48)),(DELW
.OE(43)),(EMP
.OE(43)),(F
.OE(43)),(PHI0
.OE(40)),(PSI0
.OE(44)),(TC
.OE(46)),(TG0
.OE(49)),(VCO
.OE(45)),(WX
.OE(48)),(W0
.OE(46)),(WL2
.OE(46)),(XLAM2
.OE(46)),(XG
.OE(47)),(XJAZ
.OE(47)),(ZA
.OE(50)),(XTFC
.OE(50)),(VXTFC
.OE(506))

D.15 Subroutine FC2987 (concluded)

```

XTFC = XN
YTFC = YA
ZTFC = ZA + ZW
VXTFC = VX
VYTFC = VYA
VZTFC = VZA + VZW
RETURN
END

```

```

HITS1419
HITS1420
HITS1421
HITS1422
HITS1423
HITS1424
HITS1425
HITS1426

```

D.16 Function FEXP

```
1
      FUNCTION FEXP(X)
      IMPLICIT REAL*8(A-M, O-Z)
      IF (X .LT. -90.00) GO TO 1
      FEXP=DEXP(X)
      RETURN
      FEXP=0.00
      RETURN
      END
```

HITS1427
HITS1428
HITS1429
HITS1430
HITS1431
HITS1432
HITS1433
HITS1434

D.17 Subroutine FILLA

```

SUBROUTINE FILLA(A,FCTR,PM,PT,CM,CV,NCELL)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION A(20,5)
THIS ROUTINE FILLS OUT THE OUTPUT ARRAY
CELL=2.00*PT/FCTR
ENN=0.00
SUM1=.00
SUM2=.00
COUNT TOTAL FREQUENCY
DO 1 I=1,NCELL
ENN=ENN+A(I,4)
DO 2 J=1,NCELL
A(I,5)=A(I,4)/ENN
FI=1
FIM=FI-.500
FI=FIM*A(I,5)
SUM1=SUM1+FI
SUM2=SUM2+FI*FIM
A(I,5) = A(I,5) / FCTR
A(I,1)=PM+PT*(2.*FIM/CELL-1.)
A(I,2)=A(I,1)-.500*FCTR
A(I,3)=A(I,2)+FCTR
CV=(SUM2-SUM1**2)*FCTR**2
CM=PM+FCTR*(SUM1-.500*CELL)
RETURN
END

```

HITS1435
HITS1436
HITS1437
HITS1438
HITS1439
HITS1440
HITS1441
HITS1442
HITS1443
HITS1444
HITS1445
HITS1446
HITS1447
HITS1448
HITS1449
HITS1450
HITS1451
HITS1452
HITS1453
HITS1454
HITS1455
HITS1456
HITS1457
HITS1458
HITS1459
HITS1460

D.18 Subroutine FILLIN

```

SUBROUTINE FILLIN(IA ,IOUT)
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/NAMES/HEAD(180)
IAT = IA
IF (IA.LT.0) IAT = - IA
IF (IAT.GT.100) GO TO 1
WRITE(IOUT,8000) HEAD(IAT),IAT
RETURN
CONTINUE
IF (IAT.GT.200) GO TO 2 HEAD(IAT-100),IAT
WRITE(IOUT,8001)
RETURN
CONTINUE
IF (IAT.GT.300) GO TO 3 HEAD(IAT-165),IAT
WRITE(IOUT,8002)
RETURN
CONTINUE
IF (IAT.GE.400) GO TO 4
WRITE(IOUT,8003) HEAD(IAT-259), IAT
RETURN
CONTINUE
IF (IAT.GE.500) GO TO 5
WRITE(IOUT,8004) HEAD(IAT-315), IAT
RETURN
CONTINUE
IF (IAT.GT.600) GO TO 6
WRITE(IOUT,8004) HEAD(IAT-338), IAT
RETURN
CONTINUE
WRITE(IOUT,8005)
8000 FORMAT(1H ,36X,A6,1X,'FIRE CONTROL ',9X,' CODE NUMBER ',13)
8001 FORMAT(1H ,36X,A6,1X,'REAL WORLD ',9X,' CODE NUMBER ',13)
8002 FORMAT(1H ,36X,A6,1X,'SYSTEM CONTROL',9X,' CODE NUMBER ',13)
8003 FORMAT(1H ,36X,A6,1X,'STATISTICAL VARIABLE',3X,' CODE NUMBER ',13)
8004 FORMAT(1H ,36X,A6,1X,'TRAJECTORY VARIABLE ',3X,' CODE NUMBER ',13)
8005 FDRMAT(1H1,50(/,1X,'DE ADDRESS EXCEEDS 600'))
END

```

HITS1461
HITS1462
HITS1463
HITS1464
HITS1465
HITS1466
HITS1467
HITS1468
HITS1469
HITS1470
HITS1471
HITS1472
HITS1473
HITS1474
HITS1475
HITS1476
HITS1477
HITS1478
HITS1479
HITS1480
HITS1481
HITS1482
HITS1483
HITS1484
HITS1485
HITS1486
HITS1487
HITS1488
HITS1489
HITS1490
HITS1491
HITS1492
HITS1493
HITS1494
HITS1495
HITS1496
HITS1497

D.19 Subroutine GQCOR

```

SUBROUTINE GQCOR
  IMPLICIT REAL*8(A-H, O-Z)
  COMMON /CARDOM/ B(200), C(5000), IA(200,2)
  COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV
  IF( K234.EQ. 1) RETURN
  K1 = K234 - 1
  DO 1 I = 1, KD
    DELPP = 0.0
    DO 2 J = 1, K234
      NV = NVARX( J)
      VARXJ = B(NV)
      DO 2 L = 1, K1
        N2L = N2D(I,L, J)
        DYXLJ = C( N2L)
        LI = L + 1
        DO 2 K = LI, K234
          N2K = N2D(I,K,J)
          NC = NCOV(L,K)
          DELPP = DELPP + DYXLJ * C(N2K) * C(NC) * VARXJ
        DO 3 J = 1, K1
          J1 = J + 1
          DO 3 M = J1, K234
            NC = NCOV(J,M)
            COVJM = C(NC)
            DO 3 L = 1, K1
              N2 = N2D(I,L, J)
              DYLJ = C(N2)
              LI = L + 1
              DO 3 K = LI, K234
                N2 = N2D(I,K,M)
                NC = NCOV(L,K)
                DELPP = DELPP + DYLJ * C(N2) * C(NC) * COVJM
              DELPP = 2.0 * DELPP
            NY = NVAR(I)
            C(NY) = C(NY) + DELPP
          RETURN
        END
      DO 1 I = 1, KD
    END
  END

```

HITS1498
 HITS1499
 HITS1500
 HITS1501
 HITS1502
 HITS1503
 HITS1504
 HITS1505
 HITS1506
 HITS1507
 HITS1508
 HITS1509
 HITS1510
 HITS1511
 HITS1512
 HITS1513
 HITS1514
 HITS1515
 HITS1516
 HITS1517
 HITS1518
 HITS1519
 HITS1520
 HITS1521
 HITS1522
 HITS1523
 HITS1524
 HITS1525
 HITS1526
 HITS1527
 HITS1528
 HITS1529
 HITS1530
 HITS1531
 HITS1532
 HITS1533
 HITS1534

D.20 Subroutine GQUC

```

SUBROUTINE GQUC
  IMPLICIT REAL*8(A-H,O-Z)
  COMMON /CARDCM/ B(200), C(5000), IA(200,2)
  COMMON INDEX/ KC, K234, KCYBAR, KCSUMS, KCDOUB, KCOV
  KC = K234 - 1
  DO 1 I = 1, KD
    SUM = 0.00
    DO 2 J = 1, K234
      N2 = N2D(I,J,J)
      NX = NVARX(J)
      SUM = SUM + (C(N2) * B(NX)) ** 2
      IF( K1.EQ. 0 ) GO TO 4
      DO 3 J = 1, K1
        NXJ = NVARX(J)
        BNXJ = B(NXJ)
        KB = J + 1
        DO 3 K = KB, K234
          N2 = N2D(I,J,K)
          NXK = NVARX(K)
          SUM = SUM + C(N2) ** 2 * BNXJ * B(NXK)
          NY = NVAR(I)
          C(NY) = C(NY) + SUM
        RETURN
      END
    1
  2
  3
  4
  1

```

HITS1535
 HITS1536
 HITS1537
 HITS1538
 HITS1539
 HITS1540
 HITS1541
 HITS1542
 HITS1543
 HITS1544
 HITS1545
 HITS1546
 HITS1547
 HITS1548
 HITS1549
 HITS1550
 HITS1551
 HITS1552
 HITS1553
 HITS1554
 HITS1555
 HITS1556
 HITS1557
 HITS1558
 HITS1559

D.21 Subroutine G1795

SUBROUTINE G1795
RETURN
END

HITS1560
HITS1561
HITS1562

D.22 Subroutine G2440

SUBROUTINE G2440
RETURN
END

HITS1563
HITS1564
HITS1565

D.23 Subroutine G2987

```

SUBROUTINE G2987
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/DEC/OE(600)
COMMON /CISPRN/ ISPRNT
COMMON /CINOM/ INJM
EQUIVALENCE
  A(ZBART,OE( 133)), (XBART
  EQUIVALENCE (YBAPX
  EQUIVALENCE (IFCRW
  A(IFC,OE( 203)), (ICONV
  EQUIVALENCE (XBARTI
  A(ZBARTI,OE( 303)), (YBARTI
  B(DXT,OE( 306)), (DYT
  C(DYX,OE( 309)), (DZX
  D(DVYT,OE( 312)), (DVZT
  E(DVYX,OE( 315)), (DVZX
  F(RADT,OE( 318)), (RADX
  G(RHOX,OE( 321)), (DXTI
  H(DZTI,OE( 324)), (DXTI
  I(DZTE,OE( 327)), (DXTI
  J(DYXE,OE( 330)), (DZXE
  K(DXDZT,OE( 333)), (DYDZT
  L(DXDZTI,OE( 336)), (DXDZTI
  M(DYDZXI,OE( 339)), (DXDYTE
  N(DYDZTE,OE( 342)), (DYDZXE
  EQUIVALENCE (TOL
  EQUIVALENCE (TN
  A(VTFC,OE( 502)), (ZTFC
  B(VTFC,OE( 505)), (VZTFC
  EQUIVALENCE (XT
  A(ZT,OE( 509)), (VXT
  B(VZT,OE( 512)), (VYR
  C(VXR,OE( 515)), (VYR
  D(TX,OE( 518))

```

FIRE CONTROL LOGIC

```

IF((.NOT.(IFCRW.EQ.1).AND.(IFC.EQ.1)).AND.(IFCRC.NE.1)) GO TO 1
DO 2 I = 1,100
DE(I) = OF(I+100)
CONTINUE
IF((IFC.NE.1).AND.(IFCRC.NE.1)) GO TO 3
TOL = 10.*ICONV
CALL FC2987
IF(INOM.EQ.1) GO TO 500
IF(ISPRNT.EQ.0) GO TO 600
CALL SPRNT

```

C
C
C
C
C
C

2
1

50-

HITS1566
HITS1567
HITS1568
HITS1569
HITS1570
HITS1571
HITS1572
HITS1573
HITS1574
HITS1575
HITS1576
HITS1577
HITS1578
HITS1579
HITS1580
HITS1581
HITS1582
HITS1583
HITS1584
HITS1585
HITS1586
HITS1587
HITS1588
HITS1589
HITS1590
HITS1591
HITS1592
HITS1593
HITS1594
HITS1595
HITS1596
HITS1597
HITS1598
HITS1599
HITS1600
HITS1601
HITS1602
HITS1603
HITS1604
HITS1605
HITS1606
HITS1607
HITS1608
HITS1609
HITS1610
HITS1611
HITS1612
HITS1613
HITS1614
HITS1615

(XBART,OE(131)), (YBART
(YBAPX,OE(134)), (ZBAPX
(IFCRW,OE(201)), (IFCRC
(ICONV,OE(205)), (IFCRC
(XBARTI,OE(301)), (YBARTI
(YBARTI,OE(304)), (ZBARTI
(DYT,OE(307)), (DZT
(DZX,OE(310)), (DVXT
(DVZT,OE(313)), (DVXX
(DVZX,OE(316)), (DT
(RADX,OE(319)), (RHOT
(DXTI,OE(322)), (DYTE
(DZXE,OE(325)), (DZTE
(DYDZT,OE(328)), (DZXT
(DXDZT,OE(331)), (DXDYT
(DXDZTI,OE(334)), (DXDZT
(DXDYTE,OE(337)), (DYDZT
(DYDZXE,OE(340)), (DXDZT
(TOL,OE(453)), (XTFC
(TN,OE(500)), (VXTFC
(ZTFC,OE(503)), (VZTFC
(VZTFC,OE(506)), (VZT
(VXT,OE(507)), (VYR
(VYR,OE(510)), (VYR
(VYR,OE(513)), (VYR
(VYR,OE(516)), (VYR

D.23 Subroutine G2987 (continued)

```

600 CONTINUE
3   IFC = C
    CALL RW2987
C*****
C   INCREMENTAL OUTPUTS
C*****
DXT = XT      - XTFC
DYT = YT      - YTFC
DZT = ZT      - ZTFC
DZX = YR      - YTFC
DVXT = VXT    - VXTFC
DVYT = VYT    - VYTFC
DVZT = VZT    - VZTFC
DVXX = VXR    - VXTFC
DVYX = VYR    - VYTFC
DVZX = VZR    - VZTFC
DT = TX       - TN
DXTI = DXT    - XBARTI
DYTI = DYT    - YBARTI
DZTI = DZT    - ZBARTI
DYTE = DXT    - XBART
DZTE = DZT    - YBART
DXXI = DYX    - YBARXI
DZXI = DZX    - YBARXI
DZXE = DZX    - ZBARX
C*****
SEP AND CEP CALCULATIONS USING INTERNAL AND EXTERNAL CENTERS
C*****
RADT = DSQRT(DYTE**2 + DYTE**2 + DZTE**2)
RADX = DSQRT(DYXE**2 + DZXE**2)
RROT = DSQRT(DXTI**2 + DYTI**2 + DZTI**2)
RHOX = DSQRT(DYXI**2 + DZXI**2)
C*****
CORRELATIONS
C*****
DXDYT = DXT * DYT
DXDZT = DXT * DZT
C*****
HITS1616
HITS1617
HITS1618
HITS1619
HITS1620
HITS1621
HITS1622
HITS1623
HITS1624
HITS1625
HITS1626
HITS1627
HITS1628
HITS1629
HITS1630
HITS1631
HITS1632
HITS1633
HITS1634
HITS1635
HITS1636
HITS1637
HITS1638
HITS1639
HITS1640
HITS1641
HITS1642
HITS1643
HITS1644
HITS1645
HITS1646
HITS1647
HITS1648
HITS1649
HITS1650
HITS1651
HITS1652
HITS1653
HITS1654
HITS1655
HITS1656
HITS1657
HITS1658
HITS1659
HITS1660
HITS1661
HITS1662
HITS1663
HITS1664
HITS1665

```

D.23 Subroutine G2987 (concluded)

```

DYZT = DYT
DYDZX = DYX
DXOYTI = DXTI
DXDZTI = DXTI
DYDZTI = DXTI
DYDZXI = DXTI
DXDYTE = DXTE
DXDZTE = DXTE
DYDZTE = DYTE
DYDZXE = DYXE
C INOM = 1 IN MAIN.
IF ( INOM.EQ.1 ) GO TO 260
IF ( ISPRNT.EQ.3 ) RETURN
260 CALL SPRNT
INOM = 2
RETURN
END

DZT
DZX
DYTI
DZTI
DZTI
DZTI
DZTI
DZTE
DZTE
DZXE
HITS1666
HITS1667
HITS1668
HITS1669
HITS1670
HITS1671
HITS1672
HITS1673
HITS1674
HITS1675
HITS1676
HITS1677
HITS1678
HITS1679
HITS1680
HITS1681
HITS1682

```

D.24 Subroutine HSTG1

```

SUBROUTINE HSTG1(IER)
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/CARDCM/ B(200),C(500),IA(200,2),ID(10,2),INC,IOPRNT,IOUT,
IY,KARSI(3),KA,KB,KC,KD,KT1,K234,LABC(3)
COMMON/DEC/OE(600)
COMMON/INCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),
IR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,MCC,NR1,NR2
EQUIVALENCE (KADD,KC)
THIS SUBROUTINE CHECKS THE SAMPLE OUT PUT AGAINST THE INTERNALLY
GENERATED HISTOGRAM RANGES-IF A REJECT OCCURS IT SETS IER=-1
THIS SUBROUTINE REQUIRES KADD THE NOMINAL BLOCK ADDRESS IN C = KCH
ENN=NCELL
IBN=1
LADD=2*KD* K234 + KD + (KADD - 1)
DO 1 I=1,KD
IADD=ID(I,1)
IADD IS ADDRESS OF OUTPUT VARIABLE VALUE-CALCULATE ADDRESSES
OF VARIABLE MEAN AND TOLERANCE VALUES IN C ARRAY
IMN=LADD+I
IVAR=IMN+KD
TOL=3.*DSQRT(C(IVAR))
FJ=(OE(IADD)-C(IMN)+TOL)*.5*ENN/TOL
IF ( FJ ) 4,77,77
77 J=FJ
J=J+1
J IS HISTOGRAM INTERVAL TO BE INCREMENTED-CHECK IF VALID
IF(J)4,4,2
IF(J-NCELL)3,3,4
IF J LT ZERO OR GT NCELL REJECT SAMPLE
IR1(I)=IR1(I)+1
IDT(I)=0
IER=-1
RETURN
STORE HISTOGRAM CELL ADDRESS TEMPORARILY
IDT(I)=IBN+J-1
IBN=IBN+NCELL
IF THIS POINT REACHED THE SAMPLE IS OK SO INCREMENT HISTOGRAMS
DO 5 I=1,KD
L=IDT(I)
ID1(L)=ID1(L)+1
DO 6 I=1,K234
L=IIT(I)
I11(L)=I11(L)+1
RETURN
END

```

HITS1683
HITS1684
HITS1685
HITS1686
HITS1687
HITS1688
HITS1689
HITS1690
HITS1691
HITS1692
HITS1693
HITS1694
HITS1695
HITS1696
HITS1697
HITS1698
HITS1699
HITS1700
HITS1701
HITS1702
HITS1703
HITS1704
HITS1705
HITS1706
HITS1707
HITS1708
HITS1709
HITS1710
HITS1711
HITS1712
HITS1713
HITS1714
HITS1715
HITS1716
HITS1717
HITS1718
HITS1719
HITS1720
HITS1721
HITS1722
HITS1723
HITS1724
HITS1725
HITS1726
HITS1727

D.25 Subroutine HSTG2

```

SUBROUTINE HSTG2(IER)
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/CAPDCM/ B(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT,
IY,KARSI(3),KA,KB,KC,KD,KTI,K234,LABC(3)
COMMON/OEC/OE(600)
COMMON/INCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),
IIR1(1),ID2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2
EQUIVALENCE (KADD,KC)
THIS ROUTINE CHECKS THE SAMPLE OUTPUT AGAINST THE USER SPEC
ENN=NCELL
IBN=1
DO 1 I=1,KD
L=ID(I,2)
C CHECK IF IT IS A TYPE 8 VARIABLE
IF(L)8,8,2
C IF IT IS NOT A TYPE 8 CHECK IF IT PASSED THE PREVIOUS TIME
IF (IDT(I))4,4,1
C IF NOT A TYPE 8 AND REJECTED BEFORE REJECT IT AGAIN IF PASSED
C REFOPE PASS IT AGAIN
N=ID(I,1)
FJ=(OE(N)-B(L+1)+B(L+2))*S*ENN/B(L+2)
IF ( FJ ) 4,77,77
77 J=FJ
J=J+1
C J IS THE HISTOGRAM INTERVAL NUMBER CHECK IF VALID
IF(J)4,4,3
IF(J-NCELL)5,5,4
IR2(I)=IR2(I)+1
IER=-1
RETURN
5 IDT(I)=IRN+J-1
1 IBN=IBN+NCELL
C IF GETS HERE IT IS SUCCESSFUL SO INCREMENT HISTOGRAMS
DO 6 I=1,KD
L=IDT(I)
ID2(L)=ID2(L)+1
DO 7 I=1,K234
L=IIT(I)
I12(L)=I12(L)+1
7 RETURN
END

```

HITS1728
HITS1729
HITS1730
HITS1731
HITS1732
HITS1733
HITS1734
HITS1735
HITS1736
HITS1737
HITS1738
HITS1739
HITS1740
HITS1741
HITS1742
HITS1743
HITS1744
HITS1745
HITS1746
HITS1747
HITS1748
HITS1749
HITS1750
HITS1751
HITS1752
HITS1753
HITS1754
HITS1755
HITS1756
HITS1757
HITS1758
HITS1759
HITS1760
HITS1761
HITS1762
HITS1763
HITS1764
HITS1765
HITS1766
HITS1767
HITS1768

D.26 Subroutine HYSPLT

```

SUBROUTINE HYSPLT
  IMPLICIT REAL * 8(A-H, O-Y)
  COMMON/ HYSPLT/ IOPLT, ISPLT
1  COMMON /CARDCH / B(200), C(5000), IA(200,2), ID(10,2), INC
   1 .IOPRNT, IOUT, IY, KARS(3), KA, KB, KC, KD, KI, K234
2  , LABC(3)
   NCELL1 = ZTITLE(18,1) + 1
1500 WRITE(IOUT,1500)
1000 FORMAT(1H1)
   WRITE(IOUT,1000)
1000 FORMAT(1H1, ZTITLE, )
   K = KD + K234
   DO 1 J = 1, K
1  WRITE(IOUT,1001) J, (ZTITLE(I,J), I=1,20)
1001 FORMAT('C',15,6X,1P10E12.4,/,12X,1P10E12.4)
   DO 2 J = 1, K
2  WRITE(IOUT,1002) J, ((ZDATAD( I, L, J), L = 1,4), I=1,NCELL
11)
1002 FORMAT(1H1, ZDATAD, J=,13, I=1
1=3 I=4, / (15X,1P4E12.4), )
   RETURN
   END

```

HITS1769
 HITS1770
 HITS1771
 HITS1772
 HITS1773
 HITS1774
 HITS1775
 HITS1776
 HITS1777
 HITS1778
 HITS1779
 HITS1780
 HITS1781
 HITS1782
 HITS1783
 HITS1784
 HITS1785
 HITS1786
 HITS1787
 HITS1788
 HITS1789
 HITS1790
 HITS1791
 HITS1792

D.27 Subroutine INCARD

```

SUBROUTINE INCARD
IMPLICIT REAL*8 (A-H,O-Z)
COMMON /CARDCM/ B(200), C(5000), IA(200,2), ID(10,2), INC,
1 IOPRNT, ICUT, IY, KARSI(3), KA, KB, KC, KD, KTI, K234, LABC(3)
COMMON/OEC/OE(600)
DIMENSION IOE(2,600)
DIMENSION IB(2,200)
EQUIVALENCE (IB(1,1), B(1)), (IOE(1,1), OE(1))
DO 15 I=1,3
15 LABC(I) = 1
C KTI WILL COUNT TYPE 1, K234 WILL COUNT TYPES 2, 3 AND 4.
C KD FOR THE DEPENDENT VARIABLES.
C KB FOR THE CURRENT ENTRY IN B.
C KC FOR THE CURRENT ENTRY IN C.
KTI = 0
K234 = 0
KD =
KB =
KC =
1000 READ (INC,9002) ISOE, ITYPE, FNOM, TCL, STDEV, ITL
9002 FORMAT(15, 12, 3F13.1, I4)
IF ( ISOE .LT. 0 ) GO TO 1100
WRITE(ICUT,5000) ISOE, ITYPE, FNOM, TCL, STDEV, ITL
5000 FORMAT(10X,5X,15,6X,7X,12,7X,3(3X,1PE11.4,2X),6X,15,/)
VAR = STDEV * STDEV
GO TO (100, 203, 203, 400, 500, 999, 708, 708), ITYPE
100 KTI = KTI + 1
IF ( KTI .LE. 3 ) GO TO 110
WRITE(ICUT,9004) ISOE
9004 FORMAT(26H MORE THAN 3 TYPE 1, NAME= , I4)
1000 IEXIT = 1
GO TO 1080
110 ITLADD = 1
GO TO 1010
C 120 SAVE THE KAR SUBSCRIPT FOR LATER MOVING THE TYPE 1 INFO UP TO
C THE START OF IA.
120 KARSI(KTI) = KAR
LABC(KTI) = ITL
GO TO 200
203 K234 = K234 + 1
GO TO 110
C 210 SAVE 2 SLOTS IN IB (FOR TABLE LENGTH AND ADDRESS IN C).
210 KB = KB + 2
GO TO 710
400 K234 = K234 + 1
C SKIP 3 SLOTS FOR NOM, TCL AND VAR.
ITLADD = 4
C 14 KEEPS TRACK OF READING THE W AND Y TABLES.
14 = 1

```

HITS1793
 HITS1794
 HITS1795
 HITS1796
 HITS1797
 HITS1798
 HITS1799
 HITS1800
 HITS1801
 HITS1802
 HITS1803
 HITS1804
 HITS1805
 HITS1806
 HITS1807
 HITS1808
 HITS1809
 HITS1810
 HITS1811
 HITS1812
 HITS1813
 HITS1814
 HITS1815
 HITS1816
 HITS1817
 HITS1818
 HITS1819
 HITS1820
 HITS1821
 HITS1822
 HITS1823
 HITS1824
 HITS1825
 HITS1826
 HITS1827
 HITS1828
 HITS1829
 HITS1830
 HITS1831
 HITS1832
 HITS1833
 HITS1834
 HITS1835
 HITS1836
 HITS1837
 HITS1838
 HITS1839
 HITS1840
 HITS1841
 HITS1842

D.27 Subroutine INCARD (continued)

```

      GO TO 1010
410 IF ( I4.EQ. 2 ) GO TO 420
      IA = 2
      KC = KC + 1
      GO TO 1070
420 IAW = IB(1,KB) - 1
      IAY = IAW + ITL
      KBR = KB - 5
      CALL TANTV ( IAW, IAY, ITL, KBR )
      C PUT NOMINAL VALUE INTO OE.
      LOE = IA(KAR,1)
      OE(LOE) = B(KBR+1)
      GO TO 20
500 GO TO 1010
      C 510 PUT A -1 IN COLUMN 2 FOR A LATER CHECK.
      510 IA(KAR,2) = -1
      C IS IT AN INTEGER.
      LOE = IA(KAR,1)
      IF ( LOE.GT. 0 ) GO TO 710
      LOE = - LOE
      IOE(1,LOE) = FNOM
      GO TO 20
700 KD = KD + 1
      IO(KD,1) = ISOE
      IF ( ITYPE.EQ. 7 ) GO TO 20
      KB = KB + 1
      IO(KD,2) = KB
      ITLADD = 1
      GO TO 1045
710 OE(ISOE) = FNOM
      GO TO 20
999 WRITE (IOUT,1000)
1000 FORMAT(25H INCARD ERROR, CALL EXIT.)
      CALL EXIT
1010 KAR = 0
1020 KAR = KAR + 1
      IF ( IABS(IA(KAR,1)) .EQ. ISOE ) GO TO 1030
      IF ( KAR.LT. KA ) GO TO 1020
      IEXIT = 1
      WRITE (IOUT,9003) ISOE, ITYPE
      FORMAT(24H INCARD, COULD NOT FIND, I4, 16H IN IA FOR TYPE, I2)
      IF ( ITYPE.EQ. 1 .OR. ITYPE.EQ. 4 ) GO TO 1080
      GO TO 20
1030 GO TO (1040, 1040, 1040, 1040, 510, 999, 999, 999), ITYPE
1040 KB = KB + 1
      IA(KAR,2) = KB
1045 IB(1,KB) = ITYPE
      KB = KB + ITLADD
      GO TO ( 1060, 1050, 1050, 999, 999, 999, 1050), ITYPE
1050 B(KB) = FNOM

```

HITS1843
 HITS1844
 HITS1845
 HITS1846
 HITS1847
 HITS1848
 HITS1849
 HITS1850
 HITS1851
 HITS1852
 HITS1853
 HITS1854
 HITS1855
 HITS1856
 HITS1857
 HITS1858
 HITS1859
 HITS1860
 HITS1861
 HITS1862
 HITS1863
 HITS1864
 HITS1865
 HITS1866
 HITS1867
 HITS1868
 HITS1869
 HITS1870
 HITS1871
 HITS1872
 HITS1873
 HITS1874
 HITS1875
 HITS1876
 HITS1877
 HITS1878
 HITS1879
 HITS1880
 HITS1881
 HITS1882
 HITS1883
 HITS1894
 HITS1885
 HITS1886
 HITS1887
 HITS1888
 HITS1889
 HITS1890
 HITS1891
 HITS1892

D.27 Subroutine INCARD (concluded)

```

      KB = KB + 1
      B(KB) = TOL
      KB = KB + 1
      B(KB) = VAR
      GO TO ( 999, 210, 210, 999, 999, 999, 999, 710), ITYPE
1060  IB(1,KB) = ITL
      KC = KC + 1
      KB = KB + 1
      IB(1,KB) = KC
      KCM1 = KC - 1
      READ (INC,9005) (C(I+KCM1), I=1,ITL)
      FORMAT(5F13.1)
      C THE LAST ADDRESS USED IS KCM1 + ITL.
      KC = KCM1 + ITL
      GO TO ( 120, 999, 999, 410, 999, 999, 999), ITYPE
1080  READ (INC,9005) (C(I), I=1,ITL)
      IF ( ITYPE .EQ. 1 ) GO TO 20
      C FOR ITYPE=4 WE HAVE TO BYPASS BOTH W AND Y TABLES.
      ITYPE = 1
      GO TO 1080
1100  CONTINUE
      C FINISHED READING ALL THE INPUT CARDS. WERE THERE ANY ERRORS.
      IF ( IEXIT .EQ. 1 ) CALL EXIT
      RETURN
      END

```

HITS1893
 HITS1894
 HITS1895
 HITS1896
 HITS1897
 HITS1898
 HITS1899
 HITS1900
 HITS1901
 HITS1902
 HITS1903
 HITS1904
 HITS1905
 HITS1906
 HITS1907
 HITS1908
 HITS1909
 HITS1910
 HITS1911
 HITS1912
 HITS1913
 HITS1914
 HITS1915
 HITS1916
 HITS1917

D.28 Subroutine INCON

```

SUBROUTINE INCON
  IMPLICIT REAL * 8 (A-H, O-Z)

  COMMON/OE/OE(600)
  EQUIVALENCE
    A(PHIWIN), OE(103), (V0), (RHO), (OE(101)), (VWIND), (OE(102)),
    B(VI), OE(106), (CD2), (CD1), (OE(104)), (CD1), (OE(105)),
    C(XN), OE(109), (CNA), (OE(107)), (V2), (OE(108)),
    D(CMQ), OE(112), (CMPA), (OE(110)), (SM), (OE(111)),
    E(PNITRI), OE(115), (ANG), (OE(113)), (TRIM), (OE(114)),
    F(RATEO), OE(113), (PHIRAT), (OE(116)), (PHIANG), (OE(117)),
    G(PHIGAM), OE(121), (A), (OE(119)), (GAMO), (OE(120)),
    H(D), OE(124), (W), (OE(122)), (ELL), (OE(123)),
    I(AIY), OE(127), (P), (OE(125)), (AIX), (OE(126)),
    J(CAA), OE(130), (IDUMP), (OE(128)), (ALPCON), (OE(129))

  EQUIVALENCE
    A(ALPTRM), OE(402), (ALPHA), (OE(204)), (ICNCL), (OE(206)),
    B(BETRM), OE(405), (ALPHD), (OE(400)), (ATOT), (OE(401)),
    C(BETA), OE(408), (BETA0), (OE(403)), (B), (OE(404)),
    D(CAYD), OE(411), (ALPMAX), (OE(406)), (BETADO), (OE(407)),
    E(CMTHTD), OE(414), (CD8), (OE(412)), (CMA), (OE(410)),
    F(CAY2), OE(417), (CMTHTA), (OE(415)), (CAY1), (OE(416)),
    G(DELTY), OE(420), (CDA08), (OE(418)), (CDA8), (OE(419)),
    H(DELX), OE(423), (DELY), (OE(421)), (DEL), (OE(422)),
    I(DYDYO), OE(426), (DYDX0), (OE(424)), (DZDX0), (OE(425)),
    J(DELAM), OE(429), (DZDYO), (OE(427)), (DELP), (OE(428)),
    K(FOLT), OE(432), (EYEP), (OE(430)), (EMP), (OE(431)),
    L(Y), OE(435), (EDLT), (OE(433)), (F), (OE(434)),
    M(PHI1), OE(438), (PSID0), (OE(436)), (PHI0), (OE(437)),
    N(RI), OE(441), (PHI2), (OE(439)), (PSI0), (OE(440)),
    O(R4), OE(444), (R2), (OE(442)), (R3), (OE(443)),
    P(TCO), OE(447), (RTRIM), (OE(445)), (TC), (OE(446)),
    Q(TP), OE(450), (TS), (OE(448)), (TGO), (OE(449)),
    R(TOL), OE(453), (THETA0), (OE(451)), (THETD0), (OE(452)),
    S(VYA), OE(456), (VC), (OE(454)), (VCO), (OE(455)),
    EQUIVALENCE
      (WZ), (VZA), (OE(457)), (WX), (OE(458)),
      (W2), (WZ), (OE(459)), (W0), (OE(460)),
      (XLAM0), OE(461), (W2), (OE(462)), (WL2), (OE(463)),
      B(XLAMO), OE(464), (XLAM1), (OE(465)), (XLAM2), (OE(466)),
      C(XMU1), OE(467), (XLU2), (OE(468)), (XG), (OE(469)),
      D(XMU), OE(470), (XJAY), (OE(471)), (XJAZ), (OE(472)),
      E(XJA), OE(473), (YA), (OE(474)), (ZA), (OE(475))

  DATA RFAC, G
  WX = VWIND
  WZ = PHIWIN

  V1 = VI
  V12 = (CD2 - CD1) / (1.00 / V2/V2 - 1.00 / V12)
  CAYD = (CD2 - CAYD) / V12
  CD8 = CD1 - CAYD / V12
  B = 1.00 + (WZ / (V0 - WX)) ** 2
  F = RHO * A * G / 2.00 / W

  / 57.2957795100 * 32.17400 /

```

HITS1918
 HITS1919
 HITS1920
 HITS1921
 HITS1922
 HITS1923
 HITS1924
 HITS1925
 HITS1926
 HITS1927
 HITS1928
 HITS1929
 HITS1930
 HITS1931
 HITS1932
 HITS1933
 HITS1934
 HITS1935
 HITS1936
 HITS1937
 HITS1938
 HITS1939
 HITS1940
 HITS1941
 HITS1942
 HITS1943
 HITS1944
 HITS1945
 HITS1946
 HITS1947
 HITS1948
 HITS1949
 HITS1950
 HITS1951
 HITS1952
 HITS1953
 HITS1954
 HITS1955
 HITS1956
 HITS1957
 HITS1958
 HITS1959
 HITS1960
 HITS1961
 HITS1962
 HITS1963
 HITS1964
 HITS1965
 HITS1966
 HITS1967

D.28 Subroutine INCON (continued)

```

1000 IF(IDUMP, EQ, 1) WRITE(6, 1000) WX, WZ, CAYD, CD8, B, F, CD8
      FORMAT(//, F, WZ, CAYD, CD8, B, F, CD8, /)
      THETA0 = ANGO/RFAC
      PSIO = PHANG/RFAC
      THETD0 = RATE0
      PSID0 = PHIRAT
      DYDX0 = GAM0/RFAC
      DZDX0 = PHIGAM/RFAC
      DYDT0 = V0 * DYDX0
      DZDT0 = V0 * DZDX0
      ALPH0 = THETA0 - DYDX0
      BETAD0 = DZDX0 - PSIO
      IF(IDUMP, EQ, 1) WRITE(6, 1001)
      PSID0, DYDX0, DZDX0, DYDT0, DZDT0
      FCV = F * CNA * V0
      ALPHD0 = THETD0 - FCV * ALPH0 - P * BETAD0
      IF(ICNCL, EQ, 1) ALPHD0 = ALPHD0 + F * V0 * THETA0 * (CD8 + CAYD/V0/V0)
      BETAD0 = -PSID0 - FCV * BETAD0 + P * ALPH0
      IF(ICNCL, EQ, 1) BETAD0 = BETAD0 - F * V0 * PSIO * (CD8 + CAYD/V0/V0)
      CNA = -CNA * ELL * SM / D
      EYEP = 2.00 * AIY / RHO / A / D / V0/V0
      EMP = 2.00 * W / RHO / A / G / V0
      D2V = D / 2.00 / V0
      CMTHTD = CMQ * D2V
      CMTHTA = CMA * D2V
      H = ZZ(CNA, CD8, CAYD, V0, EMP, ICNCL)
      CCON = CNA / EMP + CMTHTD / EYEP
      PII = P * AIX / AIY
      WCON = -4.00 * CMA / EYEP + PII * PII - CCON * CCON
      IF(IDUMP, EQ, 1) WRITE(6, 1002) ALPH0, BETAD0, ALPHD0, BETAD0, CMA, EYEP
      1, EMP, CMTHTD, CMTHTA, H
      W0 = DSQRT(.125 * (WCON + DSQRT(WCON * WCON + 4.00 * (PII * PII - CCON * CCON)))
      1 * CCON + 2.00 * P * CMTHTA / EYEP ** 2)
      IF(CMA, LT, 0.) GO TO 3
      PCR=0.00
      XMU=0.00
      RTRIM=1.00
      PHIROL=0.00
      GO TO 4
      3 CONTINUE
      PCRV=DSQRT(-CMA*RHO * A / 2.00 * D / (AIY-AIX))
      PCR=PCRV*DABS(V0)
      XMU=(CNA*RHO/4.00 * A/W * G-CMQ*D*D/R.D0*RHO*A/(AIY-AIX))/PCRV
      PPCR=P/PCR
      PPCR1=1.00-PPCR**2
      XMU2=2.00 * XMU * PPCR
      RTRIM=1.00 / DSQRT(PPCR1**2 + XMU2**2)
      PHIROL=DATAN(XMU2/PPCR1)
      IF(PPCR1, LT, 0.) PHIROL=PHIROL+3.141592653589800

```

D.28 Subroutine INCON (continued)

```

ABAR = DSORT( TRIM**2 + PNITRI**2)
PNITRR = ZATAN2(PNITRI, TRIM)
CONTINUE
ALPTRM = RTRIM * ABAR / RFAC
PTPR = PNITRR + PHIROL
BETTRM = ALPTRM * DSIN( PTPR )
ALPTRM = ALPTRM * DCOS( PTPR )
IF( IDUMP .EQ. 1 ) WRITE( 6, 1006 ) XMU , RTRIM, PHIROL, ALPTRM
1006 FORMAT( 1X, BETTRM, PPR, XMU, RTRIM, PHIROL, ALPTRM )
2
IX, IP10E12.4 //)
DELR = P - PII / 2.00
XLAM0 = .500* ( CMTHTD / EYEP - CNA/EMP )
DELLAM = PII / 4.00 / W0 * CCON + P * CMTHTA / 2.00 / W0 / EYEP
W1 = W0 - DELR
W2 = W0 + DELR
AA = ALPH0 - ALPTRM
XLAM1 = XLAM0 + DELLAM
XLAM2 = XLAM0 - DELLAM
BB = BETA0 - BETTRM
RCON1 = ALPHD0 + W2 * BB
RCON2 = BETA0 - W2 * AA
WL2 = 2.00 * W0 * W0 + DELLAM ** 2 )
R1 = ( W0 * RCON1 + DELLAM * RCON2 ) / WL2
R2 = ( -W0 * RCON2 + DELLAM * RCON1 ) / WL2
IF( IDUMP .EQ. 1 ) WRITE( 6, 1004 ) W0, DELR, XLAM0, DELLAM, W1, W2
1004 FORMAT( 1X, XLAM1, XLAM2, WL2, W0, DELR, XLAM0, DELLAM, W1, W2 )
2
R2, 1X, IP11E12.4 // )
R4 = AA - R2
R3 = BB - R1
CAY1 = DSORT( R1 * R1 + R2 * R2 )
CAY2 = DSORT( R3 * R3 + R4 * R4 )
XNU1 = ZATAN2( R2, R1 )
XNU2 = ZATAN2( R4, R3 )
PHI1 = DATAN2( -W1 - P, XLAM1 )
PHI2 = DATAN2( -W2 + P, XLAM2 )
PHI0 = DATAN2( -W0, XLAM0 )
1001 FORMAT( 1X, PSI00, DYDX0, DYD10, DYD20, PSI0, DZD10, THE )
2
1002 FORMAT( //, ALPH0, EMP, BETA0, CMTHTD, CMTHTA, ALPHD0, CMTHTA, BETAD0, H, )
1X, IP11E12.4 / )
IF( IDUMP )
1, XNU1, XNU2, PHI1, PHI2, PHI0, CAY1, CAY2, XNHITS2066
1005 FORMAT( 1X, XNU2, R3, PHI1, PHI2, PHI0, CAY1, CAY2, XNHITS2066 )

```

D.28 Subroutine INCON (concluded)

24 //) RETURN
 END

HITS2068
HITS2069
HITS2070

D.29 Subroutine INCONF

```

SUBROUTINE INCONF
IMPLICIT REAL * 8 (A-H, O-Z)

COMMON/DEC/OE(600)
EQUIVALENCE
A(PHIWIN,OE( 3)),(RHO,OE( 1)),(VWIND,OE( 2)),
B(VI,OE( 6)),(CD2,OE( 4)),(CD1,OE( 5)),
C(XN,OE( 9)),(CNA,OE( 7)),(SM,OE( 8)),
D(CMO,OE(12)),(CMPA,OE(10)),(TRIM,OE(11)),
E(PNITRI,OE(15)),(ANG0,OE(13)),(PHIANG,OE(14)),
F(RATE0,OE(18)),(PHIRAT,OE(16)),(GAMO,OE(17)),
G(PHIGAM,OE(21)),(A,OE(19)),(ELL,OE(20)),
H(D,OE(24)),(W,OE(22)),(AIX,OE(23)),
I(AIY,OE(27)),(P,OE(25)),(ALPCON,OE(26)),
J(CAA,OE(30)),(IDUMP,OE(24)),(ICNCL,OE(29)),
EQUIVALENCE
EQUIVALENCE
A(ALPTRM,OE(402)),(ALPHA,OE(400)),(ATOT,OE(401)),
B(BETTRM,OE(405)),(ALPHD0,OE(403)),(B,OE(404)),
C(BETA,OE(408)),(BETAO,OE(406)),(BETAD0,OE(407)),
D(CAYD,OE(411)),(ALPMAX,OE(409)),(ALPMIN,OE(410)),
E(CMHTD,OE(414)),(CD8,OE(412)),(CMA,OE(413)),
F(CAY2,OE(417)),(CMHTA,OE(415)),(CAY1,OE(416)),
G(DELT,OE(420)),(CDA0B,OE(418)),(CDAB,OE(419)),
H(DELX,OE(423)),(DELY,OE(421)),(DELT,OE(422)),
I(DYDYO,OE(426)),(DYDX0,OE(424)),(DZDX0,OE(425)),
J(DELLAM,OE(429)),(EYEP,OE(427)),(DEW,OE(428)),
K(EOLT,OE(432)),(EDLT,OE(430)),(EM,OE(431)),
L(H,OE(435)),(PSID0,OE(433)),(F,OE(434)),
M(PH11,OE(438)),(PHI2,OE(436)),(PHI0,OE(437)),
N(RI,OE(441)),(R2,OE(439)),(PSI0,OE(440)),
O(TCO,OE(444)),(RTRIM,OE(442)),(R3,OE(443)),
P(TP,OE(447)),(TS,OE(445)),(TC,OE(446)),
Q(TOL,OE(450)),(THETA0,OE(448)),(TGO,OE(449)),
R(TOL,OE(453)),(VC,OE(451)),(THETD0,OE(452)),
S(VYA,OE(456)),(VZ,OE(454)),(VCO,OE(455)),
EQUIVALENCE
EQUIVALENCE
A(WI,OE(461)),(WZ,OE(457)),(WX,OE(458)),
B(XLAM0,OE(464)),(W2,OE(459)),(W0,OE(460)),
C(XNU1,OE(467)),(XLAM1,OE(462)),(WL2,OE(463)),
D(XMU,OE(470)),(XNU2,OE(465)),(XLAM2,OE(466)),
E(XJA,OE(473)),(XJAY,OE(468)),(XG,OE(469)),
F(ALPH0,OE(476)),(YA,OE(471)),(XJAZ,OE(472)),
(ZA,OE(474)),(Z,OE(475))

DATA RFAC, G
WX = VWIND
WZ = PHIWIN

V12 = (CD2 - CD1) / (1.00 / V2/V2 - 1.00 / V12)
CAYD = CD1 - CAYD / V12
CD8 = 1.00 + (WZ / (V0 - W)) * 2
B = RHO * A * G
F =

```

D.29 Subroutine INCONF (continued)

```

1000 IF(IDUMP .EQ.1)WRITE(6,1000)WX,WZ,CAYD,CD8,B,F CD8
      FORMAT(// F, WX, WZ, CAYD /)
      1 THETA0 = ANGO/RFAC
        PSIO = PHIANG/RFAC
        THETD0 = RATE0
        PSID0 = PHIRAT
        DYDX0 = GAMO/RFAC
        DZDX0 = PHIGAM/RFAC
          DYDT0 = V0 * DYDX0
          DZDT0 = V0 * DZDX0
          ALPH0 = THETA0 - DYDX0
          BETAO = DZDX0 - PSIO
        IF(IDUMP .EQ.1)WRITE(6,1001)
          1 PSID0, DYDX0, DZDX0, DYDT0, DZDT0
            FCV = F * CNA * V0
            THETA0,PSIO,THETD0,
            ALPHD0 = THETD0 - FCV * ALPH0 - P * BETAO
            IF(ICNCL.EQ.1)ALPHD0=ALPHD0+F*V0*THETA0*(CD8+CAYD/V0/V0)
            BETAD0 = -PSID0 - FCV * BETAO + P * ALPH0
            IF(ICNCL.EQ.1)BETAD0=BETAD0-F*V0*PSIO*(CD8+CAYD/V0/V0)
            CNA = -CNA * ELL * SM / D
            EYEP = 2.00 * AIY /RHO / A / D /V0/V0
            EMP = 2.00 * W / RHO / A / G / V0
            D2V = D / 2.00 /V0
            CMTHYD = CMQ * D2V
            CMTHTA = CMPA * D2V
            H = ZZ(CNA,CD8,CAYD,V0,EMP,ICNCL)
            CCON = CNA / EMP + CMTHD / EYEP
            PII = P * AIX / AIY
            WCON = -4.00 * CNA / EYEP + PII * PII -CCON * CCON
            IF(IDUMP .EQ.1)WRITE(6,1002)ALPH0,BETAO,ALPHD0,BETAD0,CNA,EYEP
            1. EMP, CMTHD, CMTHTA, H
            1 * CCON + 2.00 * P * ( WCON + DSQRT( WCON * WCON + 4.00*(PII
              IF(CMA.LT. 0.) GO TO 3
              PCR=0.00
              XMU=0.00
              RTRIM=1.0
              PHIROL= 0.
              GO TO 4
            3 CONTINUE
              PCR= DSQRT(-CMA*RHO *A /2.00*D/(AIY-AIX))
              PCR= PCR*DBS(V0)
              XMU=( CNA*RHO/4.00 *A/W*G-CMQ*D*D/8.00*RHO*A/(AIY-AIX))/PCR
              PPCR= P/PCR
              PPCR1=1.00- PPCR**2
              XMU2=2.00 *XMU*PPCR
              RTRIM= 1.00 /DSQRT(PPCR1**2 + XMU2**2)
              PHIROL= DATAN(XMU2/PPCR1)
              IF( PPCR1 .LT. 0.) PHIROL= PHIROL+3.141592653589800

```

Line	Code	Text	Line	Code	Text
1	ABAR	= DSQRT(TRIM**2 + PNIITRI**2)	1006	1	BETTRM ,PCR XMU RTRIM
2	PNIITRR	= ZATAN2(PNIITRI,TRIM)	1006	2	FORMAT(/
3	CONTINUE		1007	1	PCR: /
4	ALPTRM	= RTRIM * ABAR / RFAC	1007	2	IP10E12.4 //)
5	PTPR	= PNIITR + PHIROL	1008	1	DELW = P - PII / 2.D0
6	BETTRM	= ALPTRM * DSIN(PTPR)	1008	2	XLAM0 = .SDO* (CMTHTD / EYEP -CNA/EMP)
7	ALPTRM	= ALPTRM * DCOS(PTPR)	1009	1	DELLAM = PII / 4.D0/ W0 *
8	IF(IDUMP	.EQ. 1) WRITE(6,1006) XMU ,RTRIM,PHIROL,ALPTRM	1009	2	W1 = W0 - DELW
9	1 . BETTRM	,PCR XMU RTRIM	1010	1	W2 = W0 + DELW
10	1TRM		1010	2	AA = ALPH0 -ALPTRM
11	1X,	IP10E12.4 //)	1011	1	XLAM1 = XLAM0 + DELLAM
12	DELW	= P - PII / 2.D0	1011	2	XLAM2 = XLAM0 - DELLAM
13	XLAM0	= .SDO* (CMTHTD / EYEP -CNA/EMP)	1012	1	BB = BETAO -BETTRM
14	DELLAM	= PII / 4.D0/ W0 *	1012	2	RCON1 = ALPHD0 + W2 * BB
15	W1	= W0 - DELW	1013	1	RCON2 = BETAD0 - W2 * AA
16	W2	= W0 + DELW	1013	2	WL2 = 2.D0 * (W0 * W0 + DELLAM ** 2)
17	AA	= ALPH0 -ALPTRM	1014	1	R1 = (W0 * RCON1 + DELLAM * RCON1) / WL2
18	XLAM1	= XLAM0 + DELLAM	1014	2	R2 = (-W0 * RCON2 + DELLAM * RCON1) / WL2
19	XLAM2	= XLAM0 - DELLAM	1015	1	IF(IDUMP
20	BB	= BETAO -BETTRM	1015	2	.EQ.1)WRITE(6,1004) W0, DELW, XLAM0 , DELLAM , W1, W2
21	RCON1	= ALPHD0 + W2 * BB	1016	1	1,XLAM1, XLAM2, W0
22	RCON2	= BETAD0 - W2 * AA	1016	2	DELW XLAM1 XLAM2
23	WL2	= 2.D0 * (W0 * W0 + DELLAM ** 2)	1017	1	W2
24	R1	= (W0 * RCON1 + DELLAM * RCON1) / WL2	1017	2	XLAM1
25	R2	= (-W0 * RCON2 + DELLAM * RCON1) / WL2	1018	1	R2 ,/1X, IP11E12.4 //)
26	IF(IDUMP	.EQ.1)WRITE(6,1004) W0, DELW, XLAM0 , DELLAM , W1, W2	1018	2	R4 = AA - R2
27	1,XLAM1, XLAM2, W0		1019	1	R3 = BB -R1
28	DELW XLAM1 XLAM2		1019	2	CAY1 = DSQRT(R1* R1 + R2 * R2)
29	W2		1020	1	CAY2 = DSQRT(R3* R3 + R4 * R4)
30	XLAM1		1020	2	XMU1 = ZATAN2(R2 , R1)
31	XLAM2		1021	1	XMU2 = ZATAN2(R4 , R3)
32	W0		1021	2	PHI1 = DATAN2(-W1 - P, XLAM1)
33	DELW		1022	1	PHI2 = DATAN2(-W2 + P, XLAM2)
34	XLAM1		1022	2	PHI0 = DATAN2(-W0 , XLAM0)
35	XLAM2		1023	1	PSID0
36	W0		1023	2	FORMAT(/
37	DELW		1024	1	ALPH0 EMP
38	XLAM1		1024	2	EYEP
39	XLAM2		1025	1	IP11E12.4 //)
40	W0		1025	2	IF(IDUMP
41	DELW		1026	1	1 . XMU1 , XMU2 , R3
42	XLAM1		1026	2	FORMAT(/
43	XLAM2		1027	1	XMU2
44	W0		1027	2	PHI0
45	DELW		1028	1	PSID0
46	XLAM1		1028	2	FORMAT(/
47	XLAM2		1029	1	ALPH0 EMP
48	W0		1029	2	EYEP
49	DELW		1030	1	IP11E12.4 //)
50	XLAM1		1030	2	IF(IDUMP
51	XLAM2		1031	1	1 . XMU1 , XMU2 , R3
52	W0		1031	2	FORMAT(/
53	DELW		1032	1	XMU2
54	XLAM1		1032	2	PHI0
55	XLAM2		1033	1	PSID0
56	W0		1033	2	FORMAT(/
57	DELW		1034	1	ALPH0 EMP
58	XLAM1		1034	2	EYEP
59	XLAM2		1035	1	IP11E12.4 //)
60	W0		1035	2	IF(IDUMP
61	DELW		1036	1	1 . XMU1 , XMU2 , R3
62	XLAM1		1036	2	FORMAT(/
63	XLAM2		1037	1	XMU2
64	W0		1037	2	PHI0
65	DELW		1038	1	PSID0
66	XLAM1		1038	2	FORMAT(/
67	XLAM2		1039	1	ALPH0 EMP
68	W0		1039	2	EYEP
69	DELW		1040	1	IP11E12.4 //)
70	XLAM1		1040	2	IF(IDUMP
71	XLAM2		1041		

D.29 Subroutine INCONF (concluded)

24 // 1 RETURN
END

HITS2221
HITS2222
HITS2223

D.30 Subroutine INITIL

```

SUBROUTINE INITIL
IMPLICIT REAL*8 (A-H,O-Y)
COMMON /CARDIM/ B(200), C(5000), IA(200,2), ID(10,2), INC,
1 IOPRNT, IOUT, IY, KARS(3), KA, KB, KC, KD, KVI, K234, LABC(3)
COMMON /CISPRN/ ISPRNT
COMMON /CMCPRN/ MCPRNT
COMMON /CRAN/ IRANNO
COMMON /CMCH/ ID1(200), ID2(200), I11(400), I12(400), IDT(10), IIT(20),
1 IRI(10), IR2(10), NTRIAL, MCALC, MCOPT, IRJ1, IRJ2, IN8, MCC, NR1, NR2
COMMON /HISPLT/ IOPLOT, ISPLOTT, ZTITLE(20,30), ZDATAD(21,4,30)
DIMENSION IB(2,200), IOEA(160), IOEB(150), IOEC(150), IOE(460),
1 KOE(3)
EQUIVALENCE (IOE(1), IOEA(1)), (IOE(161), IOEB(1)),
1 (IOE(311), IOEC(1)), (IB(1,1), B(1))
DATA IOEA /
1. 2. 3. 4. 5. 6. 7.
8. 9. 10. 11. 12. 13. 14. 15. 16.
17. 18. 19. 20. 21. 22. 23. 24. 25.
26. 27. 28. 29. 30. 101. 102. 103. 104.
105. 106. 107. 108. 109. 110. 111. 112. 113.
114. 115. 116. 117. 118. 119. 120. 121. 122.
123. 124. 125. 126. 127. 128. 129. 130. 131.
132. 133. 134. 135. -201. -202. -204. -205. -206/
DATA KOE / 70, 150, 150 /
DO 10 J=1,2
DO 10 I=1,200
10 IA(I,J) = 0
DO 20 I=1,200
20 B(I) = 0.000
C FIRST CARD IN TELLS WHICH OF THE 3 IOE DATA STATEMENTS (IY) WILL BE
C PLACED INTO THE FIRST COLUMN OF IA.
READ (INC,9001,END=70) IY, IOPRNT, ISPRNT, MCPRNT, MCOPT, MCALC, NCELL,
1 NTRIAL, IRJ1, IRJ2, MCPRNT, IRANNO, IOPLOT, ISPLOTT
3, IOPLOT, ISPLOTT
9001 FORMAT(11, 213,1015)
C
C INPUT LISTING
WRITE(IOUT,5000)
5000 FORMAT(1H1, 43X, '*** SIMULATION INPUT SUMMARY ***'////)
WRITE(IOUT,5001) IY, IOPRNT, ISPRNT, MCPRNT, MCALC, NCELL, NTRIAL,
AIRJ1, IRJ2, MCPRNT, IRANNO, IOPLOT, ISPLOTT
5001 FORMAT(30X,
1 'SINGLE-CASE, RANGE-CHECK, AND STATISTICAL PROCESSOR CON
ATROLS',////
4 '35X, ' IY = ',4X,11,5X, ' IOPRNT = ',2X,13,5X, ' ISPRNT =
2,15,///,50X,
B 'MONTE CARLO CONTROLS',//,25X, ' MCOPT = ',15,6X, ' MCALC
C ',15,5X, ' NCELL = ',15,5X, ' NTRIAL = ',15,
3 '//,24X, ' IRJ1 = ',15,5X,
D ' IRJ2 = ',15,5X, ' MCPRNT = ',15,5X, ' IRANNO = ',15,
E '//,24X, ' IOPLOT = ',15,5X, ' ISPLOTT = ',15,///)

```

D.30 Subroutine INITIL (concluded)

```

5002 WRITE(IOUT,5002)
      A10X,6X,'CODE',6X,4X,'VARIABLE',4X,5X,'NOMINAL',4X,4X,'TOLERANCE',
      B3X,4X,'STANDARD',6X,3X,'SUBSEQUENT',2X,
      C10X,5X,'NUMBER',5X,6X,'TYPE',6X,6X,'VALUE',5X,1X,'4X,'DEVIATION',
      D5X,5X,'POINTS',5X,7X,
      CALL RAND ( IRRAND )
      DO 25 I=1,20
        25 X = RANDOM ( Y )
      C SINCE THE ENTITIES IN THE DATA STATEMENTS ARE PACKED TIGHT, IADD WILL
      C INDICATE HOW MANY TO ADD ON TO THE LOOP COUNTER (IN ORDER TO PICK
      C UP THE CORRECT IOE).
        IADD = 0
        IF ( IY .GT. 1 ) IADD = 160
        IF ( IY .GT. 2 ) IADD = 310
      C KA IS THE COUNT OF THE INDEPENDENT VARIABLES IN THE FIRST COLUMN OF
        KA = KOE(IY)
        DO 30 I=1,KA
          30 IAI(I,1) = IOE(I+IADD)
          GO TO (40, 50, 60), IY
          40 CALL TG2987
            RETURN
          50 CALL TG2440
            RETURN
          60 CALL TG1795
            RETURN
          70 WRITE(IOUT,9002)
          9002 FORMAT(IH1,59(1X,6('THATS ALL FOLKS',5X)))
            CALL EXIT
            RETURN
            END

```

HITS2274
 HITS2275
 HITS2276
 HITS2277
 HITS2278
 HITS2279
 HITS2280
 HITS2281
 HITS2282
 HITS2283
 HITS2284
 HITS2285
 HITS2286
 HITS2287
 HITS2288
 HITS2289
 HITS2290
 HITS2291
 HITS2292
 HITS2293
 HITS2294
 HITS2295
 HITS2296
 HITS2297
 HITS2298
 HITS2299
 HITS2300
 HITS2301
 HITS2302
 HITS2303
 HITS2304

D.31 Subroutine INLHST

```

SUBROUTINE INLHST
  IMPLICIT REAL*8 (A-H,O-Z)
  COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),
  IIR1(1),IIR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2
  THIS SUBROUTINE ZEROS OUT THE HISTOGRAM AND MONTE CARLO COUNTERS
  NCC=0
  NR1=0
  NR2=0
  IN8=0
  DO 1 I=1,10
    IIR1(I)=0
    IIR2(I)=0
  DO 2 I=1,200
    ID1(I)=0
    ID2(I)=0
  DO 3 I=1,400
    II1(I)=0
    II2(I)=0
  RETURN
END

```

1

2

3

C

HITS2305

HITS2306

HITS2307

HITS2308

HITS2309

HITS2310

HITS2311

HITS2312

HITS2313

HITS2314

HITS2315

HITS2316

HITS2317

HITS2318

HITS2319

HITS2320

HITS2321

HITS2322

HITS2323

HITS2324

D.32 Subroutine LOADER

```

SUBROUTINE LOADER (IV, CM, CV, FCTR, NRR, PM, PV, PT, NCELL
1  , A, IPASS)
2  IMPLICIT REAL * 8 (A-H, O-Y)
3  DIMENSION A(20, 5)
4  , IB(2,200)
5  EQUIVALENCE (IB(1,1), B(1) )
6  COMMON/ HISPLT/ IOPLT , ZTITLE(20,30) , ZOATAD(21,4,30)
7  COMMON /CARDCH / B(200) , C(5000) , IA(200,2) , ID(10,2) , INC
8  , IOPRNT , IOUT , IV , KARS(3) , KA , KB , KC , KD , KTI , K234
9  , LABC(3)
10 DATA PI / 3.141592654D0 /
11 DATA J / 0 /
12 J = J+1
13 IF( J.GT. KD+K234) J = 1
14 GO TO ( 1 , 2 , 3 , 4 ) , IPASS
15 JSTRT = 1
16 JD = 0
17 J2 = 2
18 JEND = KD
19 JS = 4
20 GO TO 5
21 JEND = KD + K234
22 JSTRT = 1 + KD
23 JS = 4
24 J2 = 2
25 JD = 0
26 GO TO 5
27 JSTRT = 1
28 JEND = KD
29 JS = 11
30 J2 = 4
31 JD = 10
32 GO TO 5
33 JSTRT = 1 + KD
34 JEND = KD + K234
35 JS = 11
36 J2 = 3
37 GO TO 5
38 FPI = 1.00 / DSORT( 2.D0 * PI )
39 NCELL1 = NCELL + 1
40 IF( J.LE. KD ) GO TO 10
41 ITYPE = IB(1,IA(J-KD,2))
42 ZTITLE(20, J) = 0.
43 IF( ITYPE .EQ. 2 ) ZTITLE(20,J) = 1.
44 GO TO 11
45 ITYPE = 7
46 IF( ID(J,2) .NE. 0 ) ITYPE = 8
47 ZTITLE(1 , J) = IPASS
48 ZTITLE(2 , J) = IV

```

HITS2325
HITS2326
HITS2327
HITS2328
HITS2329
HITS2330
HITS2331
HITS2332
HITS2333
HITS2334
HITS2335
HITS2336
HITS2337
HITS2338
HITS2339
HITS2340
HITS2341
HITS2342
HITS2343
HITS2344
HITS2345
HITS2346
HITS2347
HITS2348
HITS2349
HITS2350
HITS2351
HITS2352
HITS2353
HITS2354
HITS2355
HITS2356
HITS2357
HITS2358
HITS2359
HITS2360
HITS2361
HITS2362
HITS2363
HITS2364
HITS2365
HITS2366
HITS2367
HITS2368
HITS2369
HITS2370
HITS2371
HITS2372
HITS2373
HITS2374

D.32 Subroutine LOADER (continued)

```

ZTITLE(3 ,J)= ITYPE
ZTITLE(JS ,J)= NRR
ZTITLE(JS+1,J)= CM
ZTITLE(JS+2,J)= CV
ZTITLE(JS+3,J)= PM
ZTITLE(JS+4,J)= PV
ZTITLE(JS+5,J)= FCTR
ZTITLE(JS+6,J)= PT
ZTITLE(18 ,J)= NCELL
SUM = 0.00
DO 7 IN = 1, NCELL
SUM = SUM + A(IN,4)
SUM = 100.00 / SUM
SAVE = -1.020
DO 8 IN = 1, NCELL
ZDATAD(IN, J2, J) = A(IN,4) * SUM
IF( SAVE .LT. ZDATAD(IN,J2,J) ) SAVE = ZDATAD(IN,J2, J)
IF( IPASS .EQ. 4 ) GO TO 8
IF( IPASS .EQ. 2 ) GO TO 9
ZDATAD(IN, IPASS , J) = A(IN,2)
GO TO 8
9 ZDATAD(IN, 1 , J) = A(IN , 2 )
IF( ITYPE .NE. 2 ) GO TO 20
F = DEXP( - (A(IN,1) - ZTITLE(7,J) )**2 /2.
1 /ZTITLE(8,J) ) * FPI / SORT( ZTITLE(8,J) )
ZTITLE(20,J) = 1
GO TO 21
20 IF( ITYPE .NE. 3 ) GO TO 22
ZTITLE(20,J) = 0
F = 1.00 / ( ZDATAD( NCELL +1 ,1, J) -ZDATAD(1,1,J) )
GO TO 21
22 IF( ITYPE .NE. 4 ) GO TO 23
KB = IA(J-KD,2)
ITL = IB(1, KB + 4)
KC = IB(1, KB + 5)
DATA= A(IN,1)
CALL ARTLU(1, DATA , C(KC +ITL), F, C(KC) )
ZDATAD(IN,4,J) = F * ZTITLE(9, J) *100.00
21 CONTINUE
23 CONTINUE
6 CONTINUE
IF( IPASS .NE. 2 ) GO TO 25
ZDATAD(NCELL,1,J) = A(NCELL,3)
ZDATAD(NCELL,2,J) = 0.00
27
C
C
C
GO TO 26
25 IF( IPASS .EQ. 1 ) GO TO 27
IF( IPASS .EQ. 4 ) GO TO 28
C IPASS = 3

```

HITS2375
HITS2376
HITS2377
HITS2378
HITS2379
HITS2380
HITS2381
HITS2382
HITS2383
HITS2384
HITS2385
HITS2386
HITS2387
HITS2388
HITS2389
HITS2390
HITS2391
HITS2392
HITS2393
HITS2394
HITS2395
HITS2396
HITS2397
HITS2398
HITS2399
HITS2400
HITS2401
HITS2402
HITS2403
HITS2404
HITS2405
HITS2406
HITS2407
HITS2408
HITS2409
HITS2410
HITS2411
HITS2412
HITS2413
HITS2414
HITS2415
HITS2416
HITS2417
HITS2418
HITS2419
HITS2420
HITS2421
HITS2422
HITS2423
HITS2424

D.32 Subroutine LOADER (concluded)

HITS2425
HITS2426
HITS2427
HITS2428
HITS2429
HITS2430
HITS2431
HITS2432
HITS2433
HITS2434
HITS2435
HITS2436
HITS2437
HITS2438

```

28      ZDATAD(NCELL1, 3, J) = A(NCELL, 3)
26      ZDATAD(NCELL1, 4, J) = 0.D0
        GO TO 26
        ZDATAD (NCELL1, 3, J) = 0.D0
        CONTINUE
        IF ( IPASS .EQ. 1 .OR. IPASS .EQ. 2 ) GO TO 14
        IF( ZTITLE(19,J) .LT. SAVE ) ZTITLE(19,J) = SAVE
        GO TO 15
        ZTITLE(19,J) = SAVE
14      CONTINUE
15      IF( IPASS .EQ. 4 )
C        CALL HISPLT
C        RETURN
        END

```

```

SUBROUTINE MCRL
IMPLICIT REAL*8 (A-H,O-Y)
COMMON/CARDCH/ B(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT,
1IY,KARSI(3),KA,KB,KC,KD,KTI,K234,LABC(3)
COMMON /CMCPRN/ MCPRNT
COMMON/IMCH/ID1(200),ID2(200),II1(400),II2(400),IDT(10),IIT(20),
1IRI(10),IR2(10),NTRIAL,NCELL,M(ALC,MCOPT,IRJ1,IRJ2,IN8,MCC,NRI,NR2)
COMMON/HISPLY/IOPLOT,ISPLOT,ZTITLE(3120)
COMMON/OEC/OE(600)
KCASE = 0
DO 400 IBB = 1, 3120
ZTITLE(IBB) = 0.0
CALL INLHST
CALL CHCKIN
IF(MCPRNT.EQ.1) WRITE(IOUT,5000)
5000 FORMAT('HI,39X,*** SUMMARY OF MONTE CARLO RANDOM EXPERIMENTS ***',
A,////')
C      START MONTE CARLO LOOP WITH SAMPLE GENERATOR
C      ALL SAMPLE
C      RUN CASE
KCASE = KCASE + 1
IF ( MCALC ) 20,20,25
20 CALL PICK1
GO TO 22
25 CALL CRVFT (MCALC)
22 IF ( MCPRNT ) 24,24,23
23 CONTINUE
C USE THE LAST TEN ENTRIES IN THE C ARRAY FOR OUTPUT.
X = 0
231 JC = 4990
232 JC = K + 1
IF ( K.GT. K234 ) GO TO 233
JA = IA(K,1)
JC = JC + 1
C(JC) = OE(JA)
IF ( JC.LT. 5000 ) GO TO 232
233 WRITE (IOUT,9002)KCASE, (C(I), I=4991,JC)
9002 FORMAT('IX,CASE',I5,'IX,IND. VAR.',1P10E12.4)
IF ( K.LT. K234 ) GO TO 231
K = 0
JC = 4990
234 JC = K + 1
IF ( K.GT. KD ) GO TO 235
JD = ID(K,1)
JC = JC + 1
C(JC) = OE(JD)
GO TO 234
235 WRITE (IOUT,9003) (C(I), I=4991,JC)
9003 FORMAT('IX,DEP. VAR.',1P10E12.4,////)
24 IER = 0

```

D.33 Subroutine MCRL (concluded)

```

C      PERFORM INTERVAL CHECK
      IF=0
      C      4 HSTG1(IER)
      IF(IER)2,3,3
      NR1=NR1+1
      IF(NR1-IRJ1)3,3,7
      IF(IN8)6,6,4
      IF THERE ARE TYPE 8,S PERFORM CHECK ON USER SPECS
      C      4 IER=0
      CALL HSTG2(IER)
      IF(IER)5,6,6
      NR2=NR2+1
      IF(NR2-IRJ2)6,6,7
      C      5 INCREMENT SAMPLE COUNTER AND CHECK - - END OF MONTE CARLO LOOP
      NCC=NCC+1
      IF(NCC-NTRIAL)1,7,7
      IF(IN8)8,8,9
      C      6 KEND=2
      GO TO 10
      9 KEND=4
      C      START PRINTING RESULTS
      10 WRITE(IOUT,200) NTRIAL,NR1,NR2
      200 FORMAT(1H1,2X,'MONTE CARLO RESULTS',10X,'SAMPLE ='',15,4X,'INTERNAL
      1 REJECTS ='',14,4X,'USER REJECTS ='',14)
      DO 21 L=1,KEND
      GO TO(12,13,14,15),L
      12 WRITE(IOUT,201)
      201 FORMAT( // 20X, 'DEPENDENT VARIABLES WITH INTERNAL RANGES'
      13 GO TO 11
      13 WRITE(IOUT,202)
      202 FORMAT(//20X,'INDEPENDENT VARIABLES WITH INTERNAL RANGES')
      GO TO 11
      14 WRITE(IOUT,203)
      203 FORMAT(//20X,'DEPENDENT VARIABLES WITH USERS RANGES')
      GO TO 11
      15 WRITE(IOUT,204)
      204 FORMAT(//20X,'SORTED INDEPENDENT VARIABLES FOR USERS RANGES')
      11 CALL PRINTMC(L)
      21 CONTINUE
      RETURN
      END
HITS2489
HITS2490
HITS2491
HITS2492
HITS2493
HITS2494
HITS2495
HITS2496
HITS2497
HITS2498
HITS2499
HITS2500
HITS2501
HITS2502
HITS2503
HITS2504
HITS2505
HITS2506
HITS2507
HITS2508
HITS2509
HITS2510
HITS2511
HITS2512
HITS2513
HITS2514
HITS2515
HITS2516
HITS2517
HITS2518
HITS2519
HITS2520
HITS2521
HITS2522
HITS2523
HITS2524
HITS2525
HITS2526
HITS2527
HITS2528
HITS2529

```

.34 Subroutine MOVEUP

```

SUBROUTINE MOVEUP
  IMPLICIT REAL*8 (A-H,O-Z)
  COMMON /CARDCM/ B(200), C(5000), IA(200,2), ID(10,2), NC,
  1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)
  DIMENSION IB(2,200)
  EQUIVALENCE (IB(1,1), B(1))
  C IF THERE WERE ANY TYPE 1, MOVE THEM UP TO THE START OF THE IA ARRAY.
  IF ( KT1 .EQ. 0 ) GO TO 20
  KTIR = 0
  10 KTIR = KTIR + 1
  J1 = KARS1(KTIR)
  JS1 = IA(KTIR,1)
  JS2 = IA(KTIR,2)
  IA(KTIR,1) = IA(J1,1)
  IA(KTIR,2) = IA(J1,2)
  IA(J1,1) = JS1
  IA(J1,2) = JS2
  IF ( KTIR .LT. KT1 ) GO TO 10
  GO TO 50
  20 CONTINUE
  IF ( K234 .EQ. 0 ) GO TO 50
  C RUN THRU IA. GET SUBSCRIPT OF B. IF TYPE 2, 3, OR 4, MOVE UP IN IA.
  K234R = 0
  30 IAR = 0
  IAR = IAR + 1
  IF ( IA(IAR,2) .EQ. 0 .OR. IA(IAR,2) .EQ. -1 ) GO TO 40
  JB = IA(IAR,2)
  IF ( IB(1,JB) .EQ. 1 .OR. IB(1,JB) .GT. 4 ) GO TO 40
  K234R = K234R + 1
  JS1 = IA(K234R,1)
  JS2 = IA(K234R,2)
  IA(K234R,1) = IA(IAR,1)
  IA(K234R,2) = IA(IAR,2)
  IA(IAR,1) = JS1
  IA(IAR,2) = JS2
  40 IF ( K234R .EQ. K234 ) GO TO 50
  IF ( IAR .LT. KA ) GO TO 30
  WRITE (IOUT,9006)
  9006 FORMAT(71H MOVEUP ERROR ... REACHED END OF IA BEFORE FINDING ALL
  1 TYPE 2, 3 AND 4.)
  CALL EXIT
  50 CONTINUE
  RETURN
  END
HITS2530
HITS2531
HITS2532
HITS2533
HITS2534
HITS2535
HITS2536
HITS2537
HITS2538
HITS2539
HITS2540
HITS2541
HITS2542
HITS2543
HITS2544
HITS2545
HITS2546
HITS2547
HITS2548
HITS2549
HITS2550
HITS2551
HITS2552
HITS2553
HITS2554
HITS2555
HITS2556
HITS2557
HITS2558
HITS2559
HITS2560
HITS2561
HITS2562
HITS2563
HITS2564
HITS2565
HITS2566
HITS2567
HITS2568
HITS2569
HITS2570
HITS2571
HITS2572
HITS2573

```

D.35 Function NCOV

```

FUNCTION NCOV(I1, JJ)
IMPLICIT REAL*8(A-H, O-Z)
COMMON /CARDCM/ 3(200), C(5000), IA(200,2), ID(10,2), INC,
1 IOPRNT, IOUT
COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUT, KCOV
IF(I1.NE.JJ) GO TO 100
WRITE(IOUT,99) I1, JJ
FORMAT( //, : D0234 TRIES TO LOOK UP VARIANCE IN COVARIANCE ARRAY
99 A I = , I3, : J = , I3)
100 CONTINUE
IF ( JJ.LT. I1) GO TO 1
I= I1
J= JJ
GO TO 2
I= JJ
J= I1
CONTINUE
NSUM= 0
I1= I-1
IF( I1.EQ. 0) GO TO 3
DO 4 IK= 1, I1
NSUM= NSUM + K234 -IK
CONTINUE
NCOV= KCOV + NSUM +J -I
IF(NCOV.LE.2000) RETURN
WRITE(IOUT,98) NCOV
FORMAT( //, : COVARIANCE MATRIX EXCEEDS THE C ARRAY. NCOV = .,
98 A I4)
RETURN
END

```

HITS2574
HITS2575
HITS2576
HITS2577
HITS2578
HITS2579
HITS2580
HITS2581
HITS2582
HITS2583
HITS2584
HITS2585
HITS2586
HITS2587
HITS2588
HITS2589
HITS2590
HITS2591
HITS2592
HITS2593
HITS2594
HITS2595
HITS2596
HITS2597
HITS2598
HITS2599
HITS2600
HITS2601
HITS2602
HITS2603

D.36 Function NMEAN

```

FUNCTION NMEAN(I)
IMPLICIT REAL*8(A-H, O-Z)
COMMON /CARDCH/ B(200), C(5000), IA(200,2)
COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV
NMEAN= KCSUMS+I
RETURN
END

```

```

HITS2604
HITS2605
HITS2606
HITS2607
HITS2608
HITS2609
HITS2610

```

D.37 Function NVAR

HITS2611
HITS2612
HITS2613
HITS2614
HITS2615
HITS2616
HITS2617

```

FUNCTION NVAR(I)
IMPLICIT REAL*8(A-H, O-Z)
COMMON /CARDCM/ B(200), C(5000), IA(200,2)
COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDQUB, KCOV
NVAR= KCSUMS +KD +I
RETURN
END

```


D.38 Function NVARX

```

FUNCTION NVARX(I)
IMPLICIT REAL*8(A-H, O-Z)
COMMON /CARDCM/ B(200), C(5000), IA(200,2)
COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV
NVARX= IA(I,2) +3
RETURN
END

```

HITS2618
HITS2619
HITS2620
HITS2621
HITS2622
HITS2623
HITS2624

D.39 Function NLD

```

      FUNCTION NLD(I,J)
      IMPLICIT REAL*8(A-H, O-Z)
      COMMON /CARDCM/ B(200), C(5000), IA(200,2)
      COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV
      NID= KCYBAR+ (2*J-1)*KD +I
      RETURN
      END

```

```

HITS2625
HITS2626
HITS2627
HITS2628
HITS2629
HITS2630
HITS2631

```

D.40 Function N2D

```

1      FUNCTION N2D(I, JJ, KK)
2      IMPLICIT REAL*8(A-H, O-Z)
3      COMMON /CARDCH/ R(201), C(5000), IA(200,2)
4      COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUB, KCOV
5      IF( KK.LT. JJ) GO TO 1
        J= JJ
        K= KK
        GO TO 2
1      J= KK
        K= JJ
2      CONTINUE
        IF( J.NE. K) GO TO 3
        N2D= KCYBAR + 2* J *KD +I
        RETURN
3      CONTINUE
        J1= J-1
        NSUM=0
        IF( J1.EQ. 0) GO TO 5
        DO 4 JK= 1, J1
            NSUM= NSUM + K234 -JK
        CONTINUE
        N2D= KCDOUB + KD *(NSUM +K -J-1) +I
        RETURN
5      END

```

HTS2632
 HTS2633
 HTS2634
 HTS2635
 HTS2636
 HTS2637
 HTS2638
 HTS2639
 HTS2640
 HTS2641
 HTS2642
 HTS2643
 HTS2644
 HTS2645
 HTS2646
 HTS2647
 HTS2648
 HTS2649
 HTS2650
 HTS2651
 HTS2652
 HTS2653
 HTS2654
 HTS2655

D.41 Subroutine PICK1

```

SUBROUTINE PICK1
  IMPLICIT REAL*8 (A-H,O-Z)
  COMMON /CARDCM/ R(200), C(500), IA(200,2), ID(10,2), INC,
1  IOPRNT, IOUT, IY, KARSI(3), KB, KC, KD, KTI, K234, LABC(3)
  GO TO (1, 2, 3), IY
1  CALL G2987
   RETURN
2  CALL G2440
   RETURN
3  CALL G1795
   RETURN
END
HITS2656
HITS2657
HITS2658
HITS2659
HITS2660
HITS2661
HITS2662
HITS2663
HITS2664
HITS2665
HITS2666
HITS2667

```

D.42 Subroutine PRINTMC

```

SUBROUTINE PRINTMC(IPASS)
  IMPLICIT REAL*8 (A-H,O-Y)
  COMMON/CARDCM/ B(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT,
  IIV,KARSI(3),KA,KB,KC,KD,KI,K234,LABC(3)
  COMMON/IMCH/ID1(200),ID2(200),I11(400),I12(400),IOT(10),IIT(20),
  IIR1(10),IR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,INB,MCC,NR,NR2
  DIMENSION A(20,5)
  COMMON/HISPLT/ IOPLOT,ISPLT,ZITITLE(20,30),ZDATAD(21,4,30)
  EQUIVALENCE (A(1,1),C(4900))
  EQUIVALENCE (KADD,KC)
  THIS ROUTINE NEEDS THE KADD ADDRESS OF THE NOMINAL CASE RESULTS
  IBN=0
  ENCL=NCELL
  GO TO(1,2,1,2),IPASS
  KE=KD
  LADD=2*KD+K234+KD+(KADD-1)
  GO TO 3
  KE=K234
  DO 17 I=1,KE
  GO TO(4,5,6,5),IPASS
  FIRST PASS FOR THE DEPENDENT VARIABLES
  DEFINITIONS
    PM - PRIOR MEAN VALUE
    PT - PRIOR TOLERANCE
    IV - VARIABLE CODE NO
    FCTR - CLASS INTERVAL
    NRR - NUMBER OF REJECTS
  IV=ID(I,1)
  IMN=LADD+I
  IVAR=IMN+KD
  PV=C(IVAR)
  PT=3.*DSQRT(PV)
  PM=C(IMN)
  FCTR = 2.0D0 * PT / ENCL
  NRR=IR1(I)
  GO TO 9
  FIRST AND THIRD PAS FOR INDEPENDENT VARIABLES
  IV=IA(I,1)
  L=IA(I,2)
  NRR=0
  PH=B(L+1)
  PT=B(L+2)
  PV=B(L+3)
  FCTR = 2.0D0 * PT / ENCL
  GO TO 9
  FOURTH PASS FOR USER SPEC DEP. VARIABLES
  L=ID(I,2)
  CHECK IF IT IS A TYPE 8
  IF(L)4,4,7
  IV=ID(I,1)
  NRR=IR2(I)
  GO TO 8
  CONVERT AND TRANSFER HISTOGRAMS TO OUTPUT ARRAY A

```

HITS2668
 HITS2669
 HITS2670
 HITS2671
 HITS2672
 HITS2673
 HITS2674
 HITS2675
 HITS2676
 HITS2677
 HITS2678
 HITS2679
 HITS2680
 HITS2681
 HITS2682
 HITS2683
 HITS2684
 HITS2685
 HITS2686
 HITS2687
 HITS2688
 HITS2689
 HITS2690
 HITS2691
 HITS2692
 HITS2693
 HITS2694
 HITS2695
 HITS2696
 HITS2697
 HITS2698
 HITS2699
 HITS2700
 HITS2701
 HITS2702
 HITS2703
 HITS2704
 HITS2705
 HITS2706
 HITS2707
 HITS2708
 HITS2709
 HITS2710
 HITS2711
 HITS2712
 HITS2713
 HITS2714
 HITS2715
 HITS2716
 HITS2717

D.42 Subroutine PRNTMC (concluded)

```

9      DO 14 J=1,NCELL
10     GO TO(10,11,12,13),IPASS
11     A(J,4)=ID1(IBN+J)
12     GO TO 14
13     A(J,4)=I11(IBN+J)
14     GO TO 14
15     A(J,4)=ID2(IBN+J)
16     GO TO 14
17     A(J,4)=I12(IBN+J)
18     GO TO 14
19     CONTINUE
20     FILL IN THE OUTPUT ARRAY AND CALC MEAN AND VARIANCE
21     CALL FILL(A,FCTR,PM,PT,CM,CV,NCELL)
22     IF(IOPLOT.NE.0)
23     1CALL LOADER(IV,CM,CV,FCTR,NRR,PM,PV,PT,NCELL,A,IPASS)
24     WRITE(IOUT,200) IV,CM,CV,FCTR,NRR,PM,PV,PT
25     FORMAT(//2X,'CODE NUMBER =',I5.4X,'MEAN =',E14.6,'VAR =',E14.6,
26     1,4X,'INTERVAL =',E14.6/2X,'NO. REJECTS =',I5.2X,'(MEAN) =',E14.6,
27     2,4X,'(VAR) =',E14.6/2X,'TOLERANCE =',E14.6/4X,'CLASS MARK =',3X,
28     3,'LOWER BOUND =',3X,'UPPER BOUND =',3X,'FREQ =',3X,'DIST. FUNCT. =')
29     DO 16 K=1,NCELL
30     16 WRITE(IOUT,201) (A(K,L),L=1,5)
31     201 FORMAT(3E14.6,F5.0,E14.6)
32     17 IBN=IBN+NCELL
33     RETURN
34     END

```

HITS2718
HITS2719
HITS2720
HITS2721
HITS2722
HITS2723
HITS2724
HITS2725
HITS2726
HITS2727
HITS2728
HITS2729
HITS2730
HITS2731
HITS2732
HITS2733
HITS2734
HITS2735
HITS2736
HITS2737
HITS2738
HITS2739
HITS2740
HITS2741
HITS2742

D.43 Subroutine QCOR

```

SUBROUTINE QCOR
  IMPLICIT REAL*8(A-H, O-Z)
  COMMON /CARDIM/ B(200), C(5000), IA(200,2)
  COMMON /INDEX/ KD, K234, KCYBAR, KCSUMS, KCDOUTB, KCOV
  K1 = K234 - 1
  IF( K1.EQ. 0) RETURN
  DO 1 I = 1, KD
    DEL = 0.00
    DELP = 0.00
    DO 2 J = 1, K1
      NXJ = NID(I,J)
      YXJ = C( NXJ)
      KB = J + 1
      DO 2 K = KB, K234
        N2 = N2D( I,J,K)
        NK = NID(I,K)
        NC = NCOV(J,K)
        CV = C( NC)
        DEL = DEL + C( N2) * CV
        DELP = DELP + YXJ * C( NK) * CV
      2
    1
    NY = NMEAN(I)
    C(NY) = C(NY) + DEL
    NV = NVAR(I)
    C(NV) = C(NV) + DELP
  RETURN
  END

```

HITS2743
 HITS2744
 HITS2745
 HITS2746
 HITS2747
 HITS2748
 HITS2749
 HITS2750
 HITS2751
 HITS2752
 HITS2753
 HITS2754
 HITS2755
 HITS2756
 HITS2757
 HITS2758
 HITS2759
 HITS2760
 HITS2761
 HITS2762
 HITS2763
 HITS2764
 HITS2765
 HITS2766
 HITS2767
 HITS2768
 HITS2769

D.44 Function RANDOM

```

FUNCTION RANDOM(X)
  IMPLICIT REAL*8(A-H,O-Z)
  CALL RANDU(IX,IY,X)
  IX= IY
  RANDOM=X
  RETURN
ENTRY RAND(IX)
  RETURN
END

```

```

HITS2770
HITS2771
HITS2772
HITS2773
HITS2774
HITS2775
HITS2776
HITS2777
HITS2778

```


D.45 Subroutine RANDU

```

5
6
SUBROUTINE RANDU(IY,YFL)
  IMPLICIT REAL*8(A-H,O-Z)
  IY= IX *65539
  IF( IY) 5.6.6
  IY= IY+2147483647 + 1
  YFL= IY
  YFL= YFL*.4656613D-9
  RETURN
END

```

```

HITS2779
HITS2780
HITS2781
HITS2782
HITS2783
HITS2784
HITS2785
HITS2786
HITS2787

```

```

SUBROUTINE RW2987
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/OEC/DE(500)
EQUIVALENCE
  A(EPHMIN),DE(103), (VQ)
  B(VI),DE(106), (RHO)
  C(XN),DE(109), (CD2)
  D(CMQ),DE(112), (CNA)
  E(CNITP),DE(115), (CMPA)
  F(RATE),DE(118), (ANSO)
  G(PHIGAM),DE(121), (PHIR,T)
  H(D),DE(124), (A)
  I(CAIY),DE(127), (W)
  J(CAA),DE(130), (P)
EQUIVALENCE
  A(ALPHA),DE(402), (ALPHA)
  B(BETAT),DE(405), (ALPHD)
  C(BETA),DE(408), (BETAD)
  D(CAYD),DE(411), (ALPMAX)
  E(CMTHD),DE(414), (CDB)
  F(CAY2),DE(417), (CMHTA)
  G(DELT),DE(420), (CDAOB)
  H(DELX),DE(423), (DELV)
  I(DYDT),DE(426), (DYDX)
  J(DELLAM),DE(429), (DZDTC)
  K(ELT),DE(432), (EYEP)
  L(CH),DE(435), (EDLT)
  M(PHI1),DE(438), (PSID)
  N(PRI),DE(441), (PHI2)
  O(R4),DE(444), (R2)
  P(TCO),DE(447), (RTRIM)
  Q(TP),DE(450), (TS)
  R(TOL),DE(453), (THETA)
  S(VYA),DE(456), (VC)
EQUIVALENCE
  A(W1),DE(461), (VZA)
  B(XLAM),DE(464), (WZ)
  C(XNU1),DE(467), (W2)
  D(XMU),DE(470), (XLAM1)
  E(XJA),DE(473), (XNU2)
  F(ALPH),DE(476), (XJAY)
EQUIVALENCE
  A(YTFC),DE(502), (TN)
  B(VYTFC),DE(505), (ZTFC)
EQUIVALENCE
  A(ZT),DE(509), (XT)
  B(VZT),DE(512), (VXT)
  C(VXR),DE(515), (YR)
  D(TX INCON),DE(518), (VYR)
  (VWIND)
  (CDI)
  (V2)
  (SM)
  (TRIM)
  (PHIANG)
  (GAMO)
  (ELL)
  (ATK)
  (ALPCGN)
  (ATOT)
  (B)
  (BETADO)
  (ALPMIN)
  (CMA)
  (CAY1)
  (CDAB)
  (DELT)
  (DZDX)
  (DEMP)
  (F)
  (PHIO)
  (PSIO)
  (R3)
  (TC)
  (TG)
  (THETD)
  (VC)
  (WX)
  (WO)
  (WL2)
  (XLAM2)
  (XG)
  (XJAZ)
  (ZA)
  (XTFC)
  (VXTFC)
  (YT)
  (VYT)
  (ZR)
  (VZR)
  (DE(101))
  (DE(104))
  (DE(107))
  (DE(110))
  (DE(113))
  (DE(116))
  (DE(119))
  (DE(122))
  (DE(125))
  (DE(128))
  (DE(403))
  (DE(406))
  (DE(409))
  (DE(412))
  (DE(415))
  (DE(418))
  (DE(421))
  (DE(424))
  (DE(427))
  (DE(430))
  (DE(433))
  (DE(436))
  (DE(439))
  (DE(442))
  (DE(445))
  (DE(448))
  (DE(451))
  (DE(454))
  (DE(457))
  (DE(459))
  (DE(462))
  (DE(465))
  (DE(468))
  (DE(471))
  (DE(474))
  (DE(503))
  (DE(506))
  (DE(507))
  (DE(510))
  (DE(513))
  (DE(516))
  (DE(102))
  (DE(105))
  (DE(108))
  (DE(111))
  (DE(114))
  (DE(117))
  (DE(120))
  (DE(123))
  (DE(126))
  (DE(401))
  (DE(404))
  (DE(407))
  (DE(410))
  (DE(413))
  (DE(416))
  (DE(419))
  (DE(422))
  (DE(425))
  (DE(428))
  (DE(431))
  (DE(434))
  (DE(437))
  (DE(440))
  (DE(443))
  (DE(446))
  (DE(449))
  (DE(452))
  (DE(455))
  (DE(458))
  (DE(460))
  (DE(463))
  (DE(466))
  (DE(469))
  (DE(472))
  (DE(475))
  (DE(501))
  (DE(504))
  (DE(508))
  (DE(511))
  (DE(514))
  (DE(517))

```

D.46 Subroutine RW2987 (concluded)

```

A57 = LPCON/57.29577951D0
CALL XJC(XLAMO,XC,V0,DELLAM,CAY1,CAY2,TOL,A57)
CALL TRAJT(TN,XC,VXT,XT)
CALL CROSS(XT,VXT)
CALL WIND(VXT,TN,XT,WZ,WX,V0,VZW,ZW)
YT = YA
ZT = ZA + ZW
VYT = VYA
VZT = VZA + VZW
CALL TRAJX(XN,XC,VXR,TX)
CALL CROSS(XN,VXR)
CALL WIND(VXR,TX,XN,WZ,WX,V0,VZW,ZW)
YR = YA
ZR = ZA + ZW
VYR = VYA
VZR = VZA + VZW
RETURN
END

```

```

HITS2838
HITS2839
HITS2840
HITS2841
HITS2842
HITS2843
HITS2844
HITS2845
HITS2846
HITS2847
HITS2848
HITS2849
HITS2850
HITS2851
HITS2852
HITS2853
HITS2854
HITS2855

```

D.47 Subroutine SAMPLE

```

SUBROUTINE SAMPLE
IMPLICIT REAL*8 (A-H,O-Z)
COMMON/CAROCM/ 8(200),C(5000),IA(200,2),ID(10,2),INC,IOPRNT,IOUT,
IY,KARS1(3),KA,K8,KC,KD,KT1,K234,LABC(3)
COMMON/OEC/OF(500)
COMMON/IMCH/ID1(200),ID2(200),I11(400),I12(400),IDT(10),IIT(20),
IIR1(1),IIR2(10),NTRIAL,NCELL,MCALC,MCOPT,IRJ1,IRJ2,IN8,NCC,NR1,NR2
DIMENSION IB(2,200)
EQUIVALENCE (IB(1,1),R(1))
THIS ROUTINE GENERATES THE INPUT SAMPLES AND REQUIRES A UNIFORM
RANDOM NUMBER GENERATOR CALLED RANDOM.
IRN=1
ENN=NCELL
DO 12 I=1,K234
ILP=0
IADD=IA(I,2)
CALL RANDOM(RN)
IF (IR(1,IADD)-3) 1,2,2
IF TYPE N3. IS 2 CONVERT RN TO A GAUSSIAN DISTRIBUTION
CALL CNVRT(RN)
VAL=B(IADD+1)+(2.*RN-1.)*B(IADD+2)
RNS=FN*RN
INDX=RNS
IIT(I)=IBN+INDX
VAL IS THE RANDOM VALUE AND IIT IS THE HISTOGRAM ADDRESS
IF (IR(1,IADD)-3) 5,5,4
STORE THE RANDOM VALUE IN THE OE ARRAY
IADD=IA(I,1)
OE(IADD)=VAL
GO TO 12
FOR TYPE 4 CALCULATE ACTUAL PRUB. DENSITY FOR CHOSEN RANDOM VALUE
INDX=IADD+4
JY=IB(1,INDX)
FEN=JY
FEN=FN*RN
JFEN
INDX=INDX+1
JF=IB(1,INDX)+J
JY=JY+JF
JF AND JY ARE ADDRESSES IN THE C ARRAY OF DIST. FUNC. AND VARIABL
VALUE RESP.--THAT IS VAL IS GT C (JY) AND LT C(JY+1)--INTERPOLATE
TO DETERMINE THE ACTUAL VALUE OF DIST. FCN. FOR CHOSEN VAL
FACT=C(JF)+(VAL-C(JY))*(C(JF+1)-C(JF))/(C(JY+1)-C(JY))
CALCULATE RANDOM PROBABILITY
CALL RANDOM(RN)
FTRY=RN*B(IADD+3)
IF (FACT-FTRY) 6,5,5
IF THE RANDOM VALUE IS GREATER THAN THE ACTUAL REJECT THIS SAMPLE
ILP=ILP+1
IF (ILP-20) 10,10,11

```

HITS2856
HITS2857
HITS2858
HITS2859
HITS2860
HITS2861
HITS2862
HITS2863
HITS2864
HITS2865
HITS2866
HITS2867
HITS2868
HITS2869
HITS2870
HITS2871
HITS2872
HITS2873
HITS2874
HITS2875
HITS2876
HITS2877
HITS2878
HITS2879
HITS2880
HITS2881
HITS2882
HITS2883
HITS2884
HITS2885
HITS2886
HITS2887
HITS2888
HITS2889
HITS2890
HITS2891
HITS2892
HITS2893
HITS2894
HITS2895
HITS2896
HITS2897
HITS2898
HITS2899
HITS2900
HITS2901
HITS2902
HITS2903
HITS2904
HITS2905

D.47 Subroutine SAMPLE (concluded)

HITS2906
HITS2907
HITS2908
HITS2909
HITS2910
HITS2911
HITS2912

```

C      IF A VALUE CANT BE FOUND IN 20 TRYS QUIT
      11  WRITE(IOUT,200)
      200  FORMAT(10X,'TYPE 4 RANDOM SAMPLE ERROR')
      CALL EXIT
      12  IBN=IBN+NCELL
          RETURN
          END

```

D.48

U

```

500      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+300),UNITS(I+IFOR),I=1,IFOR4)
501      FORMAT(1H1,38X,42H** PROJECTILE DISPERSION)
502      FORMAT(1H0,48X,23H FIRE CONTROL PARAMETERS)
503      FORMAT(1H0,49X,21H REAL WORLD PARAMETERS)
504      FORMAT(1H0,51X,16H SYSTEMS CONTROLS)
505      IFOR = IFOR + IFOR4
506      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+399),UNITS(I+IFOR),I=1,IFOR5)
507      IFOR = IFOR + IFOR5
508      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+499),UNITS(I+IFOR),I = 1,IFOR6)
509      IFOR = IFOR + IFOR6
510      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+599),UNITS(I+IFOR),I=1,IFOR7)
511      IFOR = IFOR + IFOR7
512      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+699),UNITS(I+IFOR),I=1,IFOR8)
513      IFOR = IFOR + IFOR8
514      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+799),UNITS(I+IFOR),I=1,IFOR9)
515      IFOR = IFOR + IFOR9
516      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+899),UNITS(I+IFOR),I=1,IFOR10)
517      IFOR = IFOR + IFOR10
518      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+999),UNITS(I+IFOR),I=1,IFOR11)
519      IFOR = IFOR + IFOR11
520      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1099),UNITS(I+IFOR),I=1,IFOR12)
521      IFOR = IFOR + IFOR12
522      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1199),UNITS(I+IFOR),I=1,IFOR13)
523      IFOR = IFOR + IFOR13
524      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1299),UNITS(I+IFOR),I=1,IFOR14)
525      IFOR = IFOR + IFOR14
526      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1399),UNITS(I+IFOR),I=1,IFOR15)
527      IFOR = IFOR + IFOR15
528      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1499),UNITS(I+IFOR),I=1,IFOR16)
529      IFOR = IFOR + IFOR16
530      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1599),UNITS(I+IFOR),I=1,IFOR17)
531      IFOR = IFOR + IFOR17
532      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1699),UNITS(I+IFOR),I=1,IFOR18)
533      IFOR = IFOR + IFOR18
534      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1799),UNITS(I+IFOR),I=1,IFOR19)
535      IFOR = IFOR + IFOR19
536      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1899),UNITS(I+IFOR),I=1,IFOR20)
537      IFOR = IFOR + IFOR20
538      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+1999),UNITS(I+IFOR),I=1,IFOR21)
539      IFOR = IFOR + IFOR21
540      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2099),UNITS(I+IFOR),I=1,IFOR22)
541      IFOR = IFOR + IFOR22
542      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2199),UNITS(I+IFOR),I=1,IFOR23)
543      IFOR = IFOR + IFOR23
544      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2299),UNITS(I+IFOR),I=1,IFOR24)
545      IFOR = IFOR + IFOR24
546      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2399),UNITS(I+IFOR),I=1,IFOR25)
547      IFOR = IFOR + IFOR25
548      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2499),UNITS(I+IFOR),I=1,IFOR26)
549      IFOR = IFOR + IFOR26
550      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2599),UNITS(I+IFOR),I=1,IFOR27)
551      IFOR = IFOR + IFOR27
552      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2699),UNITS(I+IFOR),I=1,IFOR28)
553      IFOR = IFOR + IFOR28
554      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2799),UNITS(I+IFOR),I=1,IFOR29)
555      IFOR = IFOR + IFOR29
556      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2899),UNITS(I+IFOR),I=1,IFOR30)
557      IFOR = IFOR + IFOR30
558      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+2999),UNITS(I+IFOR),I=1,IFOR31)
559      IFOR = IFOR + IFOR31
560      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3099),UNITS(I+IFOR),I=1,IFOR32)
561      IFOR = IFOR + IFOR32
562      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3199),UNITS(I+IFOR),I=1,IFOR33)
563      IFOR = IFOR + IFOR33
564      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3299),UNITS(I+IFOR),I=1,IFOR34)
565      IFOR = IFOR + IFOR34
566      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3399),UNITS(I+IFOR),I=1,IFOR35)
567      IFOR = IFOR + IFOR35
568      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3499),UNITS(I+IFOR),I=1,IFOR36)
569      IFOR = IFOR + IFOR36
570      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3599),UNITS(I+IFOR),I=1,IFOR37)
571      IFOR = IFOR + IFOR37
572      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3699),UNITS(I+IFOR),I=1,IFOR38)
573      IFOR = IFOR + IFOR38
574      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3799),UNITS(I+IFOR),I=1,IFOR39)
575      IFOR = IFOR + IFOR39
576      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3899),UNITS(I+IFOR),I=1,IFOR40)
577      IFOR = IFOR + IFOR40
578      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+3999),UNITS(I+IFOR),I=1,IFOR41)
579      IFOR = IFOR + IFOR41
580      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4099),UNITS(I+IFOR),I=1,IFOR42)
581      IFOR = IFOR + IFOR42
582      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4199),UNITS(I+IFOR),I=1,IFOR43)
583      IFOR = IFOR + IFOR43
584      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4299),UNITS(I+IFOR),I=1,IFOR44)
585      IFOR = IFOR + IFOR44
586      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4399),UNITS(I+IFOR),I=1,IFOR45)
587      IFOR = IFOR + IFOR45
588      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4499),UNITS(I+IFOR),I=1,IFOR46)
589      IFOR = IFOR + IFOR46
590      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4599),UNITS(I+IFOR),I=1,IFOR47)
591      IFOR = IFOR + IFOR47
592      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4699),UNITS(I+IFOR),I=1,IFOR48)
593      IFOR = IFOR + IFOR48
594      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4799),UNITS(I+IFOR),I=1,IFOR49)
595      IFOR = IFOR + IFOR49
596      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4899),UNITS(I+IFOR),I=1,IFOR50)
597      IFOR = IFOR + IFOR50
598      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+4999),UNITS(I+IFOR),I=1,IFOR51)
599      IFOR = IFOR + IFOR51
600      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5099),UNITS(I+IFOR),I=1,IFOR52)
601      IFOR = IFOR + IFOR52
602      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5199),UNITS(I+IFOR),I=1,IFOR53)
603      IFOR = IFOR + IFOR53
604      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5299),UNITS(I+IFOR),I=1,IFOR54)
605      IFOR = IFOR + IFOR54
606      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5399),UNITS(I+IFOR),I=1,IFOR55)
607      IFOR = IFOR + IFOR55
608      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5499),UNITS(I+IFOR),I=1,IFOR56)
609      IFOR = IFOR + IFOR56
610      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5599),UNITS(I+IFOR),I=1,IFOR57)
611      IFOR = IFOR + IFOR57
612      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5699),UNITS(I+IFOR),I=1,IFOR58)
613      IFOR = IFOR + IFOR58
614      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5799),UNITS(I+IFOR),I=1,IFOR59)
615      IFOR = IFOR + IFOR59
616      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5899),UNITS(I+IFOR),I=1,IFOR60)
617      IFOR = IFOR + IFOR60
618      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+5999),UNITS(I+IFOR),I=1,IFOR61)
619      IFOR = IFOR + IFOR61
620      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6099),UNITS(I+IFOR),I=1,IFOR62)
621      IFOR = IFOR + IFOR62
622      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6199),UNITS(I+IFOR),I=1,IFOR63)
623      IFOR = IFOR + IFOR63
624      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6299),UNITS(I+IFOR),I=1,IFOR64)
625      IFOR = IFOR + IFOR64
626      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6399),UNITS(I+IFOR),I=1,IFOR65)
627      IFOR = IFOR + IFOR65
628      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6499),UNITS(I+IFOR),I=1,IFOR66)
629      IFOR = IFOR + IFOR66
630      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6599),UNITS(I+IFOR),I=1,IFOR67)
631      IFOR = IFOR + IFOR67
632      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6699),UNITS(I+IFOR),I=1,IFOR68)
633      IFOR = IFOR + IFOR68
634      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6799),UNITS(I+IFOR),I=1,IFOR69)
635      IFOR = IFOR + IFOR69
636      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6899),UNITS(I+IFOR),I=1,IFOR70)
637      IFOR = IFOR + IFOR70
638      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+6999),UNITS(I+IFOR),I=1,IFOR71)
639      IFOR = IFOR + IFOR71
640      WRITE(IOUT,505)(HEAD(I+IFOR),OE(I+7099),UNITS(I+IFOR),I=1,IFOR72)
641      IFOR = IFOR + IFOR72
642      WRITE(IOUT,505)(HEAD(I+IFOR),OE
```

HI T S2963
 HI T S2964
 HI T S2965
 HI T S2966
 HI T S2967
 HI T S2968
 HI T S2969
 HI T S2970
 HI T S2971
 HI T S2972
 HI T S2973
 HI T S2974
 HI T S2975
 HI T S2976

D.49 Subroutine STDDEV

```

SUBROUTINE STDDEV(A,K,KD,IOUT)
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION B(10)
B(K) = DSORT(A)
IF(K.NE.KD) RETURN
WRITE(IOUT,8887)(B(I), I = 1,KD)
WRITE(IOUT,8889)
WRITE(IOUT,8888)(B(I),B(I),I=1,KD)
8887 FORMAT(1H,9HSTD. DEV.,1X,1PE12.4)
8889 FORMAT(1H,7//.3X,.STANDARD DEVIATIONS WITH HIGHER PRECISION',/)
8888 FORMAT(1H,1PE12.4,5X,D25.16)
RETURN
END
HITS2977
HITS2978
HITS2979
HITS2980
HITS2981
HITS2982
HITS2983
HITS2984
HITS2985
HITS2986
HITS2987
HITS2988
HITS2989

```


D.50 Subroutine STOREC

```

SUBROUTINE STOREC
  IMPLICIT REAL*8 (A-H,O-Z)
  COMMON /CARDCM/ B(200), C(5000), IA(200,2), ID(10,2), INC,
  1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KT1, K234, LABC(3)
  COMMON/OEC/OE(600)
  C PLACE THE KD DEPENDENT VARIABLES INTO THE C ARRAY.
  DO 10 I=1,KD
    JOE = ID(I,1)
    KC = KC + 1
    IF (KC.GT.5000) WRITE(IOUT,9000) KC
    9000 FORMAT(10X,'KC = ',I5,1X,'IN STOREC WHICH EXCEEDS THE C ARRAY.')
    C(KC) = OE(JOE)
    10 CONTINUE
  RETURN
END

```

HITS2990
 HITS2991
 HITS2992
 HITS2993
 HITS2994
 HITS2995
 HITS2996
 HITS2997
 HITS2998
 HITS2999
 HITS3000
 HITS3001
 HITS3002
 HITS3003
 HITS3004

D.51 Subroutine TGI795

SUBROUTINE TGI795
RETURN
END

HITS3005
HITS3006
HITS3007

D.52 Subroutine TG2440

SUBROUTINE TG2440
RETURN
END

HITS3008
HITS3009
HITS3010

D.53

Subroutine	Parameter	Value	Subroutine	Parameter	Value
SUBROUTINE TG2987	IMPLICIT REAL*8 (A-H,O-Z)		FIRE CONTROL SYSTEM PARAMETERS	VOF	= +1.000000D4
	COMMON /CARDCM/ B(200), C(5000), IA(200,2), ID(10,2), INC,			WVF	= +0.000000D0
	1 IOPRNT, IOUT, IY, KARS1(3), KA, KB, KC, KD, KTI, K234, LABC(3)			WYF	= +0.000000D0
	COMMON/OEC/OE(600)			RHOF	= +0.002378D0
	EQUIVALENCE			CDIF	= +3.585398D-2
	A(WYF, OE(3)), (VDF, OE(1)), (WVF, OE(2)),			V1F	= +1.674000D4
	B(WVF, OE(6)), (CD2F, OE(7)), (V2F, OE(8)),			CD2F	= +1.195133D-1
	C(XNF, OE(9)), (CNAF, OE(10)), (SMF, OE(11)),			V2F	= +3.906000D3
	D(CMQF, OE(12)), (CMPAF, OE(13)), (ATRM SF, OE(14)),				
	E(BTRMSF, OE(15)), (THETOF, OE(16)), (PSIOF, OE(17)),				
F(TDOTOF, OE(18)), (PDOTOF, OE(19)), (GAMOF, OE(20)),					
G(AZOF, OE(21)), (AF, OE(22)), (ELLF, OE(23)),					
H(DF, OE(24)), (WF, OE(25)), (AIXF, OE(26)),					
I(AIYF, OE(27)), (PF, OE(28)), (ALPCOF, OE(29)),					
J(CAAF, OE(30))					
EQUIVALENCE					
A(WY, OE(103)), (RHQ, OE(104)), (WX, OE(102)),					
B(V1, OE(106)), (CD2, OE(107)), (V2, OE(108)),					
C(XN, OE(109)), (CNA, OE(110)), (SM, OE(111)),					
D(CMQ, OE(112)), (CMPA, OE(113)), (ATRMS, OE(114)),					
E(BTRMS, OE(115)), (THETO, OE(116)), (PSIO, OE(117)),					
F(TDOT, OE(118)), (FDDOT, OE(119)), (GAMO, OE(120)),					
G(AZO, OE(121)), (A, OE(122)), (ELL, OE(123)),					
H(D, OE(124)), (W, OE(125)), (AIX, OE(126)),					
I(AIY, OE(127)), (P, OE(128)), (ALPCON, OE(129)),					
J(CAA, OE(130))					
EQUIVALENCE					
A(ZBART, OE(133)), (XBART, OE(131)), (YBART, OE(132)),					
EQUIVALENCE					
A(ZBART, OE(133)), (YBART, OE(134)), (ZBART, OE(135)),					
EQUIVALENCE					
A(FC, OE(203)), (IFCRW, OE(201)), (IFCRC, OE(202)),					
EQUIVALENCE					
A(FC, OE(203)), (IDUMP, OE(204)), (ICNCL, OE(206)),					
EQUIVALENCE					
A(FC, OE(203)), (ICNV, OE(205)), (YBARTI, OE(302)),					
EQUIVALENCE					
A(ZBARTI, OE(303)), (YBARTI, OE(301)), (YBARTI, OE(302)),					
EQUIVALENCE					
A(ZBARTI, OE(303)), (YBARTI, OE(304)), (ZBARTI, OE(305)),					
EQUIVALENCE					
A(ZBARTI, OE(303)), (TOL, OE(453))					

D.53 Subroutine TG2987 (continued)

```

XNF      = +1.000000D4
CNAF     = +1.976700D0
SMF      = +6.200000D-2
CMQF     = -7.500000D0
CNPAF    = +0.000000D0
ATRMSE   = +0.000000D0
BTRMSF   = +0.000000D0
THETOF   = +0.000000D0
PSIOF    = +0.000000D0
TDOTOF   = +0.000000D0
PDOTOF   = +0.000000D0
GAMOF     = +0.000000D0
AZOF      = +0.000000D0
AF        = +3.068000D-3
ELLF     = +3.100000D-1
DF        = +6.250000D-2
WF        = +1.100000D-1
AIXF     = +1.234600D-6
AIVF     = +2.110300D-5
PF        = +4.000000D2
ALPCQF   = +5.000000D-1
CAAF      = +0.000000D0

```

PROJECTILE PARAMETERS (REAL WORLD)

```

VO      = +1.100000D4
WX      = +0.000000D0
WY      = +0.000000D0
RHO     = +0.002378D0
CD1     = +3.585398D-2
V1      = +1.674000D4
CD2     = +1.195133D-1
V2      = +3.906000D3
XN      = +1.000000D4
CNA      = +1.976700D0
SM      = +6.200000D-2
CMQ     = -7.500000D0
CMPA    = +0.000000D0
ATRMS   = +0.000000D0
BTRMSF  = +0.000000D0
THETO   = +0.000000D0
PSIO    = +0.000000D0
TDOTO   = +0.000000D0
PDOTO   = +0.000000D0
GAMO    = +0.000000D0
AZO     = +0.000000D0
A        = +3.068000D-3

```

```

HITS3061
HITS3062
HITS3063
HITS3064
HITS3065
HITS3066
HITS3067
HITS3068
HITS3069
HITS3070
HITS3071
HITS3072
HITS3073
HITS3074
HITS3075
HITS3076
HITS3077
HITS3078
HITS3079
HITS3080
HITS3081
HITS3082
HITS3083
HITS3084
HITS3085
HITS3086
HITS3087
HITS3088
HITS3089
HITS3090
HITS3091
HITS3092
HITS3093
HITS3094
HITS3095
HITS3096
HITS3097
HITS3098
HITS3099
HITS3100
HITS3101
HITS3102
HITS3103
HITS3104
HITS3105
HITS3106
HITS3107
HITS3108
HITS3109
HITS3110

```

D 53 Subroutine TG29C7 (concluded)

```

ELL      = +3.100000D-1
D        = +6.250000D-2
W        = +1.100000D-1
AIX      = +1.234600D-6
AIY      = +2.110300D-5
P        = +4.000000D-2
ALPCON   = +5.000000D-1
CAA      = +0.000000D0
XBART    = +0.000000D0
YBART    = +0.000000D0
ZBART    = +0.000000D0
YBARX    = +0.000000D0
ZBARX    = +0.000000D0

```

C
C
C
C
C

SYSTEM CONTROLS

```

IFCRW    = 0
IFCRC     = 0
IFC       = 1
IDUMP     = 0
ICONV     = -10
ICNCL     = 1

```

C
C
C
C
C

COMPUTED QUANTITIES

```

DO 1 I = 301,600
DE(I) = +0.000000D0
TOL = 10.**ICONV
RETURN
END

```

1

```

HITS3111
HITS3112
HITS3113
HITS3114
HITS3115
HITS3116
HITS3117
HITS3118
HITS3119
HITS3120
HITS3121
HITS3122
HITS3123
HITS3124
HITS3125
HITS3126
HITS3127
HITS3128
HITS3129
HITS3130
HITS3131
HITS3132
HITS3133
HITS3134
HITS3135
HITS3136
HITS3137
HITS3138
HITS3139
HITS3140
HITS3141
HITS3142
HITS3143
HITS3144
HITS3145

```

D.54 Subroutine TRAJT

```

SUBROUTINE TRAJT( T, XC, VX, X )
  IMPLICIT REAL * 8 (A-H, O-Z)
  COMMON/GECE/OE(600)
  EQUIVALENCE
    A(PHIWIN, OE(103)), (V0, OE(101)), (VWIND, OE(102)),
    B(V1, OE(106)), (CD2, OE(104)), (CD1, OE(105)),
    C(XN, OE(109)), (CNA, OE(107)), (V2, OE(108)),
    D(CMQ, OE(112)), (CMPA, OE(110)), (SM, OE(111)),
    E(PNITRI, OE(115)), (ANGO, OE(113)), (TRIM, OE(114)),
    F(PHATE, OE(118)), (PHIRAT, OE(116)), (PHIANG, OE(117)),
    G(PHIGAM, OE(121)), (A, OE(119)), (GAMO, OE(120)),
    H(D, OE(124)), (W, OE(122)), (ELL, OE(123)),
    I(AIY, OE(127)), (P, OE(125)), (AIX, OE(126)),
    J(CAA, OE(130)), (IDUMP, OE(128)), (ALPCON, OE(129))
  EQUIVALENCE
    A(ALPTRM, OE(402)), (ALPHA, OE(204)), (ATOT, OE(401)),
    B(BETTRM, OE(405)), (ALPHD0, OE(403)), (B, OE(404)),
    C(BETA, OE(408)), (ALPHD0, OE(406)), (BETAD0, OE(407)),
    D(CAYD, OE(411)), (ALPMAX, OE(409)), (ALPMIN, OE(410)),
    E(CMTHD, OE(414)), (CD8, OE(412)), (CMA, OE(413)),
    F(CAY2, OE(417)), (CMTHA, OE(415)), (CAY1, OE(416)),
    G(DELT, OE(420)), (CDACR, OE(418)), (COAB, OE(419)),
    H(DELT, OE(423)), (DELV, OE(421)), (DELT, OE(422)),
    I(DYDT, OE(426)), (DYDX0, OE(424)), (DZDX0, OE(425)),
    J(DELLAM, OE(429)), (DZDT0, OE(427)), (DELW, OE(428)),
    K(EOLT, OE(432)), (EYEP, OE(430)), (EMP, OE(431)),
    L(H, OE(435)), (EOLT, OE(433)), (F, OE(434)),
    M(PHI1, OE(438)), (PSID0, OE(436)), (PHIO, OE(437)),
    N(R1, OE(441)), (PHI2, OE(439)), (PSIO, OE(440)),
    O(R4, OE(444)), (R2, OE(442)), (R3, OE(443)),
    P(TCO, OE(447)), (RTRIM, OE(445)), (TC, OE(446)),
    Q(TP, OE(450)), (TS, OE(448)), (TGO, OE(449)),
    R(TOL, OE(453)), (THETA0, OE(451)), (THETD0, OE(452)),
    S(VYA, OE(456)), (VC, OE(454)), (VCO, OE(455)),
    EQUIVALENCE
      A(W1, OE(461)), (WZ, OE(457)), (WX, OE(458)),
      B(XLAM, OE(464)), (W2, OE(459)), (WL2, OE(460)),
      C(XNU1, OE(467)), (XLAM1, OE(465)), (WL2, OE(463)),
      D(XMU, OE(470)), (XNU2, OE(468)), (XLAM2, OE(466)),
      E(XJA, OE(473)), (XJAY, OE(471)), (XG, OE(469)),
      F(ALPH, OE(476)), (YA, OE(474)), (XJAZ, OE(472)),
      CALL VXT(VC, TCO, V0, WX, F, B, CD8, OE(475))
  IF (DABS(XC), EQ, 1.0-10) VCO=V0
  A2X = A2(XC)
  COA0B = CD8 + DLOG( CD8 + CAYD/ VCO/VC0 ) / ( CD8 + CAYD/V0/V0 )
  XC=CAYD * TOL
  A 1) / 2.00 / F / XC
  COAB = COA0B + (CNA + CAA) / XC * A2X
  EF = DEXP( F * CDAB* XC)

```

HITS3146
 HITS3147
 HITS3148
 HITS3149
 HITS3150
 HITS3151
 HITS3152
 HITS3153
 HITS3154
 HITS3155
 HITS3156
 HITS3157
 HITS3158
 HITS3159
 HITS3160
 HITS3161
 HITS3162
 HITS3163
 HITS3164
 HITS3165
 HITS3166
 HITS3167
 HITS3168
 HITS3169
 HITS3170
 HITS3171
 HITS3172
 HITS3173
 HITS3174
 HITS3175
 HITS3176
 HITS3177
 HITS3178
 HITS3179
 HITS3180
 HITS3181
 HITS3182
 HITS3183
 HITS3184
 HITS3185
 HITS3186
 HITS3187
 HITS3188
 HITS3189
 HITS3190
 HITS3191
 HITS3192
 HITS3193
 HITS3194
 HITS3195

```

HITS3196
HITS3197
HITS3198
HITS3199
HITS3200
HITS3201
HITS3202
HITS3203
HITS3204
HITS3205
HITS3206
HITS3207
HITS3208
HITS3209
HITS3210
HITS3211
HITS3212
HITS3213
HITS3214
HITS3215
HITS3216
HITS3217
HITS3218
HITS3219
HITS3220
HITS3221
HITS3222
HITS3223
HITS3224
HITS3225
HITS3226
HITS3227
HITS3228
HITS3229
HITS3230
HITS3231
HITS3232
HITS3233
HITS3234
HITS3235
HITS3236
HITS3237
HITS3238
HITS3239
HITS3240
HITS3241
HITS3242
HITS3243
HITS3244
HITS3245
HITS3246
HITS3247
HITS3248
HITS3249
HITS3250
HITS3251
HITS3252
HITS3253
HITS3254
HITS3255
HITS3256
HITS3257
HITS3258
HITS3259
HITS3260
HITS3261
HITS3262
HITS3263
HITS3264
HITS3265
HITS3266
HITS3267
HITS3268
HITS3269
HITS3270
HITS3271
HITS3272
HITS3273
HITS3274
HITS3275
HITS3276
HITS3277
HITS3278
HITS3279
HITS3280
HITS3281
HITS3282
HITS3283
HITS3284
HITS3285
HITS3286
HITS3287
HITS3288
HITS3289
HITS3290
HITS3291
HITS3292
HITS3293
HITS3294
HITS3295
HITS3296
HITS3297
HITS3298
HITS3299
HITS3300
HITS3301
HITS3302
HITS3303
HITS3304
HITS3305
HITS3306
HITS3307
HITS3308
HITS3309
HITS3310
HITS3311
HITS3312
HITS3313
HITS3314
HITS3315
HITS3316
HITS3317
HITS3318
HITS3319
HITS3320
HITS3321
HITS3322
HITS3323
HITS3324
HITS3325
HITS3326
HITS3327
HITS3328
HITS3329
HITS3330
HITS3331
HITS3332
HITS3333
HITS3334
HITS3335
HITS3336
HITS3337
HITS3338
HITS3339
HITS3340
HITS3341
HITS3342
HITS3343
HITS3344
HITS3345
HITS3346
HITS3347
HITS3348
HITS3349
HITS3350
HITS3351
HITS3352
HITS3353
HITS3354
HITS3355
HITS3356
HITS3357
HITS3358
HITS3359
HITS3360
HITS3361
HITS3362
HITS3363
HITS3364
HITS3365
HITS3366
HITS3367
HITS3368
HITS3369
HITS3370
HITS3371
HITS3372
HITS3373
HITS3374
HITS3375
HITS3376
HITS3377
HITS3378
HITS3379
HITS3380
HITS3381
HITS3382
HITS3383
HITS3384
HITS3385
HITS3386
HITS3387
HITS3388
HITS3389
HITS3390
HITS3391
HITS3392
HITS3393
HITS3394
HITS3395
HITS3396
HITS3397
HITS3398
HITS3399
HITS3400
HITS3401
HITS3402
HITS3403
HITS3404
HITS3405
HITS3406
HITS3407
HITS3408
HITS3409
HITS3410
HITS3411
HITS3412
HITS3413
HITS3414
HITS3415
HITS3416
HITS3417
HITS3418
HITS3419
HITS3420
HITS3421
HITS3422
HITS3423
HITS3424
HITS3425
HITS3426
HITS3427
HITS3428
HITS3429
HITS3430
HITS3431
HITS3432
HITS3433
HITS3434
HITS3435
HITS3436
HITS3437
HITS3438
HITS3439
HITS3440
HITS3441
HITS3442
HITS3443
HITS3444
HITS3445
HITS3446
HITS3447
HITS3448
HITS3449
HITS3450
HITS3451
HITS3452
HITS3453
HITS3454
HITS3455
HITS3456
HITS3457
HITS3458
HITS3459
HITS3460
HITS3461
HITS3462
HITS3463
HITS3464
HITS3465
HITS3466
HITS3467
HITS3468
HITS3469
HITS3470
HITS3471
HITS3472
HITS3473
HITS3474
HITS3475
HITS3476
HITS3477
HITS3478
HITS3479
HITS3480
HITS3481
HITS3482
HITS3483
HITS3484
HITS3485
HITS3486
HITS3487
HITS3488
HITS3489
HITS3490
HITS3491
HITS3492
HITS3493
HITS3494
HITS3495
HITS3496
HITS3497
HITS3498
HITS3499
HITS3500
HITS3501
HITS3502
HITS3503
HITS3504
HITS3505
HITS3506
HITS3507
HITS3508
HITS3509
HITS3510
HITS3511
HITS3512
HITS3513
HITS3514
HITS3515
HITS3516
HITS3517
HITS3518
HITS3519
HITS3520
HITS3521
HITS3522
HITS3523
HITS3524
HITS3525
HITS3526
HITS3527
HITS3528
HITS3529
HITS3530
HITS3531
HITS3532
HITS3533
HITS3534
HITS3535
HITS3536
HITS3537
HITS3538
HITS3539
HITS3540
HITS3541
HITS3542
HITS3543
HITS3544
HITS3545
HITS3546
HITS3547
HITS3548
HITS3549
HITS3550
HITS3551
HITS3552
HITS3553
HITS3554
HITS3555
HITS3556
HITS3557
HITS3558
HITS3559
HITS3560
HITS3561
HITS3562
HITS3563
HITS3564
HITS3565
HITS3566
HITS3567
HITS3568
HITS3569
HITS3570
HITS3571
HITS3572
HITS3573
HITS3574
HITS3575
HITS3576
HITS3577
HITS3578
HITS3579
HITS3580
HITS3581
HITS3582
HITS3583
HITS3584
HITS3585
HITS3586
HITS3587
HITS3588
HITS3589
HITS3590
HITS3591
HITS3592
HITS3593
HITS3594
HITS3595
HITS3596
HITS3597
HITS3598
HITS3599
HITS3600
HITS3601
HITS3602
HITS3603
HITS3604
HITS3605
HITS3606
HITS3607
HITS3608
HITS3609
HITS3610
HITS3611
HITS3612
HITS3613
HITS3614
HITS3615
HITS3616
HITS3617
HITS3618
HITS3619
HITS3620
HITS3621
HITS3622
HITS3623
HITS3624
HITS3625
HITS3626
HITS3627
HITS3628
HITS3629
HITS3630
HITS3631
HITS3632
HITS3633
HITS3634
HITS3635
HITS3636
HITS3637
HITS3638
HITS3639
HITS3640
HITS3641
HITS3642
HITS3643
HITS3644
HITS3645
HITS3646
HITS3647
HITS3648
HITS3649
HITS3650
HITS3651
HITS3652
HITS3653
HITS3654
HITS3655
HITS3656
HITS3657
HITS3658
HITS3659
HITS3660
HITS3661
HITS3662
HITS3663
HITS3664
HITS3665
HITS3666
HITS3667
HITS3668
HITS3669
HITS3670
HITS3671
HITS3672
HITS3673
HITS3674
HITS3675
HITS3676
HITS3677
HITS3678
HITS3679
HITS3680
HITS3681
HITS3682
HITS3683
HITS3684
HITS3685
HITS3686
HITS3687
HITS3688
HITS3689
HITS3690
HITS3691
HITS3692
HITS3693
HITS3694
HITS3695
HITS3696
HITS3697
HITS3698
HITS3699
HITS3700
HITS3701
HITS3702
HITS3703
HITS3704
HITS3705
HITS3706
HITS3707
HITS3708
HITS3709
HITS3710
HITS3711
HITS3712
HITS3713
HITS3714
HITS3715
HITS3716
HITS3717
HITS3718
HITS3719
HITS3720
HITS3721
HITS3722
HITS3723
HITS3724
HITS3725
HITS3726
HITS3727
HITS3728
HITS3729
HITS3730
HITS3731
HITS3732
HITS3733
HITS3734
HITS3735
HITS3736
HITS3737
HITS3738
HITS3739
HITS3740
HITS3741
HITS3742
HITS3743
HITS3744
HITS3745
HITS3746
HITS3747
HITS3748
HITS3749
HITS3750
HITS3751
HITS3752
HITS3753
HITS3754
HITS3755
HITS3756
HITS3757
HITS3758
HITS3759
HITS3760
HITS3761
HITS3762
HITS3763
HITS3764
HITS3765
HITS3766
HITS3767
HITS3768
HITS3769
HITS3770
HITS3771
HITS3772
HITS3773
HITS3774
HITS3775
HITS3776
HITS3777
HITS3778
HITS3779
HITS378
```


D.54 Subroutine TRAJT (concluded)

```

1001  FORMAT( ' TG DID NOT CONVERGE IN 100 TRIES.....' ,/// )
      3  CONTINUE
        X = XG
        RETURN
      END
HITS3246
HITS3247
HITS3248
HITS3249
HITS3250
HITS3251

```

D.55 Subroutine TRAJX

```

SUBROUTINE TRAJX( A, XC, VX, T )
  IMPLICIT REAL * 8(A-H, O-Z)
  COMMON/OE/ OE(600)
  EQUIVALENCE
    A(PHIWIN), OE( 103)), (RHO), OE( 101)), (VWIND), OE( 102)),
    B(VI), OE( 106)), (CD2), OE( 104)), (CD1), OE( 105)),
    C(XN), OE( 109)), (CNA), OE( 107)), (V2), OE( 108)),
    D(CMG), OE( 112)), (CMPA), OE( 110)), (SM), OE( 111)),
    E(PNITRI), OE( 115)), (ANGC), OE( 113)), (TRIM), OE( 114)),
    F(RATE), OE( 119)), (PHIRAT), OE( 116)), (PHIANG), OE( 117)),
    G(PHIGAM), OE( 121)), (A), OE( 119)), (GAMG), OE( 120)),
    H(C), OE( 124)), (W), OE( 122)), (ELL), OE( 123)),
    I(CAIY), OE( 127)), (P), OE( 125)), (AIK), OE( 126)),
    J(CAA), OE( 130)), (IDUMP), OE( 128)), (ALPCON), OE( 129)),
    EQUIVALENCE
    A(ALPTON), OE( 402)), (ALPHA), OE( 400)), (ATOT), OE( 401)),
    B(BETTRM), OE( 405)), (ALPTON), OE( 403)), (B), OE( 404)),
    C(BETA), OE( 409)), (ALPMAX), OE( 407)), (BETADO), OE( 407)),
    D(CAYD), OE( 411)), (CDB), OE( 409)), (ALPMIN), OE( 410)),
    E(CMTHTD), OE( 414)), (CMTHTA), OE( 412)), (CMA), OE( 413)),
    F(CAY2), OE( 417)), (CDA09), OE( 415)), (CAY1), OE( 416)),
    G(DELTY), OE( 420)), (DELV), OE( 418)), (CDAB), OE( 419)),
    H(DELX), OE( 423)), (DYDX), OE( 421)), (DEL), OE( 422)),
    I(DYDT), OE( 426)), (DZDT), OE( 424)), (DZDX), OE( 425)),
    J(DELTA), OE( 429)), (EYEP), OE( 430)), (DELW), OE( 428)),
    K(DELTA), OE( 432)), (EDLT), OE( 433)), (EMP), OE( 431)),
    L(H), OE( 435)), (PSID), OE( 435)), (F), OE( 434)),
    M(PHII), OE( 438)), (PHI2), OE( 439)), (PHI), OE( 437)),
    N(RI), OE( 441)), (R2), OE( 442)), (PSI), OE( 440)),
    O(R4), OE( 444)), (RTRIM), OE( 445)), (R3), OE( 443)),
    P(TCO), OE( 447)), (TS), OE( 448)), (TC), OE( 446)),
    Q(TP), OE( 450)), (THETA0), OE( 451)), (TGT), OE( 449)),
    R(TOL), OE( 453)), (VC), OE( 454)), (THETD), OE( 452)),
    S(VYA), OE( 456)), (VZA), OE( 457)), (VC), OE( 455)),
    EQUIVALENCE
    A(WI), OE( 461)), (WZ), OE( 459)), (WX), OE( 458)),
    B(XLAM), OE( 464)), (W2), OE( 462)), (W0), OE( 463)),
    C(XNU1), OE( 467)), (XLAM1), OE( 465)), (WL2), OE( 463)),
    D(XMU), OE( 470)), (XNU2), OE( 468)), (XLAM2), OE( 466)),
    E(XJA), OE( 473)), (XJAY), OE( 471)), (XG), OE( 469)),
    F(ALPH), OE( 476)), (YA), OE( 474)), (XJAZ), OE( 472)),
    IF( X, LE, XC ) GO TO 1
    XI = XC
    GO TO 2
  XI = X
  CONTINUE
  IF( XI, EQ, 0.) XI=1.D-10
  CALL VXT(VCO, TC0, V)
  IF( DABS( XI), EQ, 1.D-10) VCO=V0
  .WX, F.B, CD8, XI, CAYD, TOL )

```

1
2

D.55 Subroutine TRAJX (concluded)

```

      A2X = A2( X1 )
      FX1 = F * X1
      CDA08 = CD8 + DLOG( CD8 + CAYD/ VC0/VC0 )/(CD8 +CAYD/VO/VO)
1 ) / 2.D0 / F / X1
      CDAB = CDA08 + (CNA + CAA)/X1 * A2X
      EFCX = DEXP( FX1 * CDAB )
      EFCX0 = DEXP( FX1 * CDA08 )
      DELV = VO * (1.D0/EFCX -1.D0/EFCX0 )
      DELTT = ((EFCX -1.D0 )/ CDAB - (EFCX0 -1.D0)/CDA08)/F /VO
      VC = VC0 + DELV
      TC = TC0 + DELTT
      IF( X .LE. XC ) GO TO 3
      XS = X -XC
      CALL VXT(VX , TS ,VC ,WX, F, B ,CD8, XS, CAYD, TOL )
      T = TS + TC
      GO TO 4
3  CONTINUE
      VX = VC
      T = TC
      IF( IDUMP .EQ. 1) WRITE(6,1000) X , T , X1, VC0 , TC0
1000 FORMAT( , A2X , CDA08 , CDAB , EFCX , EFCX0 X1 VC0 EFCX0
1001 FORMAT( , X CDA08 T CDAB X1 EFCX VC0 EFCX0
2 , / 1X , IP10E12.4 // )
      IF( IDUMP , .EQ. 1) WRITE(6,1001) DELV , DELTT, VC, TC ,XS,VX
1001 FORMAT( , DELV , DELTT VC TC
15 , / IP10E12.4 // )
      RETURN
      END

```

HITS3302
HITS3303
HITS3304
HITS3305
HITS3306
HITS3307
HITS3308
HITS3309
HITS3310
HITS3311
HITS3312
HITS3313
HITS3314
HITS3315
HITS3316
HITS3317
HITS3318
HITS3319
HITS3320
HITS3321
HITS3322
HITS3323
HITS3324
HITS3325
HITS3326
HITS3327
HITS3328
HITS3329
HITS3330

D.56 Subroutine TRAJXF

```

SUBROUTINE TRAJXF( X, XC, VX, Y, TY )
  IMPLICIT REAL * 8(A-H, O-Z)
  COMMON/OEC/OE(500)
  EQUIVALENCE
    A(APHIWIN, .OE( 3)), (VQ, .OE( 1)), (VWIND, .OE( 2)),
    B(VI, .OE( 6)), (CD2, .OE( 4)), (CD1, .OE( 5)),
    C(XN, .OE( 9)), (CNA, .OE( 7)), (V2, .OE( 8)),
    D(CMQ, .OE(12)), (CMPA, .OE(10)), (SM, .OE(11)),
    E(PNITRI, .OE(15)), (ANGQ, .OE(13)), (TRIM, .OE(14)),
    F(RATE, .OE(18)), (PHIRAT, .OE(16)), (PHIANG, .OE(17)),
    G(PHIGAM, .OE(21)), (A, .OE(19)), (GAMQ, .OE(20)),
    H(J, .OE(24)), (W, .OE(22)), (ELL, .OE(23)),
    I(ATY, .OE(27)), (P, .OE(25)), (ATX, .OE(26)),
    J(CAA, .OE(30)), (I, .OE(28)), (ALPCON, .OE(29))

  EQUIVALENCE
    A(ALPHA, .OE(42)), (IQUIMP, .OE(204)),
    B(BETA, .OE(45)), (ALPHA, .OE(40)),
    C(CA2, .OE(48)), (BETA, .OE(43)),
    D(CAYD, .OE(41)), (ALPMAX, .OE(405)),
    E(CMTH'D, .OE(47)), (CD8, .OE(412)),
    F(CAY2, .OE(417)), (CMTHTA, .OE(415)),
    G(DELTA, .OE(423)), (CDAOB, .OE(418)),
    H(DELX, .OE(426)), (DELV, .OE(421)),
    I(DYDT, .OE(429)), (DYDX, .OE(424)),
    J(DELTA, .OE(432)), (EYEP, .OE(430)),
    K(DELTA, .OE(435)), (EDLT, .OE(433)),
    L(PHI1, .OE(438)), (PSID3, .OE(436)),
    M(PHI, .OE(441)), (PHI2, .OE(439)),
    N(RI, .OE(444)), (R2, .OE(442)),
    O(R4, .OE(447)), (RTRIM, .OE(445)),
    P(ICO, .OE(450)), (TS, .OE(448)),
    Q(TP, .OE(453)), (THETAQ, .OE(451)),
    R(TOL, .OE(456)), (VC, .OE(454)),
    S(VYA, .OE(459)), (VZA, .OE(457)),
    EQUIVALENCE
    A(WI, .OE(461)), (WZ, .OE(452)),
    B(XLAM, .OE(464)), (W2, .OE(462)),
    C(XNU1, .OE(467)), (XLAM1, .OE(455)),
    D(XMU, .OE(470)), (XNU2, .OE(468)),
    E(XJA, .OE(473)), (XJAY, .OE(471)),
    F(ALPH, .OE(476)), (YA, .OE(474))

  IF( X .LE. XC ) GO TO 1
  XI = XC
  GU TO 2
  XI = X
  CONTINUE
  IF( XI .EQ. 0.) XI=1.D-10
  CALL VXT(VC), TCO, VQ
  IF( DARS( XI ), EQ. 1.D-10) VQ=VQ
  IF( X, F.B. CD8, XI, CAYD, TOL )

```

1 2

D.56 Subroutine TRAJXF (concluded)

```

      A2X = A2F( X1 )
      FX1 = F * X1
      COA0B = CD8 + BLOG( (CD8 + CAYD/ VC0/VC0)/(CD8 + CAYD/VC0/VC0) )
      1 ) / 2.00 / F / X1
      CDAB = COA0B + (CNA + CAA)/X1 * A2X
      EFCX = DEXP( FX1 * CDAB )
      EFCX0 = DEXP( FX1 * CDA0B )
      DELV = V0 * (1.00/ EFCX - 1.00/ EFCX0 )
      DELTT = ((EFCX - 1.00) / CDAB - (EFCX0 - 1.00)/CDA0B)/F /V0
      VC = VC0 + DELV
      TC = TC0 + DELTT
      IF( X .LE. XC ) GO TO 3
      XS = X - XC
      CALL VXT(VX , TS ,VC ,WX, F, B ,CD8, XS, CAYD, TOL )
      T = TS + TC
      GO TO 4
3     CONTINUE
      VX = VC
      T = TC
      4     IF( IDUMP .EQ. 1 ) WRITE(6,1000) X , T , X1, VC0 , TC0
      1000 FORMAT( , A2X , CDAB , EFCX , EFCX0 , VC0 , EFCX0
      1001 IS , / 1X , IPI0E12.4 , / , / )
      IF( IDUMP .EQ. 1 ) WRITE(6,1001) DELV , DELTT, VC, TC ,XS,VX
      1001 FORMAT( , DELV , DELTT , VC , TC , XS , VX , / 1PI0E12.4 , / , / )
      RETURN
      END

```

HITS3381
 HITS3382
 HITS3383
 HITS3384
 HITS3385
 HITS3386
 HITS3387
 HITS3388
 HITS3389
 HITS3390
 HITS3391
 HITS3392
 HITS3393
 HITS3394
 HITS3395
 HITS3396
 HITS3397
 HITS3398
 HITS3399
 HITS3400
 HITS3401
 THITS3402
 HITS3403
 HITS3404
 HITS3405
 XHITS3406
 HITS3407
 HITS3408
 HITS3409

D.57 Subroutine TVX

```

SUBROUTINE TVX(VI, T, VX, X)
  IMPLICIT REAL * 8(A-H, O-Z)
  COMMON/OEC/DE(600)
  EQUIVALENCE
    A(ALPTRM), DE( 402)), (ALPHA, DE( 400)), (ATJ, DE( 401)),
    B(BETPM), DE( 405)), (BETA, DE( 403)), (B, DE( 404)),
    C(CETA), DE( 408)), (CETA, DE( 406)), (BETADO, DE( 407)),
    D(CAYD), DE( 411)), (CD8, DE( 409)), (ALPMIN, DE( 410)),
    E(CMTHC), DE( 414)), (CMTHA, DE( 415)), (CAY1, DE( 416)),
    F(CAY2), DE( 417)), (CDAOR, DE( 418)), (CDAB, DE( 419)),
    G(DELTT), DE( 420)), (DELX, DE( 421)), (DELT, DE( 422)),
    H(DELX), DE( 423)), (DYDX, DE( 424)), (DZDX, DE( 425)),
    I(DYDT), DE( 426)), (DZDT, DE( 427)), (DELT, DE( 428)),
    J(DELAM), DE( 429)), (EYEP, DE( 430)), (EMP, DE( 431)),
    K(ELT), DE( 432)), (EDLT, DE( 433)), (F, DE( 434)),
    L(H), DE( 435)), (PSID, DE( 436)), (PHI, DE( 437)),
    M(PH1), DE( 438)), (PH2, DE( 439)), (PSI, DE( 440)),
    N(R1), DE( 441)), (R2, DE( 442)), (R3, DE( 443)),
    O(R4), DE( 444)), (RTRIM, DE( 445)), (TC, DE( 446)),
    P(TC), DE( 447)), (TS, DE( 448)), (TG, DE( 449)),
    Q(TP), DE( 450)), (THETA, DE( 451)), (THETD, DE( 452)),
    R(TOL), DE( 453)), (VC, DE( 454)), (VX, DE( 455)),
    S(VYA), DE( 456)), (VZA, DE( 457)), (WC, DE( 458)),
    EQUIVALENCE
      A(W1), DE( 461)), (W2, DE( 459)), (W, DE( 460)),
      R(XLAM), DE( 464)), (XLAM1, DE( 462)), (WL2, DE( 463)),
      C(XNU1), DE( 467)), (XNU2, DE( 465)), (XLAM2, DE( 466)),
      D(XMU), DE( 470)), (XJAY, DE( 468)), (XG, DE( 469)),
      E(XJA), DE( 473)), (YA, DE( 471)), (XJAZ, DE( 472)),
      F(ALPH), DE( 476)), (Z, DE( 474)), (Z, DE( 475))
  VWX = VI - WX
  CAYBC = CAYD / 8 / CD8
  BCDK = OSQRT( 1.00 / CAYBC )
  TFCKT = DTAN(F* OSQRT(CD8 * CAYC)) * T
  VX = (VI - WX - TFCKT / BCDK) / ( 1.00 + BCDK * VWX * TFCKT) + CAYBC
  X = WX * T - DLOG( ( VX-WX)** 2 + CAYBC ) / ( VI-WX)** 2 + CAYBC
  1 )
  2 ) / 2.00 / F / OSORT(B) / CD8
  IF(IDUMP.EQ.0) GO TO 69
  WRITE(6,1000) X, VX, CAYBC, VWX, BCDK, TFCKT
  69 CONTINUE
  1000 FORMAT(// ' X=', 1P8E15.8)
  RETURN
  FND

```

D.58 Subroutine T4NTV

```

SUBROUTINE T4NTV ( IAW, IAY, LT, KBR )
  IMPLICIT REAL*8 (A-H,O-Z)
  COMMON /CARDCH/ B(200), C(5000), IA(200,2), ID(10,2), INC,
  1 IQPRNT, IQUT, IY, KARSI(3), KA, KB, KC, KD, KT1, K234, LABC(3)
  SWF = 0.000
  SYMF = 0.000
  SYSWF = 0.000
  DO 10 I=1,LT
    K = IAW + I
    SWF = SWF + C(K)
    J = IAY + I
    SYMF = SYMF + C(J) * C(K)
  10 SYSWF = SYSWF + C(J)**2 * C(K)
  C NOMINAL VALUE
  EY = SYMF / SWF
  B(KBR+1) = EY
  C VARIANCE
  B(KBR+3) = SYSWF / SWF - EY**2
  C TOLERANCE
  B(KBR+2) = 3.000 * DSORT ( B(KBR+3) )
  RETURN
END

```

HITS3455
 HITS3456
 HITS3457
 HITS3458
 HITS3459
 HITS3460
 HITS3461
 HITS3462
 HITS3463
 HITS3464
 HITS3465
 HITS3466
 HITS3467
 HITS3468
 HITS3469
 HITS3470
 HITS3471
 HITS3472
 HITS3473
 HITS3474
 HITS3475
 HITS3476

D.59 Subroutine VXT

```

SUBROUTINE VXT(VX,T,VI,WX,F,B,CD8,X,CAYD,TOL)
IMPLICIT REAL*8(A-H,O-Z)
COMMON/DEC/OE(600)
EQUIVALENCE
F2=2.00*F
VIWX=VI-WX
VIWX2=VIWX*VIWX
BCK=8*CD8/CAYD
BCD8KD=DSORT(BCK)
T1=DATAN(BCD8KD*VIWX)
T0=0.00
T=0.00
F28C=F2*DSORT(8)*CD8
VX0=0.00
FSCDKD=F*DSORT(CD8*CAYD)
DO 1 I=1,100
E=FEXP(F28C*(WX*T-X))
VX=DSORT(VIWX2*E+(E-1.00)/BCK)+WX
T=
1 / FSCDKD
IF(IDU=0.0) GO TO 69
6.1000 I,T,TO,VX,VX0,E
69 CONTINUE
1.3,T=,1PIE18.8,T-OLD=,1PIE18.8,VX=,1PIE18
1.9,VX-OLD=,1PIE18.8,E=,1PIE18.9/
IF(DABS(VX-VX0)/VX)+DABS((T-T0)/T).LT.TOL)
1 RETURN
VX0=VX
TO=T
CONTINUE
1 WRITE(6,1001)
1001 FORMAT(,VX AND T ROUTINE EXCEEDED 100 ITERATIONS... //)
RETURN
END

```

HITS3477
HITS3478
HITS3479
HITS3480
HITS3481
HITS3482
HITS3483
HITS3484
HITS3485
HITS3486
HITS3487
HITS3488
HITS3489
HITS3490
HITS3491
HITS3492
HITS3493
HITS3494
HITS3495
HITS3496
HITS3497
HITS3498
HITS3499
HITS3500
HITS3501
HITS3502
HITS3503
HITS3504
HITS3505
HITS3506
HITS3507
HITS3508
HITS3509
HITS3510

D.60 Subroutine WIND

HITS3511
HITS3512
HITS3513
HITS3514
HITS3515
HITS3516
HITS3517

```

SUBROUTINE WIND( VX, T, X, WZ, WX, VO, VZ, Z )
  IMPLICIT REAL * 8(A-H, O-Z)
  VOWX = VO - WX
  VZ = WZ * (1.00 - (VX - WX) / VOWX )
  Z = WZ / VOWX * (VO * T - X )
  RETURN
END

```

D.61 Subroutine XXC

```

SUBROUTINE XXC (XL0, XC, V0, DELL, CAY1, CAY2, TOL, AC)
IMPLICIT REAL * 8(A-H, O-Z)
COMMON/OEC/OE(600)
EQUIVALENCE
  IF( CAY1+ CAY2.NE. 0.) GO TO 2
  XC=1.D-3
  RETURN
2  CONTINUE
  XLV = XL0 / V0
  XCO = DLOG( AC / (CAY1 + CAY2) ) / XLV
  DLV0 = DELL / V0
  DO 1 I = 1, 100
    E = DEXP( DLV0 * XCO )
    E1 = CAY1 * E
    E2 = CAY2 / E
    A = E1 + E2
    A2 = E1 - E2
    B = FEXP( - XCO * XLV )
    C = ( A - AC * B ) / ( A * XLV + DLV0 * A2 )
    XC = XCO - C
    IF( IDUMP.NE.1) GO TO 69
    WRITE(6,1000) I, XC, XCO, C, E, E1, E2, A, A2, B
69  CONTINUE
1000 FORMAT( ' NO.=', I4, ' XC=', IPE18.8, ' XC-OLD =', IPE18.8,
1, ' CHANGE =', IPE18.6 / IPE15.6 )
    IF( DABS( C / XC ) .GT. TOL ) GO TO 5
    IF( XC .LT. 1.D-3) XC=1.D-3
    RETURN
5  CONTINUE
  XCO = XC
1  CONTINUE
  WRITE(6,1001)
1001 FORMAT( ' XXC ROUTINE EXCEEDED 100 ITERATIONS WITHOUT CONVERGING' )
  IF( XC .LT. 1.D-3) XC = 1.D-3
  RETURN
  END

```

D.62 Function ZATAN2

HITS3554
HITS3555
HITS3556
HITS3557
HITS3558
HITS3559
HITS3560
HITS3561
HITS3562

```

1
FUNCTION ZATAN2(R1, R2)
  IMPLICIT REAL * 8(A-H, O-Z)
  IF( R1 .NE. 0.) GO TO 1
  IF( R2 .NE. 0.) GO TO 1
  ZATAN2=0.00
  RETURN
  ZATAN2= DATAN2(R1, R2)
  RETURN
END

```

D.63 Subroutine ZZ

```

FUNCTION ZZ(CNA,CD8,CAYD,V0,EMP,ICNCL)
IMPLICIT REAL * 8 (A-H,O-Z)
IF(ICNCL.EQ.1) GO TO 1
ZZ = CNA * V0/EMP
RETURN
CLA = CNA - (CD8 + CAYD/V0/V0)
ZZ = CLA * V0/EMP
RETURN
END

```

1

```

HITS3563
HITS3564
HITS3565
HITS3566
HITS3567
HITS3568
HITS3569
HITS3570
HITS3571

```

DISTRIBUTION

Copies

A. Department of Defense

Defense Advanced Research Projects Agency
ATTN: Mr. Charles R. Lehner, Jr.
1400 Wilson Blvd.
Arlington, VA 22209

1

Defense Documentation Center
ATTN: TIPDR
Cameron Station
Alexandria, VA 22314

2

B. Department of the Army

HQ DA (DAMA-WSZ, Dr. V. Garber)
Washington, DC 20310

1

HQ DA (DAMA-WSW, LTC Gibson)
Washington, DC 20310

1

Commander
US Army Material Command
ATTN: AMCRD-T, Dr. H. M. El-Bisi
 AMCRD-T, Mr. D. Stefanye
 AMCRD-FW
5001 Eisenhower Avenue
Alexandria, VA 22333

1

1

1

Commander
US Army Missile Command
ATTN: AMSMI-RGT, Mr. Knoblach
 AMSMI-RDF, Mr. Morrow
Huntsville, AL 35801

1

1

Commander
US Army Armament Command
ATTN: AMSAR-RD, Mr. J. Brinkman
 AMSAR-RDT, Mr. L. Johnson
Rock Island, IL 61201

1

1

Commander
US Army Training Doctrine Command
ATTN: ATCD-CM-A
Fort Monroe, VA 23651

1

DISTRIBUTION (Cont'd)

	<u>Copies</u>
Commander US Army Aviation Systems Command ATTN: AMSAV-EVW P. O. Box 209 St. Louis, MO 63166	1
Director US Army Ballistic Research Laboratories ATTN: AMXBR-CA, Dr. Eichelberger Mr. C. M. Frank Mr. A. S. Elder Aberdeen Proving Ground, MD 21005	1 3 1
Director AMXSY-ADE AMSAA ATTN: Mr. John L. Meredith Aberdeen Proving Ground, MD 21005	1
Commander US Army Air Defense Agency Fort Bliss, TX 79916	1
Commandant US Army Armor School Director, Combat Development ATTN: ATSD-CD-MM, Mr. M. Falkovich Fort Knox, KY 40121	1
Commandant US Army Air Defense School ATTN: ATSAD-DL-P Fort Bliss, TX 79916	1
President US Army Air Defense Board ATTN: STEBD-TD-S Fort Bliss, TX 79916	1
Commander Edgewood Arsenal ATTN: SAREA-CL-PA, Mr. A. Flatau Edgewood, MD 21010	1

DISTRIBUTION (Cont'd)

Copies

Commander
Frankford Arsenal
ATTN: SARFA-J700, Mr. C. M. Dickey 3
SARFA-MDS-BP, Mr. P. Ayyoub 1
Mr. A. Korr, Chairman-Invention 1
Evaluation Committee
Building 64-4
Dr. Sid Ross 1
Philadelphia, PA 19137

Commander
Picatinny Arsenal
ATTN: SARPA-FR-E, Mr. Hirskowitz 1
SARPA-AD-D-A-2, Mr. Michelson 1
SARPA-AD, Mr. Painter 1
Dover, NJ 07801

Commander
Rock Island Arsenal
ATTN: SARRI-LA, Mr. Wrenn 1
SARRI-LE 1
SARRI-LA, Mr. Wells 1
SARRI-LW, Mr. Stickling 1
SARRI-LR-Y 15
Rock Island, IL 61201

Commander
Watervliet Arsenal
ATTN: Mr. Austin 1
Mr. Rummel 1
Watervliet, NY 12189

C. Department of the Air Force

AFAL/DLDG (Maj. Bilsbury)
Eglin AFB, FL 32542 1
AFATC/DLM (Mr. Walter Detrick)
Eglin AFB, FL 32542 1

DISTRIBUTION (Concl'd)

	<u>Copies</u>	
AEDC/ARO (Mr. G. Norfleet) Arnold AFS, TN 37389	1	
 D. <u>Department of the Navy</u>		
Commander Naval Surface Weapons Laboratory ATTN: Dr. V. Dawson	1	
Mr. N. Sheetz	1	
White Oak, MD 25989		
 Commander Naval Research Laboratory ATTN: Mr. W. Atkins		1
Washington, DC 20315		
Naval Weapons Center Code 5115 ATTN: Mr. Ward	1	
China Lake, CA 93555		
 E. <u>General</u>		
Avco Systems Division ATTN: Dr. W. G. Reinecke 201 Lowell Street Wilmington, MA 01887	20	