

NASA CR-159043

(NASA-CR-159043) ATLAS, AN INTEGRATED
STRUCTURAL ANALYSIS AND DESIGN SYSTEM.
VOLUME 3: USER'S MANUAL, INPUT AND
EXECUTION DATA (Boeing Commercial Airplane
Co., Seattle) 698 p HC A99/MF A01 CSCL 20K G3/39

N79-31622

Unclas
35733

ATLAS – An Integrated Structural Analysis and Design System

User's Manual – Input and Execution Data

Dr. Rodney L. Dreisbach

Boeing Commercial Airplane Company
Seattle, Washington 98124

Prepared for
Langley Research Center
under contract NAS1-12911

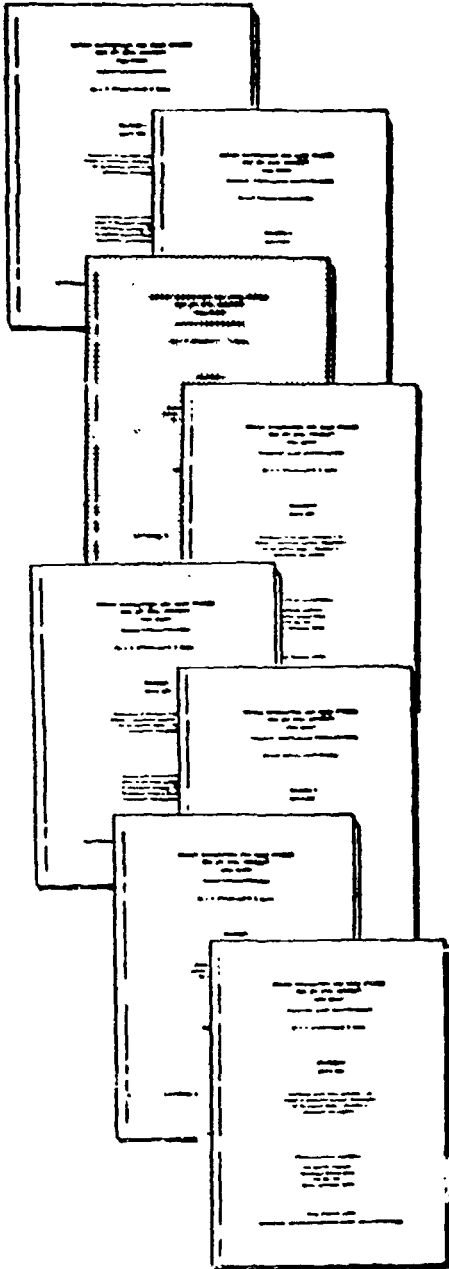


NASA

National Aeronautics and
Space Administration

1979

ATLAS SYSTEM DOCUMENTATION



VOLUME I

ATLAS User's Guide
NASA CR-159041

VOLUME II

System Design Document
NASA CR-159042

VOLUME III

User's Manual-Input and Execution Data
NASA CR-159043

VOLUME IV

Random Access File Catalog
NASA CR-159044

VOLUME V

System Demonstration Problems
NASA CR-159045

VOLUME VI

DESIGN Module Theory
NASA CR-159046

VOLUME VII

LOADS Module Theory
Boeing Commercial Airplane Company
D6-25400-0101

VOLUME VIII

SNARK User's Manual
Boeing Computer Services
BCS-G0686

FOREWORD

Development of the ATLAS integrated structural analysis and design system was initiated by The Boeing Commercial Airplane Company in 1963. Continued development efforts have resulted in the release and application of several extended versions of the system to aerospace and civilian structures. Those capabilities of the current ATLAS version developed under the NASA Langley Contract No. NAS1-12911 include the following: geometry control, thermal stress, fuel generation/management, payload management, loadability curve generation, flutter solution, residual flexibility, strength design of composites, thermal fully stressed design, and interactive graphics. The monitor of this contract was G. L. Giles. The inertia loading capability was developed under the Army Contract No. DAAG46-75-C-0072.

This document is one volume of a series of documents describing the ATLAS System. The remaining documents present details regarding the system design, the data management, the engineering method used by the computational modules, and the system-demonstration problems.

The key responsibilities for development of ATLAS have been within the Integrated Analysis/Design Systems Group of the Structures Research Unit of BCAC and the ATLAS System Group of the BCS Integrated Systems and Systems Technology Unit. R. E. Miller, Jr. was the Program Manager of ATLAS up to 1976 after which K. H. Dickenson assumed this position. The current ATLAS System is the result of the combined efforts of many Boeing engineering and programming personnel. Those who contributed directly to the current version of ATLAS are as follows:

B. F. Backman	H. B. Hansteen	C. D. Mounier
G. N. Bates	B. A. Harrison	F. D. Nelson
L. C. Carpenter	J. M. Held	M. C. Redman
R. E. Clemmons	M. Y. Hirayama	R. A. Samuel
R. L. Dreisbach	J. R. Hogley	M. Tamekuni
W. J. Erickson	H. E. Huffman	G. von Limbach
S. H. Gadre	D. W. Johnson	S. O. Wahlstrom
F. P. Gray	A. S. Kawaguchi	R. A. Woodward
D. W. Halstead		K. K. Yagi

~~ORIGINAL FILED~~ NOT FILMED

ABSTRACT

This manual describes the input data and execution control statements for the ATLAS integrated structural analysis and design system. It is operational on the Control Data Corporation (CDC) 6600/CYBER computers in a batch mode or in a time-shared mode via interactive graphic or text terminals. ATLAS is a modular system of computer codes with common executive and data-base management components. The system provides an extensive set of general-purpose technical programs with analytical capabilities including stiffness, stress, loads, mass, substructuring, strength design, unsteady aerodynamics, vibration, and flutter analyses. The sequence and mode of execution of selected program modules are controlled via a common user-oriented language. Execution of selected modules with external interfaced programs is supported by the ATLAS data base and ATLAS data manager. User interfaces are provided for interactive execution-control, data-file editing, and graphical-display of selected data. Problem-definition input data are written in a problem-oriented language. Input-data generation options and input-data checks provided by the preprocessors minimize the amount of data and flowtime for problem definition/verification. Postprocessors allow selected data to be extracted, manipulated, and displayed via online or offline prints or plots for monitoring and verifying problem solutions.

CONTENTS

		<u>Page</u>
1.0	INTRODUCTION	1.1
 <u>OVERVIEW AND EXECUTION OF THE ATLAS SYSTEM</u>		
10.0	OVERVIEW OF SYSTEM CAPABILITIES AND SYSTEM DESIGN	10.1
10.1	SYSTEM FUNCTIONS AND ANALYSIS CAPABILITIES	10.2
10.2	SYSTEM ARCHITECTURE	10.6
10.2.1	Data-Base Management	10.7
10.3	USER INTERFACES	10.8
11.0	JOB-DECK SETUP AND SYSTEM EXECUTION	11.1
11.1	JOB DECK PREPARATION	11.2
11.1.1	Control Cards for the CDC 6600 .	11.3
11.1.2	ATLAS Control-Program Deck	11.3
11.1.3	ATLAS Problem-Definition Data Deck	11.4
11.2	CDC 6600 JOB CONTROL CARDS	11.7
11.3	ATLAS EXECUTION RESOURCE REQUIREMENTS ...	11.11
11.3.1	Central Memory Requirements	11.12
 <u>ATLAS SYSTEM PREPROCESSOR INPUT DATA</u>		
100.0	INPUT DATA GENERAL INFORMATION	100.1
100.1	DATA DECK	100.2
100.2	INPUT DATA FORMAT	100.4
100.2.1	Data Records	100.4
100.2.2	Data Items	100.5
100.2.3	Data Deck User Comments	100.8
100.3	AUTOMATIC INPUT DATA GENERATION	100.9
100.4	NODES AND ELEMENTS	100.12
100.5	COORDINATE SYSTEMS AND NODAL KINEMATICS	100.13
100.6	UNITS OF MEASUREMENT	100.15
104.0	AF1 AERODYNAMICS DATA	104.1
104.1	INPUT DATA	104.2
104.1.1	Geometry Data Subset	104.2
104.1.2	Modal Data Subset	104.8
104.1.3	Option Data Subset	104.10
106.0	BOUNDARY CONDITION DATA	106.1

106.1	INPUT DATA	106.2
	106.1.1 Support Displacement Data Subset	106.8
112.0	DESIGN DATA	112.1
	112.1 INPUT DATA	112.3
	112.1.1 Table Design Data	112.7
	112.1.2 Stiffness Set-Level Design Data	112.10
	112.1.3 Stage-Level Design Data	112.27
114.0	DETAIL DATA	114.1
	114.1 INPUT DATA	114.2
	114.1.1 Cross-Section Data Subset	114.2
	114.1.2 Spacing Data Subset	114.4
116.0	DUBLAT AERODYNAMICS DATA	116.1
	116.1 INPUT DATA	116.2
	116.1.1 Geometry Data Subset	116.2
	116.1.2 Lifting-Surface Region Subsets	116.10
	116.1.3 Modal Data Subset	116.11
	116.1.4 Option Data Subset	116.14
122.0	FLUTTER DATA	122.1
	122.1 INPUT DATA	122.2
126.0	GEOMETRY DATA	126.1
	126.1 INPUT DATA	126.2
	126.1.1 Control Curve Definitions	126.2
	126.1.2 Component Geometry Input Data	126.11
130.0	INTERACT (SUBSTRUCTURE) DATA	130.1
	130.1 INPUT DATA	130.6
	130.1.1 Substructure Model Definition	130.6
	130.1.2 Boundary-Condition Change-Data Subset	130.10
	130.1.3 Highest-Level Substructure Identification	130.13
	130.1.4 Load Case Selection Data Subset	130.13
134.0	LOADS DATA	134.1
	134.1 INPUT DATA	134.2
	134.1.1 Nodal Load Data Subset	134.4
	134.1.2 Support Displacement Data Subset	134.9
	134.1.3 Element Load Data Subset	134.12
	134.1.4 Nodal Thermal Load Data Subset	134.18
	134.1.5 Element Thermal Load Data Subset	134.19

	134.1.6	Rotational Inertia Load Data Subset	134.21
136.0		MACHBOX AERODYNAMICS DATA	136.1
	136.1	INPUT DATA	136.2
	136.1.1	Geometry Data Subset	136.2
	136.1.2	Modal Data Subset	136.9
138.0		MASS DATA	138.1
	138.1	INPUT DATA	138.3
	138.1.1	Mass-Distribution Condition Data Subset	138.3
	138.1.2	Mass Element Data Subset	138.5
	138.1.3	Fuel Data Subset	138.10
	138.1.4	Payload Data Subset	138.19
	138.1.5	Concentrated Mass Data Subset ..	138.28
	138.1.6	Mass Lumping Data Subset	138.30
	138.1.7	Auxiliary Panel Data Subset	138.31
	138.1.8	Weight-Statement Label Data Subset	138.35
	138.1.9	Factor Data Subset	138.37
140.0		MATERIAL DATA	140.1
	140.1	STANDARD MATERIAL DATA	140.2
	140.2	INPUT DATA	140.4
	140.2.1	Special Material Data	140.4
	140.2.2	Composite Material Data	140.7
146.0		NODAL DATA	146.1
	146.1	INPUT DATA	146.3
	146.1.1	Direct Definition of Nodes	146.6
	146.1.2	Definition of Nodes Based on Geometry Components	146.9
150.0		RHO3 AERODYNAMICS DATA	150.1
	150.1	INPUT DATA	150.2
	150.1.1	Geometry Data Subset	150.2
	150.1.2	Modal Data Subset	150.7
	150.1.3	Option Data Subset	150.9
152.0		STIFFNESS DATA	152.1
	152.1	INPUT DATA	152.2
	152.1.1	Property Data Subset	152.2
	152.1.2	Element Data Subset	152.3
154.0		STRESS (SUPERPOSITION) DATA	154.1
	154.1	INPUT DATA	154.2
	154.1.1	Superposition Load Case Data ...	154.2
	154.1.2	Superposition Displacement and Stress Constraints	154.4

	154.1.3	Superposition Load Case Titles	.154.6
156.0		SUBSET-DEFINITION DATA156.1
	156.1	INPUT DATA156.2
	156.2	SUBSET-DEFINITION RECORDS156.5
	156.2.1	Node Subset Definition Records	.156.5
	156.2.2	Ordered Node Subset Definition Records156.6
	156.2.3	Element Subset Definition Records156.6
	156.2.4	Label Subset Definition Records156.7
	156.2.5	Records for Extraction of Subsets from Other Subsets156.8
	156.2.6	Records for Definition of Sub- sets via Subset Combinations	...156.8
	156.2.7	Subset Modification (Exclusion) Records156.10
	156.3	SUBSET REGION DEFINITIONS156.12
	156.3.1	Right Circular Cylinder of Finite Length156.12
	156.3.2	Finite-Thickness Infinite Slab	.156.13
	156.3.3	Infinite-Length Polygonal Tube	.156.15

EXECUTION DATA FOR TECHNICAL PROCESSORS AND POSTPROCESSORS

200.0		ATLAS SYSTEM EXECUTION200.1
	200.1	ATLAS CONTROL PROGRAM200.2
	200.2	CONTROL-STATEMENT FORMATS200.4
	200.3	ATLAS CONTROL STATEMENTS200.10
	200.3.1	General Control Statements200.10
	200.3.2	Technically-Oriented Control Statements200.17
	200.4	CATALOGED CONTROL PROCEDURES200.18
	200.4.1	Standard Control-Procedures for SET/STAGE Models200.18
	200.4.2	Standard Control-Procedures for Substructure Analyses200.19
	200.4.3	User-Defined Control Procedures200.21
	200.5	INTERACTIVE CONTROL-PROGRAM EXECUTION	...200.23
	200.5.1	ATLAS Interactive Mode200.24
	200.5.2	EDIT Interactive Mode200.24
	200.5.3	EXECUTE Interactive Mode200.25
	200.6	CONTROL-COMMAND FORMATS200.27
	200.6.1	TTY Command Format200.27
	200.6.2	BATCH Command Format200.29
	200.7	ATLAS CONTROL COMMANDS200.30

	200.7.1	ATLAS-Mode Control Commands200.34
202.0		ADDINT MODULES202.1
	202.1	EXECUTE ADDINT STATEMENT202.2
	202.2	PRINT OUTPUT STATEMENT202.6
204.0		AF1 AERODYNAMICS MODULES204.1
	204.1	PRINT INPUT STATEMENT204.2
	204.2	EXECUTE AF1 STATEMENT204.3
	204.3	PRINT OUTPUT STATEMENT204.5
206.0		BOUNDARY CONDITION POSTPROCESSOR206.1
	206.1	PRINT INPUT STATEMENT206.2
		206.1.1 Data-Set Print Option206.2
		206.1.2 Data-Subset Print Option206.3
208.0		BUCKLING MODULES208.1
	208.1	EXECUTE BUCKLING STATEMENT208.3
	208.2	PRINT OUTPUT STATEMENT208.5
210.0		CHOLESKY PROCESSOR210.1
	210.1	EXECUTE CHOLESKY STATEMENT210.2
212.0		DESIGN MODULES212.1
	212.1	PRINT INPUT STATEMENT212.4
	212.2	EXECUTE DESIGN STATEMENT212.5
		212.2.1 Strength Execution Option212.5
		212.2.2 Flutter Execution Option212.8
		212.2.3 History Execution Option212.9
		212.2.4 Composite Optimization Option	..212.10
		212.2.5 Smoothing Execution Option212.12
	212.3	PRINT OUTPUT STATEMENT212.13
		212.3.1 Resized Element-Property Print Option212.13
		212.3.2 Design History Print Option212.14
216.0		DUBLAT AERODYNAMICS MODULES216.1
	216.1	PRINT INPUT STATEMENT216.2
	216.2	EXECUTE DUBLAT STATEMENT216.3
	216.3	PRINT OUTPUT STATEMENT216.6
218.0		EXTRACT POSTPROCESSOR218.1
	218.1	EXECUTE EXTRACT STATEMENT218.3
		218.1.1 Nodal Data for Graphical Display218.4
		218.1.2 Stiffness Data for Graphical Display218.4
		218.1.3 Displacement Data for Graphical Display218.6
		218.1.4 Stress Data for Graphical	

		Display	218.7
	218.1.5	Design Data for Graphical Display	218.7
	218.1.6	Buckling Data for Graphical Display	218.9
	218.1.7	Mass-Element Data for Graphical Display	218.9
	218.1.8	Mass Data for Loadability Diagrams	218.10
	218.1.9	Vibration Data for Graphical Display	218.11
	218.1.10	Flutter Data for Graphical Display	218.12
	218.1.11	Matrix Data for Graphical Display	218.13
	218.1.12	Displacement and Stress Data for Printout	218.13
220.0	FLEXAIR AERODYNAMICS MODULES		220.1
	220.1	EXECUTE FLEXAIR STATEMENT	220.2
	220.2	PRINT OUTPUT STATEMENT	220.7
222.0	FLUTTER MODULES		222.1
	222.1	PRINT INPUT STATEMENT	222.3
	222.2	EXECUTE FLUTTER STATEMENT	222.4
	222.3	PRINT OUTPUT STATEMENT	222.10
224.0	FREEBODY POSTPROCESSOR		224.1
	224.1	PRINT OUTPUT STATEMENT	224.2
228.0	GRAPHICS POSTPROCESSOR		228.1
	228.1	DISPLAY OPTIONS	228.3
		228.1.1 Undeformed Geometry Plots	228.3
		228.1.2 Deformed Geometry Plots	228.4
		228.1.3 Scalar-Grid Plots	228.5
		228.1.4 Contour Plots	228.6
		228.1.5 Graph Plots	228.7
		228.1.6 Matrix Data Plots	228.8
	228.2	ORIENTATION OF ORTHOGRAPHIC PROJECTIONS	228.24
	228.3	MANAGEMENT OF GRAPHICS DATA	228.27
	228.4	EXECUTE GRAPHICS STATEMENT	228.30
		228.4.1 Undeformed Geometry Plots	228.32
		228.4.2 Deformed Geometry and Scalar-Grid Plots	228.36
		228.4.3 Contour Plots	228.38
		228.4.4 Graph Plots	228.39
		228.4.5 Matrix Data Plots	228.42
	228.5	INTERACTIVE GRAPHICS	228.44
		228.5.1 Interactive Dialogue	228.45

228.6	RESTART OF GRAPHICS POSTPROCESSOR	
	EXECUTION	228.56
	228.6.1 Saving Graphics Data	228.56
	228.6.2 Loading Graphics Data	228.57
230.0	INTERACT POSTPROCESSOR	230.1
	230.1 PRINT INPUT STATEMENT	230.2
232.0	INTERPOLATION PROCESSOR	232.1
	232.1 EXECUTE INTERPOLATION STATEMENT	232.6
234.0	LOADS MODULES	234.1
	234.1 PRINT INPUT STATEMENT	234.2
	234.2 EXECUTE LOADS STATEMENT	234.4
	234.3 PRINT OUTPUT STATEMENT	234.6
236.0	MACHBOX AERODYNAMICS MODULES	236.1
	236.1 PRINT INPUT STATEMENT	236.2
	236.2 EXECUTE MACHBOX STATEMENT	236.3
	236.3 PRINT OUTPUT STATEMENT	236.9
238.0	MASS MODULES	238.1
	238.1 PRINT INPUT STATEMENT	238.3
	238.1.1 Data-Set Print Option	238.3
	238.1.2 Data-Subset Print Option	238.4
	238.2 EXECUTE MASS STATEMENT	238.7
	238.3 PRINT OUTPUT STATEMENT	238.9
240.0	MATERIAL POSTPROCESSOR	240.1
	240.1 PRINT INPUT STATEMENT	240.2
242.0	MERGE PROCESSOR	242.1
	242.1 EXECUTE MERGE STATEMENT	242.2
244.0	MULTIPLY PROCESSOR	244.1
	244.1 EXECUTE MULTIPLY STATEMENT	244.2
246.0	NODAL POSTPROCESSOR	246.1
	246.1 PRINT INPUT STATEMENT	246.2
	246.1.1 Data-Set Print Option	246.2
	246.1.2 Data-Subset Print Option	246.2
248.0	REACTION POSTPROCESSOR	248.1
	248.1 PRINT OUTPUT STATEMENT	248.2
250.0	RHO3 AERODYNAMICS MODULES	250.1
	250.1 PRINT INPUT STATEMENT	250.2
	250.2 EXECUTE RHO3 STATEMENT	250.3
	250.3 PRINT OUTPUT STATEMENT	250.5

252.0	STIFFNESS MODULES	252.1
252.1	PRINT INPUT STATEMENT	252.2
	252.1.1 Data-Set Print Option	252.2
	252.1.2 Data-Subset Print Option	252.3
252.2	EXECUTE STIFFNESS STATEMENT	252.6
	252.2.1 Elastic Stiffness and Stress Matrices	252.6
	252.2.2 Geometric Stiffness Matrices ...	252.7
254.0	STRESS MODULES	254.1
254.1	EXECUTE STRESS STATEMENT	254.2
254.2	PRINT OUTPUT STATEMENT	254.6
	254.2.1 Displacement/Stress/Nodal-Force Print Options	254.6
	254.2.2 Composite-Element, Lamina Stress/Strain Print Options	254.10
	254.2.3 Printout of EXTRACTED Displacements and Stresses	254.12
258.0	VIBRATION MODULES	258.1
258.1	EXECUTE VIBRATION STATEMENT	258.4
258.2	PRINT OUTPUT STATEMENT	258.11
APPENDIX A--	ATLAS STANDARD MATERIAL PROPERTY DATA	A.1
APPENDIX B--	ATLAS STIFFNESS FINITE-ELEMENT CATALOG	B.1
APPENDIX C--	ATLAS MASS FINITE-ELEMENT CATALOG	C.1
APPENDIX D--	SUBSTRUCTURE ANALYSIS VIA ATLAS	D.1
APPENDIX E--	ATLAS STANDARD CATALOGED CONTROL PROCEDURES	E.1
	E.1 ATLAS SET/STAGE-STRUCTURE STANDARD CONTROL PROCEDURES	E.2
	E.2 ATLAS SUBSTRUCTURE STANDARD CONTROL PROCEDURES	E.10
APPENDIX F--	SAMPLE ATLAS DECK SETUPS	F.1
	F.1 STRESS AND FLUTTER ANALYSES OF A DELTA WING	F.2
	F.2 STRESS AND VIBRATION SUBSTRUCTURE ANALYSES OF A TRANSMISSION TOWER	F.10
APPENDIX G--	SPECIAL ATLAS-LIBRARY ROUTINES AND ATLAS INTERFACES WITH EXTERNAL PROGRAMS	G.1
	G.1 ATLAS STRUCTURAL-DATA MODIFICATION ROUTINES	G.1
	G.2 NASA-LKC AIRPLANE CONFIGURATION PROGRAM INTERFACE	G.8
	G.3 GCS (GEOMETRY CONTROL SYSTEM) PROGRAM	

	INTERFACE	G.12
G.4	ATLAS/FLEXSTAB SYSTEM INTERFACES	G.14
G.5	ATLAS/NASTRAN SYSTEM INTERFACES	G.35
APPENDIX H--	ATLAS TEXT EDITOR	H.1
H.1	FILE SIZE	H.1
H.2	METHOD OF OPERATION	H.1
H.3	KEYBOARD CONVENTIONS	H.1
H.4	MODES OF OPERATION	H.2
H.5	ABBREVIATING REQUESTS	H.2
H.6	PREFIXING REQUESTS	H.2
H.7	TAB CONTROL	H.2
H.8	MULTIPLE INPUT LINES AND EDIT REQUEST	H.3
H.9	CONTINUATION LINE	H.3
H.10	BLOCK OPERATIONS	H.3
H.11	PREDEFINED REQUEST SEQUENCES	H.3
H.12	ENDS-OF-RECORD	H.3
H.13	ENDS-OF-FILE	H.3
H.14	LENGTH OF TERMINAL LINE	H.3
H.15	REMEMBERED PARAMETERS	H.4
H.16	PARAMETER CONVENTIONS	H.4
H.17	EDIT REQUESTS	H.4
REFERENCES		R.1

FIGURES

	<u>Page</u>
10-1	ATLAS System Modular Design 10.9
11-1	ATLAS Job Deck 11.2
100-1	Kinematic and [Force] Labels and Conventions100.14
104-1	Sample AF1 Aerodynamic Model104.4
116-1	Sample DUBLAT Aerodynamic Model116.5
126-1	Reference Frame and Control Curves for Detailed Geometry Components126.4
126-2	Reference Frame and Control Curves for Simplified Geometry Components126.5
126-3	Example Control-Curve-Definition and Curve-Segment Constraint Records126.10
136-1	Mach Box Aerodynamic Grid136.4
138-1	TYPE1 and TYPE2 Geometric Shapes138.13
146-1	Nodal Reference Frame Coordinates146.6
146-2	SURFACE Node Generation on an Open Section Contour146.16
146-3	SURFACE and MIDSURFACE Node Generation on a Closed Section Contour146.17
150-1	Sample RHO3 Aerodynamic Model150.3
228-1	Point Plot of Nodes228.8
228-2	Point Plot of Mid-Surface Nodes228.9
228-3	Stiffness Element Grid with Node Numbers228.10
228-4	Stiffness Element Grid with Element Numbers228.11
228-5	Stiffness Element Grid with Element-Type Numbers228.12

228-6	Exploded Geometry Plot	228.13
228-7	Deformed Geometry Plot--Points Plus Vectors	228.14
228-8	Deformed Geometry Plot--VECTOR3 Option	228.14
228-9	Deformed Geometry Plot--VECTOR4 Option	228.14
228-10	Deformed Geometry Plot--VECTOR2 Option	228.15
228-11	Deformed Geometry Plot--VECTOR1 Option	228.15
228-12	Scalar-Grid Plot	228.16
228-13	Contour Plot--Rectangular Mesh	228.17
228-14	Contour Plot--Triangular Mesh	228.18
228-15	Graph Plot--Stress Vs. Loadcases	228.19
228-16	Flutter V-g Graph	228.20
228-17	Flutter V-f Graph	228.21
228-18	Loadability Diagram	228.22
228-19	Matrix Data Plot	228.23
228-20	GLOBAL/Display Axes	228.25
228-21	Model-Orientation Under Sequential Rotations	228.26
228-22	Equivalent Rotations for a VX-VY-VZ Orientation ..	228.26
228-23	Global-Axis View Orientations	228.26
228-24	Management of Graphics-Related Data	228.29
228-25	Center of Gravity Vs. Gross Weight Loadability Diagram	228.42
228-26	Function Menu Options	228.50
228-27	Typical GNOME Directory	228.51
228-28	Typical Plot ID Directory	228.51
228-29	Plot Transformation Menu	228.52

228-30	Example Use of the Plot Transformation Menu	228.52
228-31	Zoomed Online Display	228.55
232-1	Motion-Axis Definition	232.3
F-1	Delta-Wing Structural and Airload Grids	F.6
F-2	Plot of Delta-Wing Element Data	F.7
F-3	Flutter V-g Plot for Delta-Wing	F.8
F-4	Flutter V-f Plot for Delta-Wing	F.9
F-5	Transmission Tower Structural Model	F.18
F-6	Substructure Models	F.19
F-7	Exploded View of Substructure 1	F.20
G-1	ATLAS Job Deck with LRCGEOM Data	G.10
G-2	ATLAS/FLEXSTAB Job Deck	G.15
G-3	FLEXSTAB/ATLAS Job Deck	G.22
G-4	ATLAS Job Deck with VAMAT and VAMSCN Data	G.25

TABLES

		<u>Page</u>
10-1	ATLAS Preprocessors, Processors and Postprocessors	10.10
11-1	Processor/Postprocessor Executions for Standard Types of Analyses	11.5
11-2	Problem-Definition Input Data for Standard Types of Analyses	11.6
11-3	Computer Central Memory Requirements of the ATLAS Modules	11.14
11-4	Numeric Conversion Table	11.16
104-1	Summary of AF1 Data Records	104.17
106-1	Summary of Boundary Condition Data Records	106.14
112-1	Design Data Preprocessing Modes	112.6
112-2	Input Variations of Element Design-Stress Properties	112.12
112-3	Fixed, Lower-Bound and Upper-Bound Element Property Data	112.14
112-4	Input Variations of Element-Property Margins of Safety	112.16
112-5	Correspondence of Element Stresses and Element Allowable Stresses	112.19
112-6	Input Variations of Interaction-Expression Parameters	112.26
112-7	Summary of Design Data Records	112.31
114-1	Input Variations of Properties for Stiffness-Element Cross-Section Concepts	114.6
114-2	Input Variations of Element Support-Spacing Properties	114.8
114-3	Summary of DETAIL Data Records	114.9

115-1	Summary of DUBLAT Data Records	116.18
122-1	Summary of Flutter Data Records	122.8
126-1	Summary of Geometry Data Records	126.23
130-1	Summary of Interact Data Records	130.16
134-1	Input Variations of Nodal Load Intensities	134.15
134-2	Input Variations of Element Nodal-Temperature Increments	134.23
134-3	Summary of Loads Data Records	134.24
136-1	Summary of MACHBOX Data Records	136.11
138-1	Input Variations of Mass Element Data	138.9
138-2	Summary of Mass Data Records	138.41
140-1	Summary of Material Data Records	140.10
146-1	Summary of Nodal Data Records	146.18
150-1	Summary of RHO3 Data Records	150.14
152-1	Input Variations of Stiffness-Element Properties..	152.9
152-2	Summary of Stiffness Data Records	152.11
154-1	Summary of Stress Data Records	154.7
156-1	Directory of ATLAS Data-Component Labels and Standard Label-Subset (LSUB) Names	156.18
156-2	Summary of ATLAS Subset-Definition Records	156.26
200-1	Summary of ATLAS Control Statements	200.11
200-2	Summary of ATLAS Interactive-Control Commands	200.36
218-1	Data Displays Associated with Data Extraction for the ATLAS Standard Label Subsets	218.2
218-2	Execution Parameters Associated with Extraction of Data Values for the ATLAS Standard Label Subsets	218.15
228-1	Technical-Data Plot Types	228.2

228-2	Summary of Execution Parameters for the EXECUTE GRAPHICS Statement	228.43
228-3	Applicable FB,N Commands When a Menu or Graphical Display is Activated	228.53
228-4	Plot Transformations Effected Via FB,N Commands and Crosshairs	228.54
G-1	Correspondence of LRCGEOM Data to the Converted, Equivalent, ATLAS Geometry Components	G.9
G-2	ATLAS Input Data and Processor Executions Required for Conversion of Data to NASTRAN Bulk Data	G.38
G-3	Description of ATLAS to NASTRAN Data Conversion...	G.44
G-4	Description of NASTRAN to ATLAS Data Conversion...	G.46

1.0 INTRODUCTION

ATLAS is an integrated structural analysis and design system operational on the Control Data Corporation (CDC) 6600/CYBER computers in a batch mode or in a time-shared mode via interactive text or graphic terminals. It is a modular system of computer codes with common executive and data-base management components. ATLAS provides an extensive set of general-purpose technical programs with aeroelastic analytical capabilities including stiffness, stress, loads, mass, substructuring, strength design, unsteady aerodynamics, vibration and flutter analyses. A finite-element structural-analysis approach is used wherein the distributed physical properties of the problem are represented by a finite number of idealized elements.

The sequence and mode of execution of selected program modules for a multi-technology problem are controlled by the user via a common user-oriented language. User interfaces are provided for interactive control of module executions, data-file editing and graphical-display of selected input and calculated data. Communication of data between program modules is performed automatically by the ATLAS data-base manager.

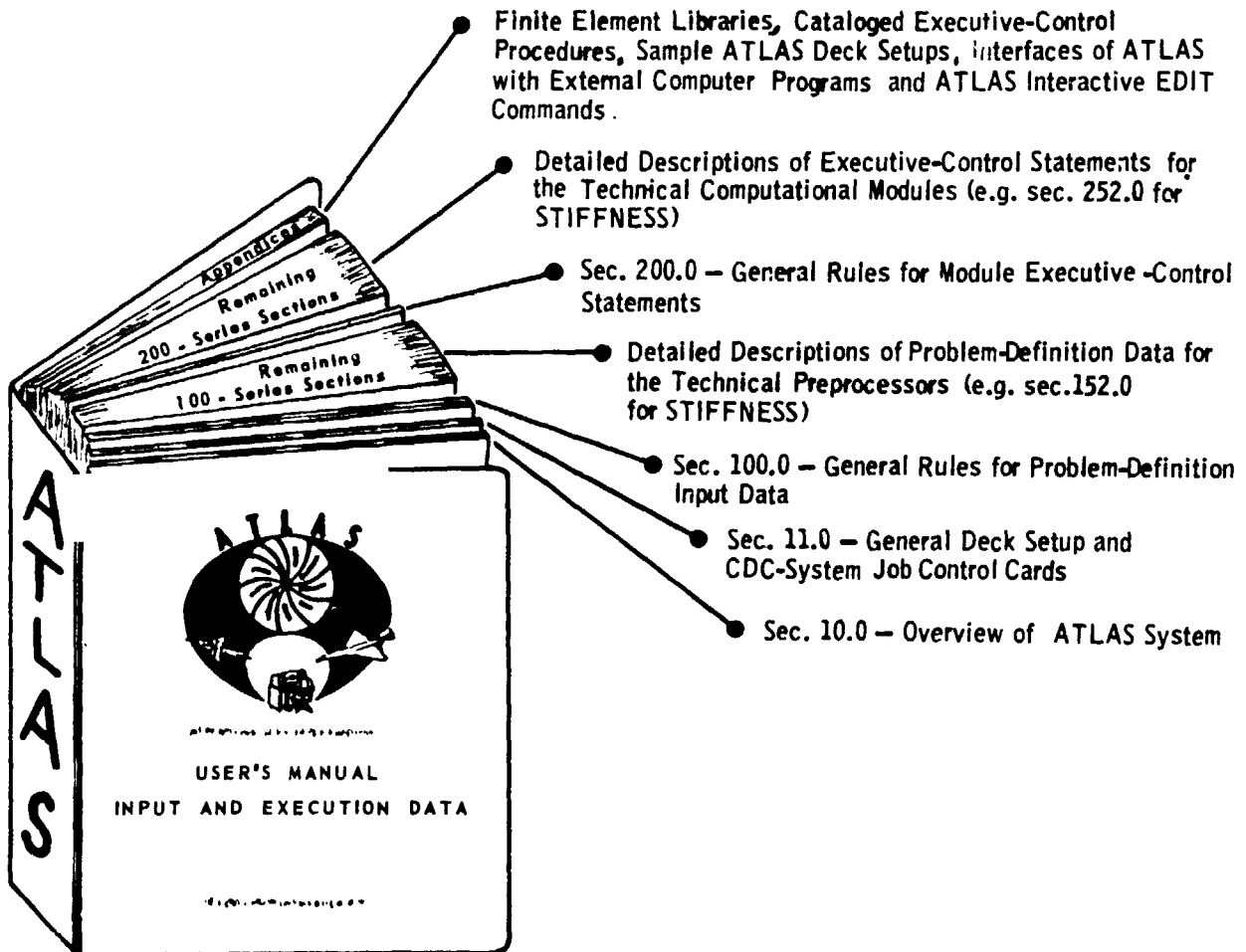
Problem-definition input data are written in a problem-oriented language. Versatile input-data generation capabilities and input-data checks are provided by the preprocessors to minimize the amount of data and flowtime required to define and verify the problem to be solved. Postprocessors allow selected input data and analysis/design results to be extracted, manipulated and displayed via on-line or off-line prints or plots for monitoring and verifying problem solutions.

This document presents detailed descriptions of all problem-definition input data and executive-control statements for the ATLAS System. It is assumed that the reader is familiar with matrix structural-analysis methods and as such, no attempt is made to discuss structural idealization techniques herein. Documentation of the system design and the data-base management system are contained in references 1-1, 1-2 and 1-3.

This document is comprised of four major portions which describe the following:

- a) Section 10.0 provides an overview of the analytical capabilities and architecture of the ATLAS System.

- b) Section 11.0 describes the general deck setup and the CDC-system job control cards required for execution of ATLAS.
- c) Section 100.0 presents the general rules and document notation concerning problem-definition input data. Detailed descriptions of the input data associated with each of the ATLAS System technical-data preprocessors are presented in the remaining 100-series sections.
- d) Section 200.0 presents the general rules and document notation concerning execution-control of the ATLAS System modules. Detailed descriptions of the executive-control statements associated with each of the technical computational modules are presented in the remaining 200-series sections.



10.0 OVERVIEW OF SYSTEM CAPABILITIES AND SYSTEM DESIGN

This section presents an overview of the ATLAS System functions, the system architecture, the user interfaces, and the management of data matrices. An overview of the technical capabilities provided by the system is also presented. Detailed descriptions of the system-design and data-base management are documented in references 1-1 and 1-2.

REPRODUCED FROM FILM NOT FILMED

10.1 SYSTEM FUNCTIONS AND ANALYSIS CAPABILITIES

ATLAS provides the user with three major types of capabilities:

- a) Analysis Control
- b) Data Communication
- c) Numerical Computations

The user has complete command of the analysis control. This function is defined via his executive control deck. Data communication within the system is handled automatically. The user, however, is required to name certain analysis data (User Matrices) that describe the total problem. For example, the structural stiffness, loads and displacement matrices are identified by user-assigned names. This is part of the analysis control function as defined by the user.

The computational functions provided by ATLAS are divided into three categories:

- a) Input Data Preprocessing
- b) Data Processing
- c) Output Data Postprocessing

All computational functions are selected by the user via his control deck.

The technical-analysis computational capabilities provided by ATLAS are summarized in the following outline.

Data Management and Execution Control

- A data-base manager
- A user-oriented execution-control language
- Catalogs of control-statement procedures
- Interactive, on-line and batch-mode processing
- Interactive data-file and control-file editor
- Automated execution-restart procedures
- Numerous problem conditioning-checks during analyses

Utility Matrix Algebra

- A user-oriented language for performing matrix and scalar mathematics
- A module for assembling elemental stiffness, mass, loads, and displacement matrices
- A module for solving out-of-core systems of linear symmetric equations

- A module for adding and multiplying out-of-core matrices

Data Preprocessing

- Free-field, input-data format
- Problem-definition data written in a problem-oriented language; many data default values are provided
- Common, automatic data-generation options for all data types
- Automatic generation of nodes, element grids, loads, etc.
- Extensive number of warning and error diagnostics

Data Postprocessing

- Extract selected subsets of data for print/plot displays
- User-selected data printouts
- User-selected on line and off line plots
 - Orthographic, pictorial geometry plots
 - Data displays superimposed on element grids
 - Isocurve contour plots
 - X-Y graphs
- Interactive graphics

Geometry Modeling

- Lofting of three-dimensional structural components
- Maximum of 4095 nodes per data set or substructure
- Local rectangular, cylindrical, and spherical reference frames for node definition and structural response.

Boundary Conditions (BC)

- Constraints on selected nodal freedoms
- Symmetric and antisymmetric options
- Elastic supports
- Maximum of 10 different BC stages

Structural Modeling

- Maximum of 32,767 elements
- Nodal freedoms with no stiffness are automatically ignored
- Library of elements
 - Axial ROD and general BEAM
 - Membrane PLATE and bending GPLATE elements with orthotropic capability
 - Built-up SPAR and COVER elements
 - Family of 3-D isoparametric BRICK elements
 - Grounded SCALAR spring (elastic support)
 - Shear rod (SROD) and shear panel (SPLATE)

-Composite-material plate (CPLATE) and
built-up composite COUPLER elements

Multilevel Substructuring

- No limit on number of interact levels
- Automatic management of substructure-interact data
- Capability for stress, mass, and vibration analyses

Applied Static Loads

- Nodal loads
- Element distributed loads
- Thermal loading
- Rotational inertia loads
- Specified displacements
- Loadcase superposition
- No limit on number of loadcases

Linear Stress Analysis

- Displacement formulation
- Superposition of displacements and stresses
- Freebody--internal nodal forces on elements
- Equilibrium checks
- Stress contour plots

Bifurcation Buckling Analysis

- Maximum of 400 degrees of freedom
- Geometric stiffnesses for ROD, BEAM, PLATE, GPLATE,
and BRICK elements
- Mode-shape plots

Strength Resizing

- Fully-stressed design
 - Panel buckling interaction
 - Geometric and margin-of-safety constraints
 - Thermal effects
- Smoothing of resized element properties
- Margin-of-safety plots

Structural Optimization

- Regional optimization of composite structures

Mass Analysis

- Maximum of 100 mass/weight distribution conditions

- Maximum of 32,767 nonstructural mass elements
- Library of elements
 - RCD and general BEAM
 - Built-up SPAR and COVER elements
 - PLATE elements
 - Concentrated SCALAR masses
- Detailed weight statements
- Diagonal, nondiagonal, and Guyan-reduced mass matrices
- Panel-weight matrices
- Fuel and payload management
- Fuel and payload loadability diagrams

Vibration Analysis

- Normal modes and frequencies
- Generalized mass and stiffness matrices
- Maximum of 400 degrees of freedom
- Mode-shape plots

Flutter Analysis

- Assumed-pressure function and doublet-lattice methods for subsonic compressible flow
- Strip-theory method for subsonic incompressible flow
- Mach-box method for supersonic flow
- Residual structural-flexibility effects
- Mode interpolation functions
 - Surface spline
 - Motion axis and motion point
 - Polynomial
 - Beam spline
- Automated V-g and "matched point" solution options
- V-g and V-f flutter-solution plots

ATLAS Interfaces with External Programs

- ATLAS/FLEXSTAB--Interface ATLAS structural and mass data with FLEXSTAB for performing aeroelastic and elastic stability analyses; Interface FLEXSTAB steady-state loads with ATLAS for performing stress analysis and structural-design functions.
- ATLAS/NASTRAN--Interface ATLAS input data to NASTRAN and interface NASTRAN input data to ATLAS
- Interface geometry input data for the NASA-LRC aerodynamic configuration program with ATLAS
- Interface geometry data generated by the Boeing GCS program with ATLAS.

10.2 SYSTEM ARCHITECTURE

The ATLAS System has been designed primarily on the basis of convenient, user-selection of system functions. Other considerations such as system maintainability and the ability to extend the system capabilities also influenced its design. The architecture developed for ATLAS is a modular system of computational modules with common executive and data-base management components. The executive component is comprised of a system-execution monitor and an analysis-control module (Control Program) that is created from the user-defined control deck. The system architecture is illustrated in figure 10-1.

The three types of computational modules in the system and their basic functions are as follows:

- a) Preprocessors --Read, decode, generate and interrogate the problem definition data; load problem-execution restart data.
- b) Processors --Perform technical, numerical computations.
- c) Postprocessors --Extract, format and display (print/plot) input data and analysis results; save problem-execution restart data.

Table 10-1 contains a summary of all computational modules in the system. It should be noted that for convenient referencing, the 100-series sections of this document are ordered alphabetically according to the names of the Preprocessors. The 200-series sections are also ordered alphabetically according to the names of the Processors and Postprocessors. The input data for the STIFFNESS Preprocessor, for example, are described in section 152.0, whereas the execution-control directives for the STIFFNESS Processor and Postprocessor are described in section 252.0.

The remaining modules of the system are called executive modules. These modules support the analysis control and data communication functions. The executive modules and their basic functions are as follows:

- a) Precompilers --Support the system data-base management and the creation of Control Program modules from control decks; translate user-oriented executive-language

directives into machine instructions.

- b) Control Program --Control the sequence and mode of execution of selected computational modules as specified by the user via the control deck.
- c) ATLAS 0.0 Overlay --Monitor an ATLAS job during batch-mode or interactive execution; interpret control directives input via a time-shared terminal during interactive processing.

10.2.1 Data-Base Management

Automatic transmission of data from one module to another is accomplished primarily by the use of named, random-access disk files. Each data file is identified by a name consisting of seven characters wherein the last three characters are RNF. All input data formatted by the preprocessors are stored in the file named DATARNF, whereas the data generated by a processor are stored in a separate file that is named by use of the first four letters in the processor name. For example, data generated by the STIFFNESS Processor are stored in STIFRNF (see table 10-1). Each matrix within a file is identified by a unique name comprised of 1 to 7 characters which serves as the random-access, index name for data management. Detailed descriptions of all data files and matrices associated with the ATLAS System are presented in reference 1-2.

Management of all internal data is performed automatically. Management of execution-restart data and naming of User Matrices, however, are responsibilities of the user as defined via the control deck. A User Matrix is a set of data such as the structural stiffness matrix, a total mass matrix, etc., that describes the total problem. Certain naming and uniqueness requirements exist for User Matrices referenced by some of the computational modules. These requirements are defined in section 200.C.

10.3 USER INTERFACES

User interfaces with ATLAS are defined via the problem data deck and the control deck. The problem deck defines the problem to be analyzed by the computational functions, whereas the control deck defines the analysis control function.

The problem data deck, which normally represents the bulk of the input data, is described in section 100.0.

The directives supplied via the control deck define the following:

- a) Sequence of computations,
- b) User Matrix names,
- c) Disposition of analysis results,
- d) Scheduled problem-execution restarts,
- e) Contingencies in case of problem data errors.

The control deck may also be used to perform special analytical computations, manipulate ATLAS data, and manage data for interfacing ATLAS computational modules with computer programs that are external to ATLAS. Established interfaces between ATLAS and external programs are described in appendix G.

Execution of a Control Program and the ATLAS System may be performed in a batch mode or in a time-shared mode via interactive text or graphics terminals. Execution-control statements may be input via a terminal during interactive processing of a Control Program. Additionally, on-line plots of selected data may be created on interactive graphics consoles via input to the common, interactive (conversational) GRAPHICS Postprocessor as described in section 228.0.

Requirements of a control program deck and detailed descriptions of the user-defined control statements are presented in section 200.0.

OVERVIEW

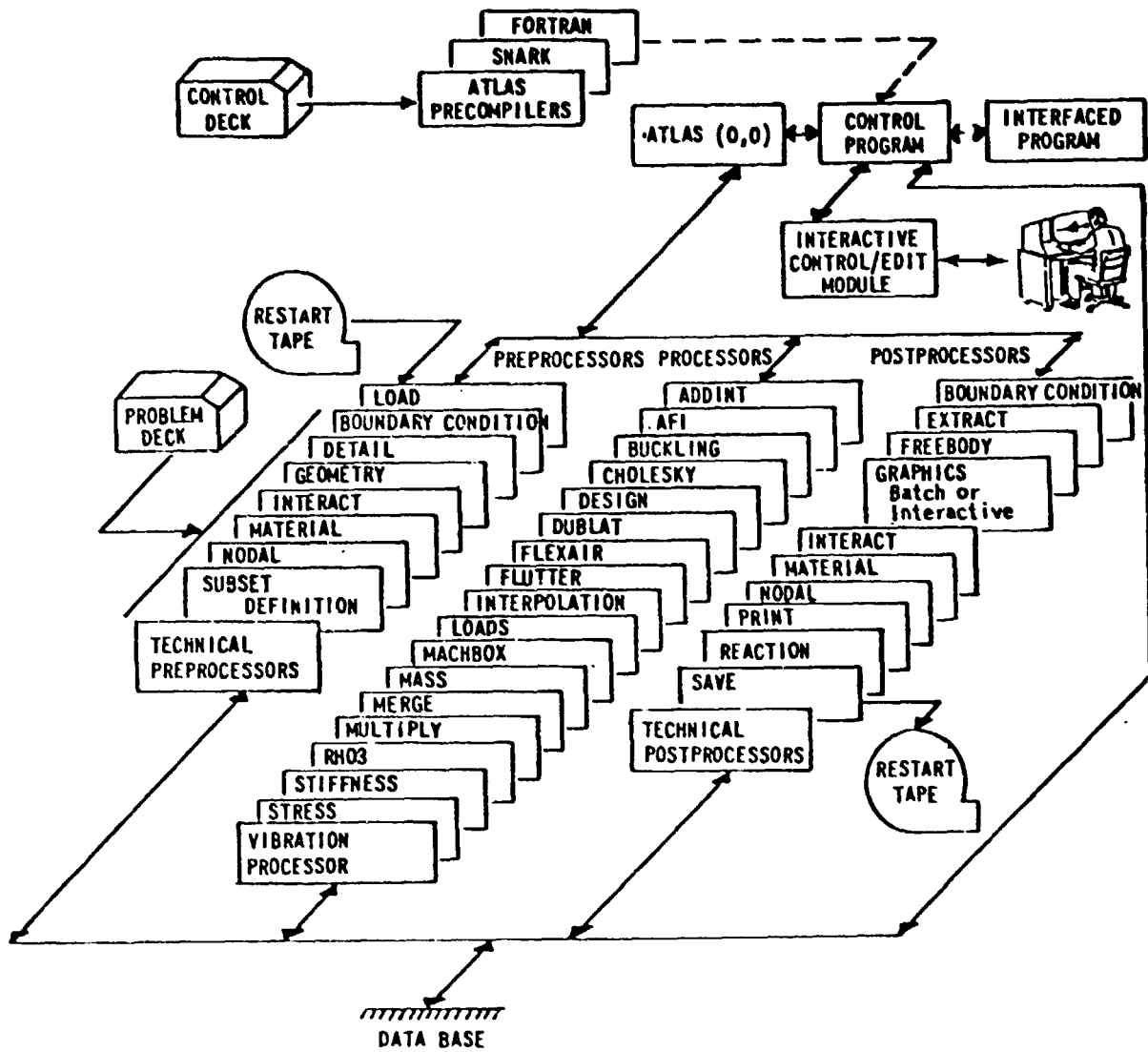


Figure 10-1. ATLAS System Modular Design

Table 10-1. ATLAS Preprocessors, Processors and Postprocessors

Module Name	Technical Function	Document Section Number		
		Preprocessor	Processor	Postprocessor
ADDINT	Add and/or interpolate with respect to reduced frequencies those generalized airforce matrices generated by AF1, DUBLAT, FLEXAIR, MACHBOX, RH03, or by previous execution of ADDIN		202	202
AF1	Define aerodynamic model. Calculate subsonic, incompressible-flow aerodynamic loads for FLUTTER. Strip theory method	104	204	204
BOUNDARY CONDITION	Define boundary conditions for structural model	108		208
BUCKLING	Calculate bifurcation buckling loads and mode shapes		208	208
CHOLESKY	Solve systems of linear symmetric equations		210	
DESIGN	Perform regional optimization of composite structures. Resize structural models based on a fully-stressed design, thermal and local-buckling effects, and geometric and margin-of-safety constraints	112	212	212
DETAIL	Define finite-element cross-section shapes and stiffener locations for plate-element local buckling	114		
DUBLAT	Define aerodynamic model. Calculate subsonic compressible-flow aerodynamic loads for FLUTTER. Doublet-lattice method	118	218	218
EXTRACT	Extract selected problem-definition and analysis data from the primary ATLAS data-base for GRAPHICS			218
FLEXAIR	Calculate generalized airforce matrices that include flexibility effects of truncated structural modes		220	220
FLUTTER	Define structural damping. Modify and solve the flutter equations	122	222	222
FREEBODY	Print internal nodal forces acting on selected finite elements			224
GEOMETRY	Define, generate, and interrogate three-dimensional geometry components for structural and mass models	128		
GRAPHICS	Generate on-line and off-line plots of data selected via EXTRACT. Main and interactive-graphics modes of execution			228
INTERACT	Define substructure interaction problem	130		230
INTERPOLATION	Define mode shape interpolation function for AF1, DUBLAT, FLEXAIR, MACHBOX and RH03		232	
LOAD	Load previously-generated problem data from SAVE to restart execution	200		
LOADS	Define static loads. Calculate nodal loads due to inertial forces, pressure gradients, and thermal gradients. Cumulate static loads and specified displacements	134	234	234
MACHBOX	Define aerodynamic model. Calculate supersonic-flow aerodynamic loads for FLUTTER, Mach Box method	136	236	236
MASS	Define mass model that may complement STIFFNESS. Define fuel and payload management data, generate mass matrices and detailed weight statements for primary and secondary structure, fuel, and payload	138	238	238
MATERIAL	Define finite-element material property data and design allowable stresses for STIFFNESS, STRESS, MASS, and NODAL	140		240
MERGE	Assemble stiffness, mass, loads, and displacement matrices		242	
MULTIPLY	Add and multiply matrices		244	
NODAL	Define node data for STIFFNESS and MASS	146		246
PRINT	Print system-generated matrices			200
REACTION	Print reaction forces and force-equilibrium checks for structural model			248
RH03	Define aerodynamic model. Calculate subsonic compressible-flow aerodynamic loads for FLUTTER. Assumed pressure modes method	150	250	250
SAVE	Save problem data for LOAD to restart execution			200
STIFFNESS	Define structural model. Generate elastic-stiffness, geometric-stiffness, and stress matrices for elements	152	252	252
STRESS	Assemble nodal displacements. Calculate element stresses and internal nodal forces. Superimpose displacements and stresses	154	254	254
SUBSET DEFINITION	Define subsets of nodes and elements in STIFFNESS and MASS models. Define subsets of data-component labels for EXTRACT	156		
VIBRATION	Calculate natural vibration frequencies and mode shapes in addition to generalized mass and stiffness matrices		258	258

11.0 JOB-DECK SETUP AND SYSTEM EXECUTION

The general deck-setup requirements and the CDC 6600 job-control cards required for execution of the ATLAS System are presented herein. Additionally, the resource requirements for execution of ATLAS are discussed in section 11.3.

11.1 JOB DECK PREPARATION

The complete deck setup required for execution of the ATLAS System is shown in figure 11-1. Although many modes of operation may be used to assemble a job deck, only card-input of a deck is illustrated. Disk files, for example, may be used to supply the ATLAS Control Program and problem-definition data deck. If the Control Program is retrieved from a disk file in which it was previously established, the Control-Program source deck and its end-of-record card as shown in figure 11-1 would not be included in the job deck.

The primary components of an ATLAS job deck are:

- a) Control cards for the CDC 6600
- b) ATLAS Control-Program deck
- c) ATLAS problem-definition data deck

Each of these components is described in the following sections.

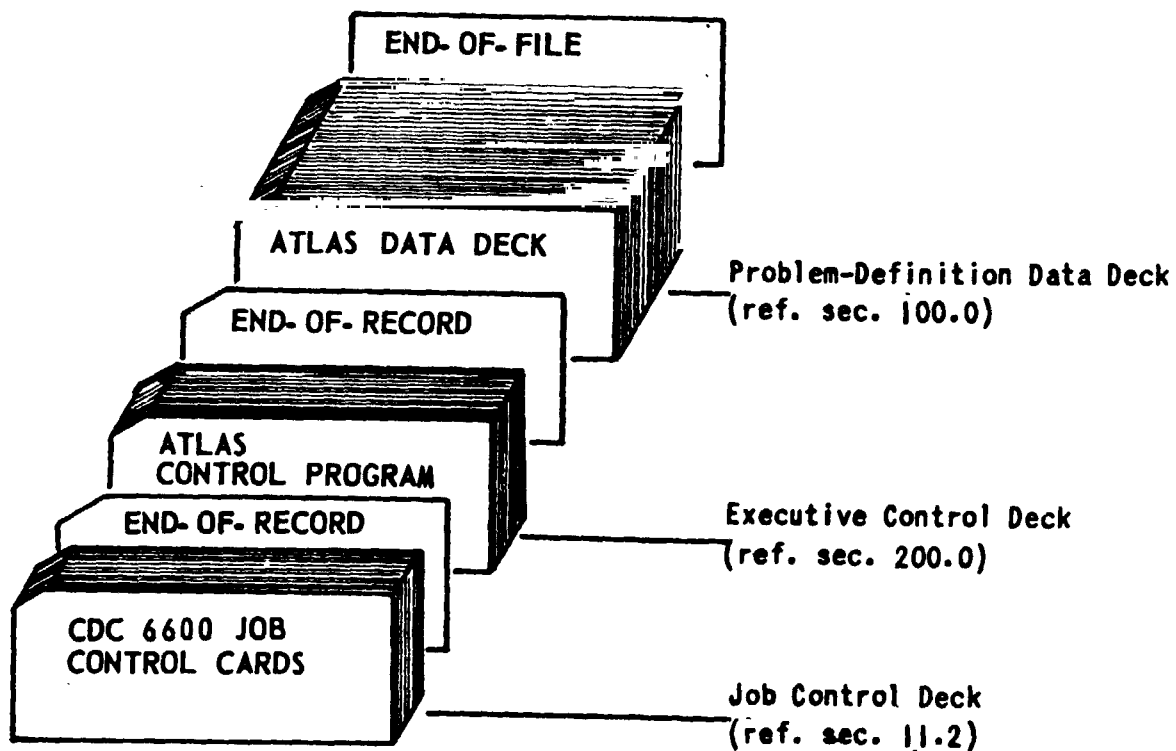


Figure 11-1. ATLAS Job Deck

11.1.1 Control Cards for the CDC 6600

For many jobs, the only CDC 6600 control cards required for execution of ATLAS are shown in the following list:

```

Job Card
Account Card
GET(CATM/UN=UATLASU)
CALL(CATM,ATTACH)
CALL(CATM,CONTROL)
ATLAS(PL=100000)

```

Each control card statement begins in column one of a card. These control cards and the control cards required to execute most types of ATLAS jobs are described in detail in section 11.2.

11.1.2 ATLAS Control-Program Deck

The primary function of the ATLAS Control-Program deck is to define the sequence and mode of execution of selected ATLAS-system modules to analyze the problem established by the data deck. Each execution-directive statement used to create a Control Program is written in a user-oriented language. Detailed descriptions of all Control-Program statements are presented in section 200.0. The remaining 200-series sections of this document describe the control directives and parameters therein that are oriented toward particular technologies. These sections are ordered alphabetically according to the Processor and Postprocessor names for convenient referencing.

The following example Control Program to perform a stress analysis illustrates the use of some of the ATLAS control statements.

```

BEGIN CONTROL PROGRAM
PROBLEM ID (TYPICAL FORMAT OF A CONTROL PROGRAM)
READ INPUT
PRINT INPUT (NODAL)
PRINT INPUT (STIFFNESS)
PERFORM STRESS
PRINT OUTPUT (STRESSES)
END CONTROL PROGRAM

```

Each control statement initiates one or more execution steps as required in the analysis of the problem defined by the data deck. These statements begin in column seven of a card.

Those Processors which must be executed via Control Program statements to perform standard types of analyses using the ATLAS

System are identified in table 11-1. Processors which may be executed in support of typical analyses are also shown in table 11-1. Because of the many analysis options provided by ATLAS, the sequence of execution of these modules is best illustrated by the sample job decks presented in appendix F. In all cases, execution of the Postprocessors to generate print/plot displays is optional.

11.1.3 ATLAS Problem-Definition Data Deck

The mathematical model of the problem to be analyzed by execution of the ATLAS System is defined via the problem-definition data deck. All input data are written in a problem-oriented language. All data items may be input in a free-field format using all columns of a card. The primary component of a data deck is referenced as an input data set (a block of information). Each data set is associated with one of the Preprocessors identified by table 10-1.

The structure of a data deck and general information applicable to all input data are presented in section 100.0. Detailed descriptions of each input data set are presented in the remaining 100-series sections of this document. These sections are ordered alphabetically according to the data-set names (Preprocessor names) for convenient referencing.

Only those data sets that are required to define the problem to be analyzed need be included in the data deck. Those data sets which must be input, and those data sets which are optional for standard types of analyses are identified in table 11-2. The general stacking sequence of the data sets within a data deck is also shown in table 11-2.

Table 11-1. Processor/Postprocessor Executions for Standard Types of Analyses ①

Processors and Postprocessors		Module Executions for Standard Analyses			
Name	Ref. Sec.	Stress	Weights	Vibration	Flutter
ADDINT	202				✓
AFI	204				*
BOUNDARY CONDITION	206 †	*	*	*	*
BUCKLING	208	*			
CHOLESKY	210	✓		✓	✓
DESIGN	212	*			*
DJBLAT	216				*
EXTRACT	218 †	*	*	*	*
FLEXAIR	220				*
FLUTTER	222				✓
FREEBODY	224 †	*			
GRAPHICS	228 †	*	*	*	*
INTERACT	230 †	*	*	*	*
INTERPOLATION	232				✓
LOADS	234	✓			
MACHBOX	236				*
MASS	238		✓	✓	✓
MATERIAL	240 †	*	*	*	*
MERGE	242	✓		✓	✓
MULTIPLY	244	✓		✓	✓
NODAL	246 †	*	*	*	*
REACTION	248 †	*			
RHO3	250				*
STIFFNESS	252	✓	*	✓	✓
STRESS	254	✓			
VIBRATION	258			✓	*

- ①
- A cross (†) identifies a Postprocessor module only
 - A check (✓) identifies a Processor that must be executed for the corresponding type of analysis.
 - An asterisk (*) identifies a Processor or Postprocessor that may be used to perform the corresponding analysis type.
 - Execution of all Postprocessors for print/plot displays is selected by the user as required.
 - All modules shown in the table or any combination thereof may be executed via an ATLAS job. The corresponding input data sets (ref. Table 11-2), however, must be provided as necessary.

Table 11-2. Problem-Definition Input Data for Standard Types of Analyses.

Typical Stacking Sequence of Input Data Sets		Required and Optional Input Data Sets ①							
		Stress Analysis		Weights Analysis		Vibration Anal.		Flutter Analysis	
Name	Ref. Sec.	Required	Optional	Required	Optional	Required	Optional	Required	Optional
GEOMETRY	126		*		*		*		*
NODAL	146	1	✓	1	✓	1	✓	1	✓
BOUNDARY CONDITION	106	2	✓		*	2	✓	2	✓
MATERIAL	140		*		*		*		*
STIFFNESS	152	3	✓		*	3	✓	3	✓
MASS	138			2	✓	4	✓	4	✓
SUBSET DEFINITION	156		*		*		*	5	✓
DETAIL	114		*						
INTERACT	130		*		*		*		*
LOADS	134	4	✓						
STRESS	154		*						
DESIGN	112		*						
AF1	104							6	✓
DUBLAT	116							6	✓
MACHBOX	136							6	✓
RH03	150							6	✓
FLUTTER	122								*

①

- A check (✓) identifies a data set that is required for the corresponding type of analysis.
- An asterisk (*) identifies a data set that may be used to define the problem for the corresponding analysis type. Any one or combination of these data sets may be input.
- Any one or combination of AF1, DUBLAT, MACHBOX, and RH03 data sets may be input in any order.
- All of the data sets shown in the table or any combination thereof may be included in a data deck. Only those data applicable to the problem - execution statements specified in the Control Program are used.

11.2 CDC 6600 JOB CONTROL CARDS

The CDC 6600 control cards required to execute most types of ATLAS jobs are supplied as cataloged procedures on the ATLAS program tape and permanent disk files. Thus, only a small number of control cards must be supplied by the user for normal runs. These control cards, valid for the NOS 1.2 operating system, are described below. Each of the integers within parentheses on the left side of the following list refers to a subsequent explanatory note. Each control-card statement begins in column one of a card.

- (1) Job Card
- Account Card
- (2a) GET (CATM/UN=UATLASU)
- (2a) CALL (CATM,ATTACH)
- (2b) VSN (ATLASMT=66xxxx)
- (2b) LABEL (ATLASMT,NT)
- (2b) REWIND (ATLASMT)
- (2b) COPYBF (ATLASMT,CATM)
- (2b) CALL (CATM,ATODISK)
- (3a) CALL (CATM,control)
- (3b) VSN (B=66xxxx)
- (3b) LABEL (B,F=I,LB=KL)
- (3b) COPYBF (B,CONTROL)
- (4) VSN (C=66xxxx)
- (4) LABEL (C,F=I,LB=KL)
- (4) COPYBF (C,Savefile)
- (4) REWIND (Savefile)
- (5) ATLAS (PL=100000)
- (6) REWIND (Savefile)
- (6) REQUEST (D,F=I,LB=KL,PO=AW) SAVE
- (6) COPYBF (Savefile,D)
- (7) PLOTFIL (Plotter,TAPE99,0)
- (8) CALL (CATM,DROPRNF)

NOTES:

- (1) The requirements for time and field length on the job card vary with the size of the problem and the modules being executed (see sec. 11.3).
- (2a) Control cards required to access the ATLAS program on permanent file.
- (2b) Control cards required to copy the ATLAS program files from magnetic tape to disk. Either the set of cards tagged by (2a) or the cards tagged by (2b) are input. Both sets are not included in a job deck.

- (3a) This card causes a Control Program in source format to be compiled and loaded. The word "control" must be input as one of the following words:

CCONTROL --The Control Program contains no SNARK language statements.

KONTROL --The Control Program contains SNARK statements (ref. 1-3).

The Control Program source code must be supplied in the INPUT deck and all user-defined cataloged control statements, if referenced via PERFORM control statements, must be on the file USERPRO (ref. sec. 200.4). Printout from the compilation and loading is directed to the OUTPUT file.

A more general form of this control card is as follows:

```
CALL(CATM,control(INPUT=infile,OUTPUT=outfile,  
USERPRO=userfile))
```

In this case, different file names may be assigned to INPUT, OUTPUT and USERPRO via "infile," "outfile" and "userfile," respectively. If any of the parameter pairs is not specified, the upper-case file names as shown in this statement are used during compilation and loading the Control Program.

- (3b) These cards should be input when an absolute Control Program is to be copied from tape.
- (4) These cards should be input if data generated by a previous job are to be loaded to restart execution of a problem by this job. Data are loaded via the LOAD Control-Program statements described in section 200.3.1. Refer to note (6).
- (5) This control card initiates execution of the ATLAS program. The data deck, unless restart-data are loaded, must be supplied in the INPUT file. All computer printout is directed to the OUTPUT file.

A more general form of this control card is as follows:

```
ATLAS(PL=nxxx,infile,outfile,ttyfile)
```

The parameters within this statement provide the capability of changing the job printout line limit and the input and output file names. These parameters are as follows:

- PL=nnnn --The printout line limit is set to the integer nnnn.
Default: PL=5000
- infile --ATLAS reads input from the file "infile."
Default: INPUT
- outfile --Program printout is directed to the file "outfile."
Default: OUTPUT
- ttyfile --A duplicate of the DAYFILE messages is directed to the file "ttyfile."
Use of this file is intended primarily for time-shared execution of ATLAS.
In this case, "ttyfile" is set to OUTPUT and "outfile" is assigned a name different from OUTPUT.
Default: TAPE96

Whenever the line limit option is used, it must be the first parameter inside the parentheses. If this option is not exercised, the general form of this control card should be:

ATLAS(infile,outfile)

The indicated sequence of the input and output files should be observed. If "infile" is not specified, the comma preceding "outfile" must still be input.

Examples:

ATLAS (PL=40000,IN,OUT)
ATLAS (PL=100000,,OUT)
ATLAS (DATA,OUTFILE)
ATLAS (,OUT)

- (6) These cards should be input if the ATLAS System restart facilities are to be used. Data generated by this job are to be saved for restart of problem execution by a subsequent job (see cards tagged with 4). The specified restart file-name denoted by "Savefile" must be one of the names defined in section 200.2. Data are saved via the SAVE Control-

Program statements described in section 200.3.1. If data associated with multiple "Savefiles" are to be saved, multiple cards of this type should be input. Positioning of all "Savefiles" and handling of magnetic tapes (permanent files) on which saved data are written are responsibilities of the user.

- (7) This card is required whenever off-line plots are requested via an EXECUTE GRAPHICS control statement (see sec. 228.0). Either the Stromberg Carlson SC4020, the CALCOMP Model 763 or the GERBER drafting-machine off-line plot system may be requested. The plot hardware unit to be used is identified by the word "Plotter" as shown in the following cards.

```
PLOTFIL(SC4020,TAPE99,0)
PLOTFIL(CALCOMP,TAPE99,0)
PLOTFIL(GERBER,TAPE99,0)
```

Only one plot hardware system may be requested per ATLAS job execution. Any CDC 6600 COMMENT control cards that immediately follow the PLOTFIL card are automatically directed to the operator of the specified plot unit.

- (8) This card should be input prior to each subsequent execution of the ATLAS program (card 5) in the same job. This card causes the data-communication files used by ATLAS to be returned.

11.3 ATLAS EXECUTION RESOURCE REQUIREMENTS

The amount of central memory (CM, core or field length), central processor time (CP) and printout-line (PL) computer resources required to execute an ATLAS job depends on the size and type of problem being solved. Estimates of the CM and CP resource requirements are requested by the user via the CDC 6600 "Job Card," whereas the PL requirement is requested by the ATLAS execution card (see sec. 11.2). The default values for CM, CP and PL are not sufficient for most ATLAS jobs and therefore, estimates of these resources must generally be made by the user. Management of disk storage requirements for "large problems" should also be performed by the user to make execution possible.

Insufficient computer resources requested to execute an ATLAS job will cause the job to be aborted except that additional CP time may be requested during a time-shared execution. The execution cost is not affected by overspecification of CP and PL. Improper core management, however, causes unnecessary expenses. In general, the CM cost of an ATLAS job exceeds 50 percent of the total cost. Thus, a user can avoid substantial expenses by proper core management. A detailed discussion on CM requirements is presented in section 11.3.1.

During execution of ATLAS, most of the CP time is consumed by the processors. To assist the user in controlling execution costs, the CP-time requirements of a job are reported in the following two manners:

- a) Before a "Module" is executed, ATLAS(0,0) issues the DAYFILE message "EXECUTING Module CP=NN" where NN is the total elapsed CP-time for the job.
- b) After execution of a module (including returns from a Control Program), ATLAS(0,0) issues to the output file a message that contains the CP-time used by the module and the total elapsed CP-time for the job.

The bulk of the printout from an ATLAS job is generated by execution of the Postprocessors as requested by the user. The total number of printed lines (the PL requirement) for a job is printed by the CDC 6600 operating system.

Assistance in management of disk storage requirements for analysis of "large problems" may be obtained from the ATLAS Staff. "Large Problems" are those that require large amounts of disk storage and many direct-access data records. Such problems arise when a structural model is comprised of many

nodes or when many modules are executed. A stress analysis, for example, of a model with less than 1500 nodes does not normally require special attention to disk-storage management.

11.3.1 Central Memory Requirements

Central memory requirements are dependent on the problem that is being solved and vary from one module to another. The initial amount of available core is defined by the user via the CDC 6600 "Job Card" (see sec. 11.2). Management of the amount of core available during execution of a job may be performed by the user via "CALL REQFL" statements within his Control Program (see sec. 200.0). The amount of core available at execution time of a module must be sufficient for execution of that module.

As a rule, the initial CM request is the maximum core that is required by any of the modules to be executed during a job. Use of the "REQFL" facility within a Control Program allows the available core to be varied for execution of each module. This, in general, results in a lower execution cost.

The requested field length must be sufficient to accommodate three logical components, (1) the executable program code (PC) and certain fixed-length data blocks, (2) the random-access data index tables (RAT), and (3) the open core (OC or blank common) used for computations by the modules. With respect to execution of a module, the PC and RAT components are fixed values (problem independent). The OC component, used by the ATLAS modules for dynamic management of data, is defined by the expression $OC = (CM - PC - RAT)$. Thus, the OC is controlled by the user via the specified field length. The OC requirement for a module is one of the following types:

- a) Problem independent (PI)
- b) Problem-dependent maximum requirement (PDMAX)-- additional OC available at execution time is of no benefit.
- c) Problem-dependent minimum requirement (PDMIN)-- additional OC available at execution time may be beneficial. There is an optimum OC with respect to execution cost.

To determine the required field length for a job, the user must know the following:

- a) the program length (PC) of each of the modules to be executed
- b) the RAT space
- c) the OC requirement of each of the modules to be executed

The field length required by the PC of each module is presented in table 11-3. It should be noted that each module, in general, has several secondary overlays with different lengths and depending on the problem, only some or all of these overlays may be executed. The PC lengths presented in table 11-3 accommodate the longest secondary overlay included in a module.

The RAT space is 3000 (decimal) words (5670 octal) in length unless changed by the user via a statement "CALL OPENR(0,n)" in his Control Program. The letter "n" denotes the desired RAT space subsequent to this statement. This facility may be used to reduce execution costs for small problems.

The OC requirement of a module is, in general, difficult to define by simple arithmetic expressions because of the problem dependency. That is, the OC requirement depends on model parameters such as the number of nodes and elements in addition to what type of execution is requested. Many expressions per module would have to be evaluated to determine the minimum OC that may be requested depending on the mathematical models and the execution modes. To alleviate this problem, simplified approximate expressions for OC requirements are presented in table 11-3 for most of the modules. The data in this table are based on the default RAT space and default internal data-record lengths. A module marked with an asterisk in table 11-3 denotes that core usage is monitored and reported on the output file for user reference. This information allows the user to develop the ability to estimate the CM requirements for future executions of ATLAS such that reduced costs may be realized.

Table 11-3. Computer Central Memory Requirements of the ATLAS Modules

MODULE NAME	PREPROCESSOR		PROCESSOR		POSTPROCESSOR	
	CODE LENGTH AND TYPE	TOTAL CORE REQUIREMENTS	CODE LENGTH AND TYPE	TOTAL CORE REQUIREMENTS	CODE LENGTH AND TYPE	TOTAL CORE REQUIREMENTS
ADJOINT			19200 PDMAF	43000 + 2M (without ACS), 23500 + 4M (with ACS)	19500 PDMAF	21000 + 2M
AFI	25100 PDMAF	37000	25100 PDMAF	100 + 14000 + 2M	19500 PDMAF	30700 + max { 2M, 2M } 23000 + 95 STAGE + 13M LOCAL + F
BOUNDARY CONDITION	32500 PDMAF	{ 34000 + 3M (no specified displacements), 45000 + 3M (specified displacements) }			20000 PDMAF	
BUCKLING			27600 PDMAF	max { 32800 + 20F, 28300 + 6F + F(F+1)/2 }	19500 PDMAF	
CHOLESKY			21000 PDMAF	33300		
DESIGN	28500 PDMAF	38500 + E	34500 PDMAF	45000 + 3E + 30000S	28500 PDMAF	36000 + 2E
DETAIL	25000 PDMAF	50700 + 2E				
DUBLAT	24600 PDMAF	34800	32500 PDMAF	35800 + 2080X + 2M	19100 PDMAF	22500 + 2/8CE + 50F + M
EXTRACT						
FLEXAIR			7600 PDMAF	79700 + max { 1/2 E, 1/2 E + 5/2 F, 1/2 E + 7F, 1/2 E + 1/2 M + 3/2 F }	16400 PDMAF	74800 + 2M
FLUTTER	37800 PDMAF	37800	28200 PDMAF	79700 + { 9M + 72M (no vectors), 9M + 9M + 72M (with vectors) }	7500 PDMAF	21000 + 4M + 63M
FREEBODY					7400 PDMAF	30000 + 2M + 1
GEOMETRY	52000 PDMAF	56000				
GRAPHICS					7400 PDMAF	45000 + 14M + M/60 (for 150 dependent data)
INTERACT	6500 PDMAF	30000 + 10M + 5M (max) + 2M			30000 PDMAF	36500 for other data types
INTERPOLATION					33000 PDMAF	33000
LOAD	74500 PDMAF	30000 + M			33000 PDMAF	30000 + 4M + 2M
LOADS	30700 PDMAF	43000 + 10M + 5E + 6M	37700 PDMAF	15000 + 8M + 2E + 6M (no theta cards)	24000 PDMAF	35000 + 50C + 10M + 13M (local) + 95 STAGE (PRINT) + 2M
MACHBOX	47000 PDMAF	34800	64000 PDMAF	48000 + 8M + 2M (theta loads)	73000 PDMAF	37500 + 10M (PRINT) + 2M

Table continued on next page

Table 11-3. Computer Central Memory Requirements of the ATLAS Modules (Cont'd.) (1)

MODULE NAME	PREPROCESSOR		PROCESSOR		POSTPROCESSOR	
	CODE LENGTH AND TYPE	TOTAL CORE REQUIREMENTS	CODE LENGTH AND TYPE	TOTAL CORE REQUIREMENTS	CODE LENGTH AND TYPE	TOTAL CORE REQUIREMENTS
MASS	48000 PDMAX*	41000	25000 PDMAX*	OPTIONAL: OPTIOM=2,3 max { Δ }, OPTIOM=4 max { Δ }, 16864 Δ 33'004E, 48N 10'559M1, 4+4E* 2'5'004M1, 4+4E* 366004E, 48N 48N + 4N	25600 PDMAX*	41000
MATERIAL	48000 PDMAX*		25000 PDMAX*		17500 PT	23000
MERGE						
MULTIPLY						
MODAL						
PRINT						
REACTION						
RHO3	48000 PDMAX*	18400	28000 PDMAX*	max { Δ }, 13000 + 2044, 48N	19000 PDMAX	22000 + 48
SAVE						
STIFFNESS	45000 PDMAX*	max { Δ }, 13500, 28000+4E	29000 PDMAX*	max { Δ }, 8'500M, 13000 + 113B(18+3)/2, 24600M/8+5B(11948)}	23000 PDMAX	37000 + 2E
STRESS	45000 PDMAX*	18000 + 3E	19000 PDMAX*		22500 PDMAX	35000 + N + E
SUBSET DEFINITION	18000 PDMAX*	18000 + 7N + 4E	27500 PDMAX*			
VIBRATION				max { Δ }, 12800 + 20F, 28300 + 6F + F(F+1)/2}	19500 PDMAX	22500 + (F+1)M

(1) All numbers and terms in the expressions shown in the table are decimal numbers. Conversion to octal numbers is facilitated by Table 11-4. An asterisk (*) denotes that the module core usage is reported on the output file. Core requirements are PI (problem independent), PDMAX (problem dependent maximum) or PDMIN (problem dependent minimum). The following symbols denote integer numbers of various quantities associated with the mathematical models.

- B = Maximum number of nodes defining a BRICK
- BCE = Body doublet elements
- BOA = Boxes
- CE = Design data subsets
- DAF = Downwash points
- E = Efficiency elements
- EI = Efficiency elements or mass elements, whichever is larger
- F = Retained degrees-of-freedom
- FALC = Freedoms in the AIC matrices
- K = Reduced frequencies (h-values)
- LC = Load cases
- LOCAL = Local node reference frames
- M = Vibration modes
- MB = Maximum data-block (record) size (generally 5000)
- N = Nodes
- N₁ = Nodes in a substructure
- PSD = Spanwise panel divisions
- PT = Retained degrees-of-freedom or mass panels, whichever is larger
- STAGE = Boundary condition stages

Table 11-4. Numeric Conversion Table

Octal	Decimal	Octal	Decimal	Octal	Decimal
60000	24576	140000	49152	220000	73728
61000	25088	141000	49664	221000	74240
62000	25600	142000	50176	222000	74752
63000	26112	143000	50688	223000	75264
64000	26624	144000	51200	224000	75776
65000	27136	145000	51712	225000	76288
66000	27648	146000	52224	226000	76800
67000	28160	147000	52736	227000	77312
70000	28672	150000	53248	230000	77824
71000	29184	151000	53760	231000	78336
72000	29696	152000	54272	232000	78848
73000	30208	153000	54784	233000	79360
74000	30720	154000	55296	234000	79872
75000	31232	155000	55808	235000	80384
76000	31744	156000	56320	236000	80896
77000	32256	157000	56832	237000	81408
100000	32768	160000	57344	240000	81920
101000	33280	161000	57856	241000	82432
102000	33792	162000	58368	242000	82944
103000	34304	163000	58880	243000	83456
104000	34816	164000	59392	244000	83968
105000	35328	165000	59904	245000	84480
106000	35840	166000	60416	246000	84992
107000	36352	167000	60928	247000	85504
110000	36864	170000	61440	250000	86016
111000	37376	171000	61952	251000	86528
112000	37888	172000	62464	252000	87040
113000	38400	173000	62976	253000	87552
114000	38912	174000	63488	254000	88064
115000	39424	175000	64000	255000	88576
116000	39936	176000	64512	256000	89088
117000	40448	177000	65024	257000	89600
120000	40960	200000	65536	260000	90112
121000	41472	201000	66048	261000	90624
122000	41984	202000	66560	262000	91136
123000	42496	203000	67072	263000	91648
124000	43008	204000	67584	264000	92160
125000	43520	205000	68096	265000	92672
126000	44032	206000	68608	266000	93184
127000	44544	207000	69120	267000	93696
130000	45056	210000	69632	270000	94208
131000	45568	211000	70144	271000	94720
132000	46080	212000	70656	272000	95232
133000	46592	213000	71168	273000	95744
134000	47104	214000	71680	274000	96256
135000	47616	215000	72192	275000	96768
136000	48128	216000	72704	276000	97280
137000	48640	217000	73216	277000	97792

100.0 INPUT DATA GENERAL INFORMATION

General conventions and terminologies that are applicable to all input data as described in the 100-series sections of this document are defined in this section. This general information includes the following:

- a) Structure of an input data deck;
- b) Input data formats including a description of the notation used in this document to describe the input data;
- c) Automatic input-data generation capabilities;
- d) Definitions of nodes, elements, coordinate systems, analysis reference frames used to describe nodal kinematics and units of measurement.

Example data decks for selected problems are illustrated in appendix F. These decks illustrate the various formats in which the input data records may be written.

~~REPLACING THIS LABEL NOT FILMED~~

100.1 DATA DECK

A data deck as illustrated in figure 11-1 is comprised of all the user-supplied input data which define the problem to be analyzed via one execution of ATLAS. The basic input data component is a data record and its data items. Their relationships in a data deck are defined as follows:

- a) Data Deck --A data deck is composed of one or more data sets.
- b) Data Set --Input data associated with one of the preprocessors indicated by table 10-1. A data set is comprised of two or more data subsets.
- c) Data Subset --A group of data within a data set. A data subset is comprised of many data records.
- d) Data Record --A string of numbers or alphanumeric words. A record commonly implies an input card although it may require two or more cards. Each data record is composed of one or more data items.
- e) Data Item --Either a number or a character string.

The first record of a data deck may be

```
BEGIN PROBLEM DATA
```

or it may be the first "BEGIN Set-Name DATA" record of a data set. The last record of a data deck, however, must be

```
END PROBLEM DATA
```

In special cases, multiple data decks may be required when executing a single job. In this situation, each data deck ending with the record "END PROBLEM DATA" is stacked one behind the other. Correspondingly, multiple READ INPUT statements (sec. 200.3) must be included in the Control Program such that each data deck is read and preprocessed.

Each data set begins with an input record of the form

```
BEGIN Set-Name DATA
```

and ends with a record of the form

END Set-Name DATA

Each data set is identified by a unique Set-Name. The data defined by a data set are, in general, associated with a particular technology. The remaining 100-series sections of this document describe the input data sets associated with each of the data preprocessors. Restrictions regarding the input order of multiple data sets are described in the appropriate sections.

Generally, a data subset is identified by an input record of the form

BEGIN Subset-Name DATA

and is terminated by a record of the form

END Subset-Name DATA

Each subset within a data set is identified by a unique Subset-Name. The data associated with each subset of a data set are described in the appropriate sections herein. Some data subsets are identified as being optional. Input of these data subsets is not required if the default data values are acceptable for definition of the problem.

100.2 INPUT DATA FORMAT

General rules concerning input data formats for data records, data items and data comments are described in this section.

Throughout this document, input data records are enclosed within rectangular boxes to distinguish them from descriptions and explanations of the input items.

100.2.1 Data Records

Although input data can be defined using either card input or direct input via a computer terminal, reference to card input data is made below. Input format rules regarding all data records are as follows.

- a) All data items which define a record may be input in a free-field format. Data item delimiters are blanks, commas, colons, semicolons or card boundaries.
- b) Some or all of the items in a record may initiate data generation (part (b) of sec. 100.3). Two types of data generation are available: 1) item generation within a record and 2) generation of one or more complete records from a parent record. Item generation within a record is performed before interrogating all the data specified by a record. This causes the length of the original data record to be expanded. A record that is not a parent record for generation of other records may contain a maximum of 250 items in its expanded form. However, a parent record used for record generation may contain a maximum of 125 items in its expanded form. See section 100.3 for descriptions of the automatic, data-item generation capabilities.
- c) A sequence (list) of data items may continue onto as many cards as required to complete a record (ref. (e) below). All 80 columns of a card may be used.
- d) Short records in sequence may be input on the same card. A dollar-sign \$ must be input to separate records punched on a single card.
- e) By default, a single record input by a single card or the last record on a card must be terminated by either / or \$. This form of record termination is referenced as the "MODE1" input format. The user may select the "MODE2" input format which has the following characteristics:

- 1) Each right-hand card boundary is a record terminator unless otherwise specified (see 2) below). Multiple records may be input on one card as described by (d).
- 2) A plus sign (+) as the last character on a card denotes that the record continues onto the next card.
- 3) If the last character on a card is / it is treated as a record terminator.

The mode format is initially assumed to be MODE1. This initial input format may be changed to MODE2 via the READ INPUT Control Program statement (sec. 200.3.1). The format may also be specified by input records. A record of the form */ MODE2 / denotes that subsequent records are to be interpreted in the MODE2 format. Conversely, the input record */ MODE1/ denotes that subsequent records are to be interpreted in the MODE1 format. The input mode established by an input record of this type takes precedence over the format specified by the READ INPUT statement. Additionally, the last input format specified via a record within the input stream is the one that is effective for subsequent data records. Both data-record, input-format modes may be used in setting-up a data deck.

All data records enclosed within boxes as described in this document are shown in the MODE2 input format.

100.2.2 Data Items

Each data item is categorized as one of the following types:

- a) Decimal Number--The first character may be +, -, a decimal point, or a numeric. A decimal point and/or an E, when the FORTRAN E-Format is employed, must be contained in the number. A decimal number may have a maximum of 20 characters.
- b) Integer Number--All characters must be numeric except the first one which may be + or -. An integer must not contain a decimal point and may have a maximum of 18 numeric characters.
- c) Alphanumeric Word--At least one character in the item must be nonnumeric. It is an item that cannot be interpreted as a "decimal" or an "integer" number. The first character may not be *, / or \$ and the item may not have any embedded blanks, commas, colons or

semicolons. Input "alphanumeric words" exceeding 10 characters are automatically truncated to the left-most 10 characters. Words that are typed in all upper case letters within the boxed-in data records are called ATLAS System key-words.

- d) Alphanumeric Text--Text strings may contain any symbol in the CDC 6600 character set except the apostrophe. A text string is initiated by the two characters *# and must be completed by the character #. The character # and an apostrophe are interpreted identically by the computer. Input of this character is effected by a multiple 4-8 punch. The length of a text string is limited to 2500 characters including blanks.

The following notation is used throughout this document in describing all input data items:

- a) { } --Data items enclosed by braces have optional input formats. One of the indicated options must be selected.
- c) < > --Data items enclosed by brackets have default values. If the default is acceptable for definition of the problem data, the particular item or items need not be input. All default values are defined in the descriptions of the input data.
- c) ITEM --An item typed in all upper case letters is called a key-word. At least the underlined portion of a key-word must be input.
- d) Item --An item with only the leading character typed in upper case denotes that it must either be selected from a list of system key-words or that it is identical to an item previously-defined by the user.
- e) item --An item typed in all lower case letters is defined strictly by the user.
- f) ATlist --This word denotes that one item or a list of items may be input. A list may denote item generation. The options are:

$$\text{ATlist} = \left\{ \begin{array}{l} \text{item} \\ \text{list} \\ \text{a TO b <BY c>} \end{array} \right\}$$

The word "list" denotes a list of two or more items. The input list "a TO b <BY c>" causes the sequence of items a, a+c, a+2c, ..., b to be generated. Items "a" and "b" are either positive or negative integers or they are alphanumeric words with trailing digits (e.g., N8). The leading characters of "a" and "b," when they are alphanumeric words, must be identical. The item "c" is a positive or a negative integer and the words TO and BY are input as shown. The right-most integer component of "b", such as 8 in N8, minus the integer component of "a", such as 2 in N2, when divided by "c" (e.g., 2) must be a positive integer (the number 3 in this case). If the list-generation increment "c" is 1 or -1, the items "BY c" need not be input. One or more of the input forms enclosed within the braces may be used to define the corresponding items.

Note: Throughout this document, an item that is not indicated as a key-word must be input as a "decimal number" unless noted otherwise in the detailed data-item descriptions. Brackets < > and braces { } used in describing the record formats are not input.

Input of ATlist is illustrated by the following examples:

2 TO 10 BY 2	denotes 2,4,6,8,10
3,5,N2 TO N4,12	denotes 3,5,N2,N3,N4,12
3 TO 1 BY -2,13	denotes 3,1,13
A26,T1A05 TO T1A11 BY 2	denotes A26,T1A05,T1A07,T1A09,T1A11

In some cases, use of an ATlist list generator may be useful even though one or more of the generated items is not valid. To illustrate this, consider the last example shown above which might be used to reference previously defined items. Although T1A07, for example, was not previously defined, use of the string generator still offers a user-convenience in many situations. The non-existent item T1A07 is simply ignored by the preprocessor which is interrogating the particular data. If a list generated by ATlist is not allowed to have non-existent items, it is noted accordingly in the item description.

100.2.3 Data Deck User Comments

Comments are text which may be embedded in the data deck for conveniences of user-identification. Comments do not affect the data preprocessing activities. User comments may be included within a data deck in the following ways:

- a) A record beginning with the two characters */ is a user-comment record. All characters in a record of this type are treated as data-deck comments. The first word of user comments in a record of this type must not be MODE1 or MODE2.
- b) Comments may follow the / record terminator and continue to the right-hand boundary of a card. This option may be used when either the MODE1 or MODE2 data-record input format is used.
- c) Comments enclosed between the characters *(and) may be inserted any place within a data record. Pairs of parentheses, data-item delimiters and record termination characters may be included within this type of comment.

User comments of type (a) and (c) may continue onto two or more cards, as required.

Examples: 10 2.0 3.0 100.0 / DEFINITION OF NODE 10
 */ NODE 11 IS DEFINED BY THE NEXT RECORD/
 11 *(NODE 11) 4.0 3.0 100.0/

100.3 AUTOMATIC INPUT DATA GENERATION

There are two types of input-data generation capabilities inherent to ATLAS. Use of these capabilities minimizes the amount of required user-supplied data as described below.

- a) Automatic generation of problem-definition data associated with a particular data preprocessor. These capabilities are discussed under the appropriate 100-series section. User-oriented capabilities for automatic generation of the nodes, elements and geometry required for finite element modeling and the availability of carefully-selected system default values for the many required quantities characteristic of a finite element analysis are typical examples. Another example of data item generation is the ATlist "a TO b BY c" option provided by several of the preprocessors.
- b) Automatic generation of input data on a data record basis. These special input features allow the user to repeat items within a record, generate items within a record or generate whole records. Each of these capabilities is controlled by the use of an asterisk-type data item and is applicable to any input data record in a data deck. All types of data items except alphanumeric text (sec. 100.2.2) may be generated. These optional capabilities are:
 - 1) *N--A data item of the form *N, where N is an unsigned integer, indicates that the next N data items in that record are identical to the corresponding N items in the preceding record. An isolated * is treated as *1.
 - 2) **--Double asterisks indicate that all of the remaining items in a particular record are identical to the corresponding items in the preceding record. The same number of items is defined by both the ** record and the parent record.
 - 3) *=N--A data item of the form *=N, where N is an unsigned integer, indicates that the next N items in a particular record are identical to the immediately preceding item in that record. The characters *= followed by a blank are treated as *=1.
 - 4) *=N±STEP--A data item of the form *=N±STEP, where N is an unsigned integer and STEP is an unsigned

numeric item (integer or decimal), indicates that the next N items of a record are to be generated by consecutively incrementing the immediately preceding item of that record by \pm STEP (see note below).

- 5) $^{*}N$ --A data record beginning with the characters $^{*}N$, where N is an unsigned integer, indicates the next N records are to be generated from the preceding record. Each item of a generated record is formed by adding an item of the $^{*}N$ record to the corresponding item of the immediately preceding, input or generated, record (see note below). A zero (integer) item should be inserted in an $^{*}N$ record to indicate that an alphanumeric word in this record is identical to the corresponding alphanumeric word in the preceding record. Any combination of the preceding asterisk items (see 1) through 4) above) may be used in an $^{*}N$ record.

Note: An item of an $^{*}N$ record to be added to a corresponding item of the preceding record (see 5) above) and the STEP item (see 4) above) must meet certain requirements. Each of these items must be a decimal number if the corresponding item is a decimal number, whereas each of them must be an integer if the corresponding item is an integer number or an alphanumeric word. It should be understood that the $^{*}N$ record generation option is based on the expanded representation of the previous record. The generation does not operate on the card image of the preceding record if it contains data-generation items. Therefore, it is not possible to repeat or increment an asterisk-type item.

Consider the following seven input records to illustrate the asterisk, automatic-generation input features.

```
1 2 3 4 5 6 7 8 9 10 11 12 /
2 1 *2 4 *=2 1 *=2+2 ** /
**+1 0 *=3 0 *=5 ** /
**+1 0 *=11 /
**+1 *12 /
**+1 ** /
** /
```

Twelve data items are defined by each of these records. Each of the last six records is translated into the same internal record which is

2 1 3 4 4 4 4 1 3 5 11 12 /

If this sequence of data were required within a data deck, the last five records could be replaced by the single record

**5 ** /

to achieve the same results.

Generation of alphanumeric items is illustrated by the following two records.

T1A01 TO T1A15 *=2+2 10.0 /
*+2 1 0 2 3 4 1.0 /

These records are equivalent to the following records:

T1A01 TO T1A15 T1A17 T1A19 10.0 /
T1A02 TO T1A17 T1A20 T1A23 11.0 /
T1A03 TO T1A19 T1A23 T1A27 12.0 /

100.4 NODES AND ELEMENTS

A physical structure is idealized as a network of nodes and finite elements. Nodes are points in three-dimensional space which define the geometry of a mathematical model. The coordinates of all nodes are specified by the user via the nodal data (sec. 146.0). The physical structure is idealized by finite elements that are interconnected at the nodes.

Stiffness (structural) elements required for static and dynamic analyses are defined by the stiffness data (sec. 152.0). These data include the geometric properties, material properties and mass density of the stiffness elements.

Mass (non-structural) elements may be defined by the mass data (sec. 138.0) to supplement the mass associated with the stiffness elements. Non-structural, distributed mass properties such as fuel and payload, can be defined conveniently by this data set. Alternatively, if only a mass analysis is to be performed, the mathematical model may be defined strictly by nodal and mass data.

100.5 COORDINATE SYSTEMS AND NODAL KINEMATICS

It is necessary to have one overall (common) coordinate system for referencing geometrical data to ensure data compatibility between the ATLAS modules for an integrated analysis. This coordinate system is identified as the GLOBAL reference frame which is a right-handed, Cartesian triad denoted by X-Y-Z.

Geometry data are input relative to the GLOBAL coordinate system and, in some cases, relative to local coordinate systems defined by the user. A local frame may be a rectangular x-y-z, cylindrical r- θ -z or a spherical ρ - θ - ϕ coordinate system. All coordinate systems are orthogonal and right-handed (see fig. 100-1). Compatibility of data between GLOBAL and local coordinates employed within a data set is automatically ensured. It is, however, the user's responsibility to ensure compatibility of GLOBAL geometrical data between data sets when an integrated analysis is performed.

The kinematics of a node are described relative to its analysis frame. The analysis frame of one node may be different from the analysis frame of another node. Additionally, the analysis frame of a node may be different from its input reference frame relative to which its coordinates are defined. All nodal input and analysis reference frames are defined via the nodal data (sec. 146.1.1).

Figure 100-1 illustrates the default notation used in ATLAS for kinematic freedoms and the associated nodal forces relative to rectangular, cylindrical and spherical coordinate systems. These kinematic and force labels may be changed to user-selected labels via the boundary condition data (sec. 106.0).

Kinematic constraints on nodes are specified by the boundary condition data (sec. 106.0). Static, concentrated loads applied at the nodes or pressure loads on the stiffness elements are specified by the loads data (sec. 134.0).

Other local coordinate systems applicable to data associated with certain preprocessors are defined where they are first encountered. For example, properties of the stiffness finite elements and some of the mass finite elements are defined relative to right-handed, Cartesian reference frames x-y-z which are associated with the elements (see appendices B and C).

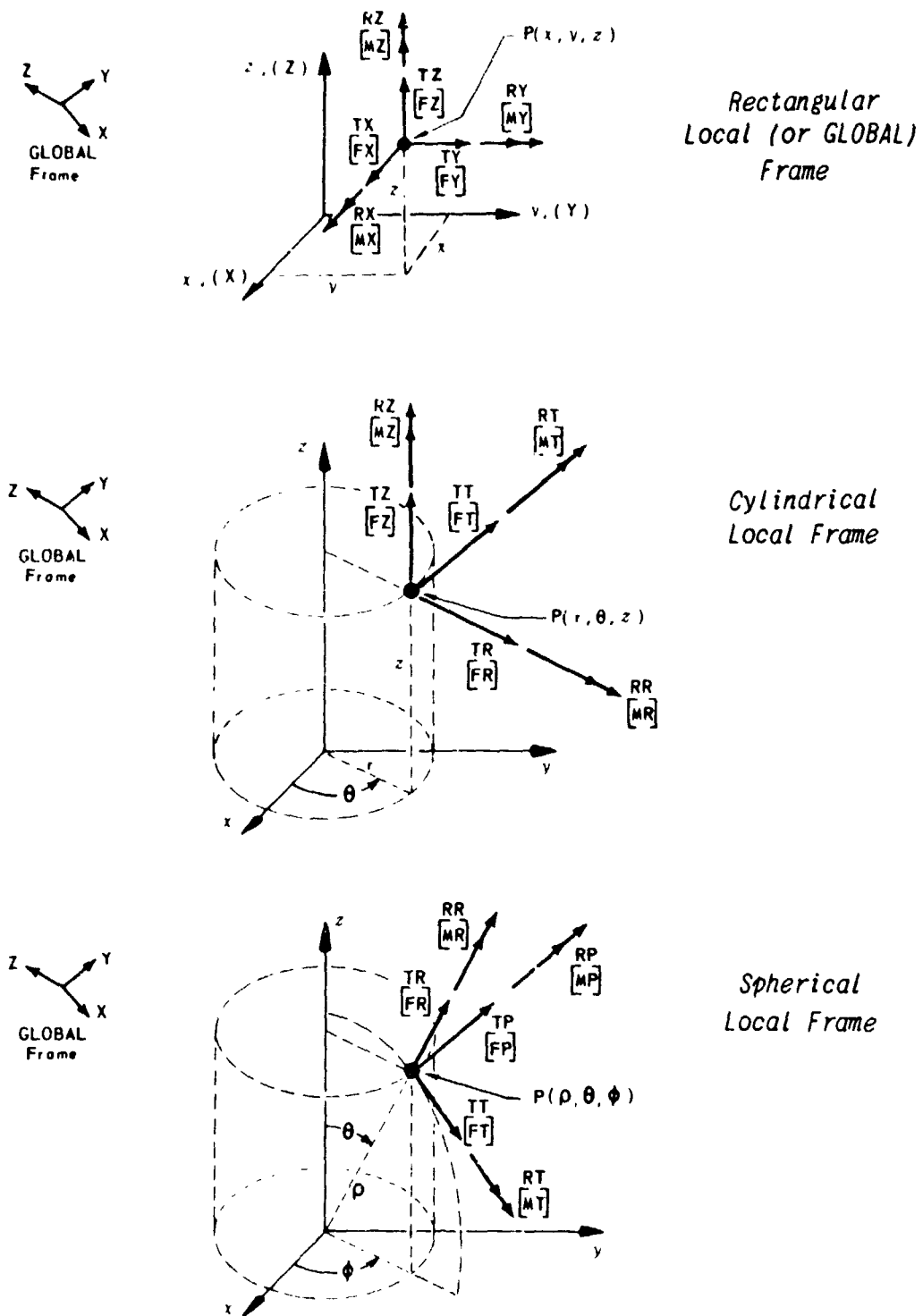


Figure 100-1. Kinematic and [Force] Labels and Conventions

100.6 UNITS OF MEASUREMENT

The standard measurement units used in ATLAS are length in inches, force in pounds, and angles in radians. Compatible units of other input items are noted in the descriptions of the data records. The material property tables built into the system are based on these units with temperatures measured in degrees Fahrenheit. These data are used by the STIFFNESS, MASS and DESIGN Processors when Standard materials are selected. Thus, data supplied by the user must be consistent with these units. That is, coordinates must be in inches and forces or weights must be in pounds if Standard material property tables are used (see sec. 140.0). The user may, however, choose any system of units that is desired if caution is exercised to ensure consistency of units throughout the analyses performed by the selected program modules. Special material property data may be input in the desired units provided that data for every other module are defined with consistent units.

104.0 AF1 AERODYNAMICS DATA

Unsteady aerodynamic analyses of rigid or elastic, nonplanar lifting surfaces defining a general three-dimensional configuration in subsonic incompressible flow can be performed by the AF1 Processor (sec. 204.0). Calculations performed by this processor are based on the Theodorsen two-dimensional aerodynamic strip theory (ref. 104-1) modified to include the effects of static induction (ref. 104-2). Empirical data may be used to replace or modify portions of the theoretically derived information.

The input data set supporting AF1 aerodynamic analyses is comprised of the following data subsets.

- a) Geometry --Aerodynamic lifting surfaces with controls/tabs and strip data that define the aerodynamic model.
- b) Modal --Vibration modes or "AIC" mode substitutes (see sec. 232.0) associated with the aerodynamic model.
- c) Option --Modifications for the theoretically derived lift distributions and aerodynamic coefficients.

The only restriction on the input sequence of these data subsets is that the Geometry subset must appear first. This data set need not be preceded by any other data set within a data deck.

Each set of aerodynamics data is referred to as a CASE. A maximum of 36 AF1 data cases may be identified per job. Printout of the input case data may be requested from the AF1 Postprocessor (sec. 204.1).

PRECEDING PAGE BLANK NOT FILMED

104.1 INPUT DATA

Record 1 Data Set Identification

BEGIN AF1 DATA

This record initiates execution of the AF1 Preprocessor.

Record 2 Data Case Number (optional)

Each set of AF1 aerodynamics data is referenced as a case that is identified by an integer number. If multiple data sets are used in a job, each case must be assigned a unique number.

CASE n

n = Data case number (integer) in the range 1 to 36, inclusive.

Default Record: CASE 1

104.1.1 Geometry Data Subset

This data subset includes the following geometrical definitions.

- a) Planform definition of main lifting surfaces and axes used for calculating stability derivatives.
- b) Planform definition of control and tab surfaces and hinge lines thereof.
- c) Aerodynamic strip boundaries.

All geometry data are defined relative to the GLOBAL reference frame whose positive X-axis is in the aft (streamwise) direction. If the configuration is symmetric about the X-Z plane and the flow is symmetric or antisymmetric, geometric information for only the positive-Y half of the model need be input. Only half of the total aerodynamic forces are calculated by the AF1 Processor for a surface that lies in the X-Z plane of symmetry.

An aerodynamic model may be comprised of any number of spatially-oriented lifting surfaces to idealize a configuration. The model may be defined via multiple main lifting surfaces with arbitrary planforms and dihedral with or without full or partial-span control surfaces and tabs. Intersecting and/or

interfering non-coplanar configurations such as T-tail, V-tail and wing-tail combinations may be analyzed.

Each surface is defined via straight line segments and must be planar. Chordwise edges of all aerodynamic surfaces must be parallel to the free stream (parallel to the X-axis). Trailing-edge control surfaces and tabs associated with a main surface are subject to the following conditions (see fig. 104-1).

- a) Control surfaces must be contained within the main surface planform.
- b) A tab surface must be contained within a control surface planform.
- c) Multiple control and tab surfaces may share common streamwise edges but they may not overlap.

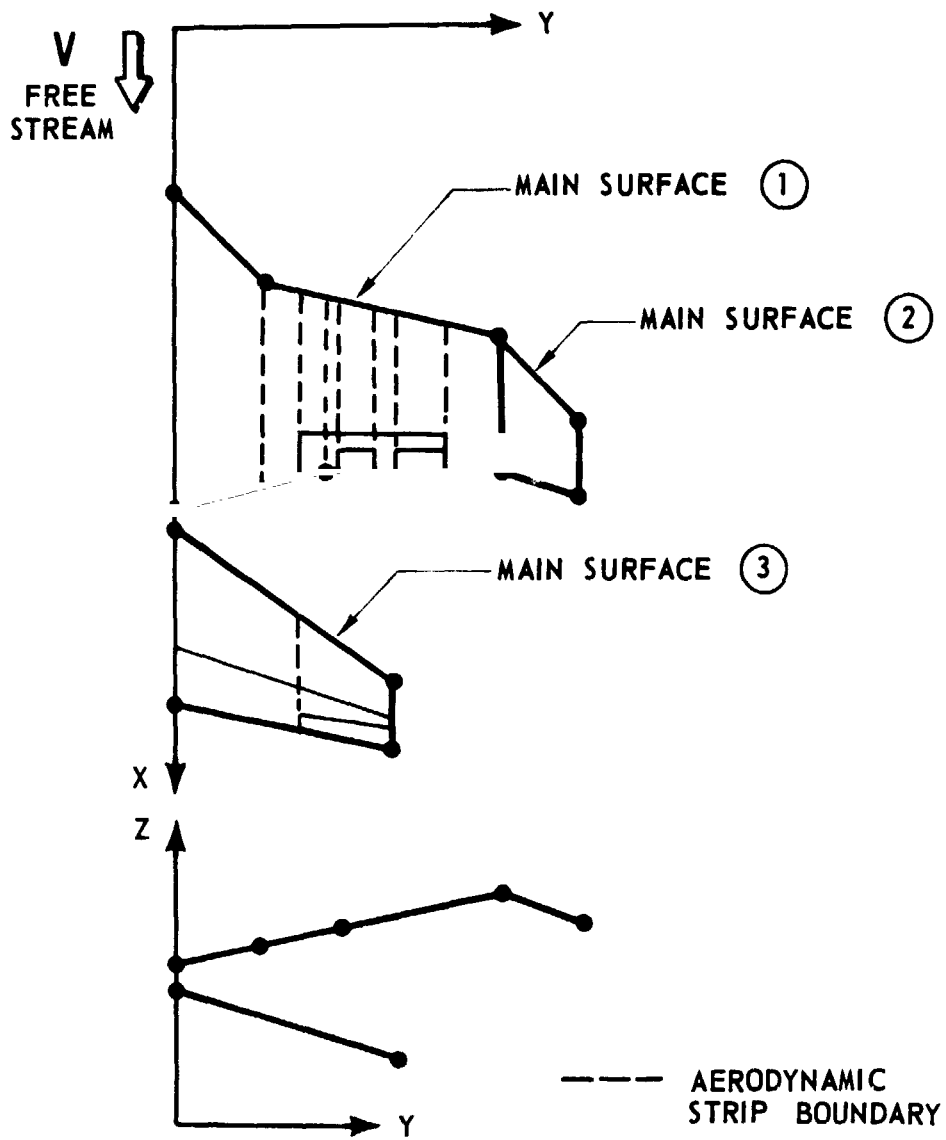


Figure 104-1. Sample AFI Aerodynamic Model

Record 3 Begin Geometry Subset

BEGIN GEOMETRY DATA

An aerodynamic surface is defined by Records 4-11. The main lifting surface is defined by Records 4-8 and any associated control/tab surfaces are defined by subsequent Records 9-11. Records 4-11 are repeated to define multiple aerodynamic surfaces associated with the case specified by Record 2. Only Records 4-6 are required within this data subset.

Record 4 Main Surface Identification

MAIN SURFACE ident

ident = Alphanumeric word of 1 to 5 characters that identifies a main surface.

Records 5 and 6 define the leading and trailing edges, respectively, of the main surface. Each edge is defined by a list of coordinates of 2 or more points input in an order such that the Y and/or the Z coordinates are monotonically increasing or decreasing. The input order establishes a spanwise direction for the surface. The first points used to define the leading and trailing edges must coincide with one streamline, whereas the last points used to define the edges must coincide with a different streamline.

Record 5 Leading Edge Coordinates

LEADING EDGE x1 y1 z1 x2 y2 z2 ...

x_i, y_i, z_i = GLOBAL X-Y-Z planform coordinates of the i-th point defining the main surface leading edge.

Record 6 Trailing Edge Coordinates

TRAILING EDGE x1 y1 z1 x2 y2 z2 ...

x_i, y_i, z_i = GLOBAL X-Y-Z planform coordinates of the i-th point defining the main surface trailing edge.

Record 7 Elastic Axis Coordinates (optional)

Stability derivatives and sectional airforces are calculated about the elastic axis defined by this record. The elastic axis need not be related to the physical elastic axis, nor to the motion axis that may have been

used in modal interpolation. The location of the elastic axis does not affect the Generalized Airforce Matrices nor the Generalized Component Matrices.

ELASTIC AXIS x1 y1 z1 x2 y2 z2 ...

x_i, y_i, z_i = GLOBAL X-Y-Z planform coordinates of the i-th point defining the main surface elastic axis. The axis is comprised of straight line segments through the input points. The list of coordinates of 2 or more points must be input in an order that is consistent with the selected spanwise direction. Additionally, the axis must be in the plane of the main surface and it must be defined over the total span of the surface.

Default Record: The elastic axis is coincident with the leading edge of the main surface.

All the aerodynamic surfaces are divided into streamwise strips for analysis purposes. Strip boundaries are created automatically through each input point used to define the main surface leading edge, trailing edge and elastic axis (Records 5-7) and through each point used to define the hinge lines and leading edges of the associated control surfaces and tab surfaces (Records 9-11). Additional strip boundaries may be defined by Record 8. Strips should be concentrated, for example, in regions where rapidly varying spanwise loads occur such as near streamwise wing tips and control/tab surface edges.

Record 8 Additional Strip Boundaries (optional)

STRIP	{ FRACTIONS DISTANCES }	d1 d2 ...
-------	----------------------------	-----------

J_i = Either the FRACTION of main-surface span ($0.0 \leq d_i \leq 1.0$) or the actual DISTANCE along the main-surface span at which the i-th additional strip boundary is to be located. The "d_i" values are measured in the plane of the main surface and are input in a spanwise order consistent with that used for the data defined by Records 5-7.

Default Record: No additional strip boundaries are generated.

Records 9-11 are input only to define trailing-edge control and tab surfaces associated with the main surface defined by Record 4. A control or tab surface is completely defined by its hinge line and leading edge which must lie in the plane of the main surface.

Record 9 Control or Tab Surface Identification (optional)

{ CONTROL }	SURFACE ident
{ TAB }	

ident = Alphanumeric word of 1 to 5 characters that identifies the CONTROL or TAB surface defined by Record 10.

Default Record: No control or tab surfaces are associated with the main surface.

Record 10 Control/Tab Surface Hinge Line (optional)

HINGE LINE xi yi zi xo yo zo

$\left. \begin{array}{l} xi, yi, zi \\ xo, yo, zo \end{array} \right\}$ = Two triplets of GLOBAL X-Y-Z planform coordinates defining the inboard and outboard limits of the control/tab hinge line, respectively.

This record must be used if Record 9 is used.

Record 11 Control/Tab Surface Leading Edge (optional)

LEADING EDGE xi yi zi xo yo zo

$\left. \begin{array}{l} xi, yi, zi \\ xo, yo, zo \end{array} \right\}$ = Two triplets of GLOBAL X-Y-Z planform coordinates defining the inboard and outboard limits of the control/tab leading edge, respectively.

Default Record: Leading edge coincides with the hinge line defined by Record 10.

Records 9-11 may be repeated to define additional control or tab surfaces for the main surface identified by Record 4.

Record 12 End Geometry Subset

END GEOMETRY DATA

This record indicates that all geometry data have been defined for the case identified by Record 2.

104.1.2 Modal Data Subset (optional)

This subset defines the vibration modes, "AIC" mode substitutes or rigid-body modes to be associated with each aerodynamic surface of the model. If no modal data are input, only the static-induction matrix would be generated by execution of the AF1 Processor.

The unsteady aerodynamic downwash distribution is approximated by a linear combination of modal displacements and rotations about the quarter-chord of each strip. The aerodynamic theory requires the translational components to be normal to the strip surface and requires the rotational components to be normal to the streamlines. Thus, the analysis frame associated with the retained nodal freedoms must have a corresponding orientation. Modes generated via the POLYNOMIAL option of the INTERPOLATION Processor (sec. 232.1) are evaluated in GLOBAL X and Y or X and Z coordinates.

Record 13 Begin Modal Subset

BEGIN MODAL DATA

Record 14 defines the vibration modes or "AIC" mode substitutes that are to be used in the analysis. Only one of the two variations of this record may be used per data case. Variation 1 must be used when mode shape coefficients generated by the INTERPOLATION Processor are to be employed. Mode shapes may be defined directly via Variation 2 for a rigid-body aerodynamic analysis. Rotational modes for the control/tab surfaces may be defined directly via Record 15. At least one Record 14 or Record 15 must be input.

Record 14 Modal Data (optional)

Variation 1 USE Name WITH { MAIN CONTROL TAB } SURFACE Ident

Name = The name of an interpolation-coefficient matrix previously

generated via the INTERPOLATION Processor (see sec. 232.0).

{ MAIN
CONTROL
TAB }

= Key-word denoting the type of surface with which "Name" is to be associated. A particular coefficient matrix may be used with one or more of the three types of surfaces. It should be noted that each matrix is associated with a subset of nodes, all of which have the same analysis frame.

Ident = Alphanumeric word identifying the surface. "Ident" must have been previously defined by a Record 4 or 9.

This record is repeated to associate an interpolation coefficient matrix with other lifting surfaces of the aerodynamic model.

Variation 2 allows the user to define 1 to 6 rigid body modal freedoms relative to the GLOBAL reference frame.

Variation 2 RIGID BODY <x y z> Rbf1 maq1 Rbf2 maq2 ...

x,y,z = GLOBAL X-Y-Z coordinates defining the reference point to be used in generating the rigid body modes. Default: The origin of the GLOBAL frame is used as the reference point.

Rbfi = Key-word denoting the i-th rigid body freedom selected from the list TX, TY, TZ, RX, RY and RZ. Within each of these key-words, the letter "T" is associated with a translation freedom and "R" is associated with a rotation freedom relative to the GLOBAL X-Y-Z triad.

maq1 = The relative magnitude of the i-th rigid body freedom with respect to the reference point. Items "Rbfi" and "maq1" are input in pairs, 1 to 6 times, in an order which defines

the sequence of rigid body modes used in the aerodynamic analysis.

Default Record: No vibration modes or "AIC" modes are used in the analysis.

Record 15 Control and Tab Surface Rotation Modes (optional)

Modal vectors representing unit-amplitude oscillatory rotating motions (unit rotation modes) of selected combinations of control or tab surfaces about their hinge lines are defined by this record. These vectors are added to the sequence of modes identified by previous Records 14. Unit-rotation modes specified for control surfaces are added first followed by those specified for tab surfaces.

UNITROT	{ CONTROL TAB }	SURFACE List
---------	--------------------	--------------

List = List of alphanumeric identifiers denoting which CONTROL or TAB surfaces are to have an associated unit rotation mode. All identifiers must have been previously defined by a Record 9.

Default Record: No unit rotation modes are generated.

This record may be repeated. Each record of this type generates one additional modal vector.

Record 16 End Modal Subset

END MODAL DATA

This record indicates that all modal data have been defined for the case identified by Record 2.

104.1.3 Option Data Subset (optional)

This data subset (Records 17-28) contains empirical information for modification of all or part of the following theoretically derived data.

- a) Aerodynamic strip coefficients
- b) Lift distribution coefficients

c) Pitching-moment distribution coefficients

If no modifications are to be made to these calculated data, Records 17-28 should not be input.

Record 17 Begin Option Subset

BEGIN OPTION DATA

Records 18-19, 20-21, 22-25 and 26-27 are input in groups, only as required, to modify the calculated aerodynamic strip coefficients directly or to modify the lift and pitching moment coefficients used in calculation of the strip coefficients.

Records 18 and 19 are used to modify strip coefficient data directly within partitions of the oscillatory aerodynamic derivative matrix, [D]. Each partition is identified by a 2-character alphanumeric label as described below.

$$[D] = \begin{bmatrix} LH & LA & LB & LG \\ MH & MA & MB & MG \\ CH & CA & CB & CG \\ TH & TA & TB & TG \end{bmatrix}$$

Each letter of a partition label denotes the following:

L--lift on a main surface

M--moment about the elastic axis of a main surface

C--moment about the hinge line of a control surface

T--moment about the hinge line of a tab surface

H--the lift or moment is due to a unit-amplitude oscillatory displacement of the main surface

A, B, G--the lift or moment is due to a unit-amplitude oscillatory rotation of the main, control and tab surface, respectively.

All rotations and moments are about axes perpendicular to the free stream and translations are normal to the main surface. Each partition is a diagonal matrix whose dimension is equal to the number of aerodynamic strips associated with all the main surfaces defined for the case identified by Record 2. Each strip is identified by a unique integer number. Sequential numbers beginning with unity are automatically assigned to the strips beginning with the most inboard strip to the most outboard strip of each main surface. The spanwise direction is implicitly defined via the data input by Records 5-7. Main surfaces are ordered according to the sequence in which they are defined by Record 4.

Record 18 Direct Empirical Data

DIRECT DATA

Record 19 Aerodynamic Data Modifications

{REPLACE } PARTITION Pid STRIPS s1 <TO sn> value1
 {SCALE }
 <value2... valuen>

- Pid = Alphanumeric label of partition to be modified (e.g., LH, TA).
- s1 = Integer number of the first strip whose data are to be modified.
- sn = Integer number of the last strip whose data are to be modified.
 "valuei" data modifications are input for strips identified by "s1, s1+1, ...sn."
 Default: Data modification is only for strip "s1." Only "value1" should be input.
- valuei = The quantity that is to REPLACE or SCALE the values within partition "Pid" associated with the i-th strip between s1 and sn (strip s1+i-1).

This record may be repeated.

Records 20 and 21 allow the user to directly modify the lift curve slope "mo" from the theoretical value of 2π .

Record 20 Lift Curve Slope Data

MZERO DATA

Record 21 Lift Curve Slope Data Modifications

SURFACE Ident eta1 value1 eta2 value2 ...

- Ident = Alphanumeric identifier of the main surface whose lift curve slope is to be modified. "Ident" must have been previously defined by Record 4.
- eta1 = Fraction of the surface span, $0.0 \leq \text{eta1} \leq 1.0$, at which "value1" is defined. "eta1" is measured in the plane of the main surface and in a spanwise direction consistent with that used for the data defined by Records 5-7.
- value1 = The value of "mo" at "eta1."

This record may be repeated.

Records 22-27 allow the user to incorporate experimental strip coefficient data from one or more experiments into the calculation of the oscillatory aerodynamic derivative matrix [D]. Each test may be used to modify the lift curve slope "mo" and/or aerodynamic centers for some or all of the surfaces present in that test. The test should have taken the form of rotating a surface or configuration of surfaces about an axis perpendicular to the free stream. Test measurements at several spanwise locations on each surface should have been made such that the lift coefficients are obtained.

For example, it may be desirable to replace the theoretical lift distribution on a wing with experimental data obtained using a wing with nacelles. Only the lift distribution on the wing was measured. Record 24 is used to specify that only the wing surface data are to be modified. Record 25 permits the experimental wing lift distribution to be input. A subsequent Record 25 may be used to specify that the surfaces representing the nacelles should be taken into account, using the theoretically derived nacelle-wing interference.

Record 22 Experimental Lift Data

LIFT DATA

Record 23 Test Parameters

$\left\langle \begin{array}{l} \text{SYMMETRIC} \\ \text{NONSymm} \\ \text{ANTISymm} \end{array} \right\rangle$ TEST tid <THETA theta> <ALPHA alpha>

$\left\langle \begin{array}{l} \text{SYMMETRIC} \\ \text{NONSymm} \\ \text{ANTISymm} \end{array} \right\rangle$

= Key-word indicating the symmetry of the test data to be used to calculate "mo." SYMMETRIC or ANTISYMMETRIC test data are ignored when the NONSYMMETRIC option of the AF1 Processor is used.
Default: It is assumed that the test data are compatible with the symmetry option for which the AF1 Processor is executed.

tid = Alphanumeric word of 1 to 5 characters that identifies the test data.

theta = Angle in radians from the GLOBAL X-Y plane to the test rotation axis measured according to the right-hand rule relative to the X-axis.
Default: theta = 0.0

alpha = Magnitude of the rotation about the test rotation axis in radians.
Default: alpha = 1.0

Record 24 Modified Surfaces

CHANGE Ident1 Ident2 ...

Ident_i = Alphanumeric identifier of the i-th main surface whose lift data are to be replaced by data from this test. "Ident_i" must have been previously defined by Record 4.

Record 25 Contributing Surfaces

This record is input for each surface that was used in obtaining the data for test "tid." A record of this type must be input for each surface identified by Record 24. Aerodynamic interference effects from other surfaces are included via additional records of this type.

SURFACE Ident <eta1 value1 eta2 value2 ...>

- Ident = Alphanumeric identifier of the surface contributing to the test data. "Ident" must have been previously defined by Record 4.
- eta1 = Fraction of the surface span, $0.0 \leq \text{eta1} \leq 1.0$, at which "value1" is defined. "eta1" is measured in the plane of the main surface and in a spanwise direction consistent with that used for the data defined by Records 5-7.
- value1 = Value of the sectional lift ($c \cdot C_L a$) where "c" is the section chord and "a" is "alpha" defined by Record 23 at "eta1." The sign and units of this item must be consistent with those used to define "alpha" in Record 23. "eta1" and "value1" must be input if "Ident" is specified via Record 24.

Default Items: The aerodynamic contribution of "Ident" to the test data is based on the theoretically-calculated lift distribution and any modifications of "mo" defined by previous Records 23-25. (All direct modifications to "mo" resulting from Records 20 and 21 are used in this calculation).

Record 26 Experimental Moment Data

MOMENT DATA

Record 27 Sectional Pitching Moment Data

This record permits an experimentally derived pitching moment distribution to be used to replace the theoretical distribution on a particular surface.

SURFACE Ident eta1 value1 eta2 value2 ...

- Ident = Alphanumeric identifier of the main surface whose moment distribution is to be modified. "Ident" must have been previously defined by Record 4.
- eta1 = Fraction of the surface span, $0.0 \leq \text{eta1} \leq 1.0$, at which "value1" is defined. "eta1" is measured in the plane of the main surface and in a spanwise direction consistent with that used for the data defined by Records 5-7.
- value1 = Fraction of the local chord, $0.0 \leq \text{value1} \leq 1.0$, which defines the location of the local aerodynamic center relative to the quarter-chord. This quantity is positive if the aerodynamic center is upstream from the quarter-chord.

This record may be repeated.

Record 28 End Option Subset

END OPTION DATA

Additional cases may be defined by repeating Records 2-28 with a unique identification number assigned to each case by Record 2.

Record 29 End AF1 Data Set

END AF1 DATA

Additional cases may be input by repeating Records 1-29.

Table 104 -1. Summary of AF1 Data Records

Reference Page	Data Records
104.2	BEGIN AF1 DATA
104.2	CASE n
104.5	BEGIN GEOMETRY DATA
104.5	MAIN SURFACE ident
104.5	LEADING EDGE x1 y1 z1 x2 y2 z2 ...
104.5	TRAILING EDGE x1 y1 z1 x2 y2 z2 ...
104.6	ELASTIC AXIS x1 y1 z1 x2 y2 z2 ...
104.6	STRIP { FRACTIONS } { DISTANCES } d1 d2 ...
104.7	{ CONTROL } { TAB } SURFACE ident
104.7	HINGE LINE xi yi zi xo yo zo
104.7	LEADING EDGE xi yi zi xo yo zo
104.8	END GEOMETRY DATA
104.8	BEGIN MODAL DATA
104.8	USE Name WITH { MAIN } { CONTROL } SURFACE Ident { TAB }
104.9	RIGID BODY < x y z > Rbf1 mag1 Rbf2 mag2 ...
104.10	UNITROT { CONTROL } { TAB } SURFACE List
104.10	END MODAL DATA
104.11	BEGIN OPTION DATA
104.12	DIRECT DATA
104.12	{ REPLACE } { SCALE } PARTITION Pid STRIPS si <T0 sn> value1 <value2 ...>
104.13	MZERO DATA
104.13	SURFACE Ident etal value1 eta2 value2 ...
104.14	LIFT DATA
104.14	{ SYMMETRIC } { NONSYMM } { ANTISYMM } TEST tid <THETA theta> <ALPHA alpha>
104.14	CHANGE Ident1 Ident2 ...
104.15	SURFACE Ident <etal value1 eta2 value2 ...>
104.15	MOMENT DATA
104.16	SURFACE Ident etal value1 eta2 value2 ...
104.16	END OPTION DATA
104.16	END AF1 DATA

106.0 BOUNDARY CONDITION DATA

Boundary condition (BC) data for a structural model are comprised of the following:

- a) Labels for referencing nodal freedom kinematics and nodal forces;
- b) Retained-freedom order specification;
- c) Nodal freedom activities;
- d) Specified support displacements.

All kinematic input data are defined and effected relative to the analysis frames of the nodes as specified via the nodal data. Only the nodal data set (sec. 146.0) must be input prior the BC data set.

Each group of constraint conditions (a), b) and c) above) to be applied to a structural model is referenced as a STAGE. A maximum of 10 different STAGES and SUPSTAGES (superposition STAGES defined by the stress data set--sec. 154.0) may be defined per nodal data set. Specified support displacements, if required, may be input partially or wholly within the BC data set and/or within the loads data set (sec. 134.0). Each group of displacements is identified by one or more load case labels and is associated with a particular STAGE and nodal SET.

Printout of the STAGE data may be requested from the BC Postprocessor (sec. 206.0), whereas printout of all nodal displacements (including support displacements) may be requested from the STRESS Postprocessor (sec. 254.2).

PROCESSED FROM SLIPS NOT FILMED

106.1 INPUT DATA

Record 1 Data Set Identification

BEGIN BC DATA

This record initiates execution of the BC (Boundary Condition) Preprocessor.

Within this data set, optional Records 2 through 6 may be input in any order. They must be input before Record 7, however. Only Record 7 is required.

Record 2 Set/Stage Identification (optional)

<SET Se> <STAGE st>

Se = Integer identifying a nodal SET to which the BC data are to be applied.
Default Items: The SET number specified by the last record of this type or SET 1.

st = Unique stage number (integer) in the range 1 through 10, inclusive, to be associated with SET "Se."
Default Items: STAGE 1

Default Record: SET 1 STAGE 1

Record 3 Freedom Activity Labels (optional)

FREEDOM ACTIVITIES a1 a2 a3

a1, a2, a3 = Three alphanumeric words of 1 to 7 characters for referencing freedom activities. These labels are associated with the three row partitions of the stiffness equilibrium matrix equations, respectively, as shown below.

Row Partition
Number

$$\begin{array}{c} 1 \\ \text{-----} \\ 2 \\ \text{-----} \\ 3 \end{array} \begin{bmatrix} K11 & K12 & K13 \\ K21 & K22 & K23 \\ K31 & K32 & K33 \end{bmatrix} \begin{bmatrix} D1 \\ D2 \\ D3 \end{bmatrix} = \begin{bmatrix} L1 \\ L2 \\ L3 \end{bmatrix}$$

K_{ij} , D_i and L_i denote the matrix partitions of the gross stiffness, displacement and loads matrices, respectively. The association of freedom activity labels with the three row partitions of the equilibrium equations is further illustrated in the following table.

Partition Row Number	Freedom Activity Label		Description
	User-Specified	Default	
1	a1	FREE	Unconstrained freedoms that are statically condensed (reduced) from gross matrices during generation of reduced matrices
2	a2	RETAIN	Unconstrained freedoms that are retained in reduced stiffness, flexibility, loads and mass matrices
3	a3	SUPPORT	Freedoms for which displacements are specified; zero displacements are enforced by default

If this record is not input, the default freedom activity labels are FREE, RETAIN and SUPPORT for partitions 1 through 3, respectively. The selected labels remain in effect for multiple SETS and/or STAGES until redefined.

Record 4 Freedom Labels (optional)

FREEDOM LABELS

Records 4 and 5 allow user-selected alphanumeric labels to be associated with the nodal displacements (kinematics) and/or forces. These labels are used for input and printing of BC and loads data associated with the specified SETS and STAGES. The default labels associated with the nodal freedoms for the three types of reference frames (rectangular, cylindrical, spherical) are shown in the table below. Note that all references to REC frames within this data set include the GLOBAL frame in addition to any local rectangular frames defined via the nodal data.

NODAL FREEDOM NUMBER	NODAL REFERENCE FRAME TYPE					
	RECTangular		CYLindrical		SPHERical	
	Kinematic Label	Force Label	Kinematic Label	Force Label	Kinematic Label	Force Label
1	TX	FX	TR	FR	TR	FR
2	TY	FY	TT	FT	TT	FT
3	TZ	FZ	TZ	FZ	TP	FP
4	RX	MX	RR	MR	RR	MR
5	RY	MY	RT	MT	RT	MT
6	RZ	MZ	RZ	MZ	RP	MP

The first characters, T and R, of the default displacement labels denote translations and rotations, respectively. The first characters, F and M, of the default force labels denote forces and moments, respectively. The second character within each label denotes one of the reference frame axes as illustrated in figure 100-1.

Record 5 Kinematic Freedom and Force Labels

Records of this type are input only if Record 4 is input.

<pre> { REC } { CYL } {<DISP dlist> <FORCE flist>} { SPH } </pre>

```

{ REC }
{ CYL }
{ SPH }

```

= Key-word identifying a type of reference frame. All reference frames of this type included in the specified SET/STAGE will be affected.

dlist = A list of six, 2-character, alphanumeric words to be used as the kinematic labels associated with freedoms 1 through 6.

flist = A list of six, 2-character, alphanumeric words to be used as the force labels associated with freedoms 1 through 6.

The default labels are shown in the foregoing table. Labels remain in effect until redefined.

Record 6 Retained-Freedom Order Specification (optional)

ORDER RETAIN { <table style="display: inline-table; vertical-align: middle;"> <tr> <td style="padding: 2px 5px;">AS LISTED</td> </tr> <tr> <td style="padding: 2px 5px;">BY INTERNALID</td> </tr> <tr> <td style="padding: 2px 5px;">BY USERID</td> </tr> </table> }	AS LISTED	BY INTERNALID	BY USERID
AS LISTED			
BY INTERNALID			
BY USERID			

This record denotes the sequence in which the retained degrees of freedom will be ordered for the identified SET/STAGE. The options are as follows:

- AS LISTED -- the freedoms will be ordered as they are listed in Record 7, Variation 1. If any freedoms are retained via Variation 2 of Record 7, the retained order indicator is automatically set to "BY INTERNALID."

- BY INTERNALID -- the freedoms specified by Record 7 will be ordered in an increasing internal node number sequence.

- BY USERID -- the freedoms specified by Record 7 will be ordered in an increasing user node number sequence.

Multiple freedoms retained for a single node via the INTERNALID or USERID options are ordered according to the nodal freedom numbers 1 through 6 shown in the foregoing table.

Default Record: ORDER RETAIN AS LISTED

Record 7 BC Specifications

This record, Variation 1 and/or 2, may be repeated to identify all constraints for a particular STAGE. Multiple STAGES for a particular model are effected by repeating records 2-7.

If more than one activity is specified for a particular freedom, the last one specified takes precedence. It is sometimes convenient, for example, to RETAIN or SUPPORT a large group of freedoms using a data-generation capability, and then FREE selected freedoms within the group. If no activity is specified for a freedom, it is included in the FREE partition of the cross stiffness matrix. An inactive freedom (one that has zero stiffness) is automatically ignored. An activity is imposed on the freedom of a Shear node only if the second item of this record identifies the first kinematic freedom of the GLOBAL analysis frame.

Variation 1

Activity	{ Dlist ALL }	FOR	ATlist(n)
----------	------------------	-----	-----------

- Activity = One of the freedom activity labels defined by Record 3.
- Dlist = List of 1 to 6 kinematic labels as defined by "dlist" in Record 5. If the word ALL is selected for this item, all six nodal freedoms are implied.
- ATlist(n) = ATLAS list of user node numbers (integers). The specified activity is imposed on the "Dlist" freedoms of these nodes. Only those freedoms of the ATlist nodes that have kinematic labels identified by "Dlist" are affected.

If the "AS LISTED" option is selected via Record 6, retained freedoms are ordered in the sequence specified via Dlist and ATlist. The order is defined as all freedoms in Dlist for the first node in ATlist, followed by the same Dlist freedoms for the second node in ATlist, etc. If ALL is selected, the order of the six retained freedoms will be as shown in the foregoing table.

Variation 2 of the BC specification record allows activities to be effected on selected freedoms of nodes that lie in one of the principal surfaces of a particular analysis frame. A principal surface is defined as a surface in which one of the coordinates is invariant. The principal surfaces associated with the coordinates of the analysis frames are identified by the integers 1 to 3 as illustrated in the following table (also see fig. 100-1).

SURFACE IDENTIFIER	ANALYSIS FRAME TYPE		
	REC	CYL	SPH
1	x, X	r	ρ
2	y, Y	θ	θ
3	z, Z	z	ϕ

The location and configuration of a surface is defined via a selected node "Nn" and its associated analysis frame.

The surface passes through "Nn." The selected activity is performed on the specified freedoms of all nodes that lie in the surface and have the same analysis frame as "Nn." A node is considered to be in the defined surface if the perpendicular distance from the node to the surface is less than 10^{-8} units.

Freedoms may be specified via Dlist or the word ALL as in Variation 1. Two additional key-words, SYMM and ASYM, may be used in Variation 2 to designate certain combinations of freedoms upon which the specified activity is to be performed. These key-words are associated with symmetrical and anti-symmetrical boundary conditions relative to the prescribed surface. The combinations of freedoms associated with these key-words and the surface identifiers are shown in the table below.

SURFACE IDENTIFIER	SYMMETRY KEY-WORD	NODAL FREEDOM NUMBER					
		1	2	3	4	5	6
1	SYMM		✓	✓	✓		
	ASYM	✓				✓	✓
2	SYMM	✓		✓		✓	
	ASYM		✓		✓		✓
3	SYMM	✓	✓				✓
	ASYM			✓	✓	✓	

Variation 2 Activity $\left\{ \begin{array}{l} \text{Dlist} \\ \text{ALL} \\ \text{SYMM} \\ \text{ASYM} \end{array} \right\}$ IN SURFACE $\left\{ \begin{array}{l} 1 \\ 2 \\ 3 \end{array} \right\}$ <THROUGH Nn>

Activity }
Dlist }
ALL }

= Same as defined for Variation 1

SYMM }
ASYM }

= Key-word SYMM or ASYM may be used to denote a combination of freedoms as discussed above.

$\left\{ \begin{array}{l} 1 \\ 2 \\ 3 \end{array} \right\}$

= Integer identifying the principal surface in which the activity is to be effected (see foregoing table).

Nn = User node number (integer). The surface in which the activity is to be performed passes through this node.

Default: The surface passes through the GLOBAL origin and the specified activity is performed on the freedoms of nodes that have a GLOBAL analysis frame.

Example Use of Record 7:

A structural model, which is symmetrical relative to the GLOBAL X-Z plane, is to be constrained in the X-Z plane such that anti-symmetrical displacements of the model relative to this plane are zero. Freedoms TZ and RY are to be retained for nodes 8 and 50 through 65. Additionally, the TX freedom for nodes 8 and 32 are to be retained. The data records could be input as:

```
SUPPORT ASYM IN SURFACE 2 /  
RETAIN TZ RY FOR 8 50 TO 65 /  
RETAIN TX FOR 8 32 /
```

106.1.1 Support Displacement Data Subset

This data subset, Records 8-16, may be used to input specified displacements. These data may be input partially or wholly in the BC and/or loads data sets. Displacements input by BC data and by loads data for a particular SET and STAGE (Record 4) are cumulated. Multiple displacements specified for the same freedom and load case are cumulated.

Displacements may be specified only for freedoms which have been previously SUPPORTED. All displacements are defined and effected relative to the analysis frame of the associated node. If no displacement is specified for a SUPPORTED freedom, it is rigidly fixed (zero displacement is enforced).

Each group of specified displacements is uniquely identified by a load case label. Additional load case titling and combinations of defined load cases can be input by the loads data (see sec. 134.0). Specified lists of displacements may be associated with a group of load cases (Records 9, 10 and 11), a group of nodes (Records 9, 12 and 13) or a group of kinematic freedoms (Records 14 and 15). Rotational displacements are input in radians.

Record 8 Specified Displacement Data Subset

BEGIN SUPPORT DISPLACEMENT DATA

Record 9 Nodal-Freedom Order (optional)

This record defines the nodal-freedom correspondence list associated with lists of displacements input by Records 11 and/or 13.

ORDER Nf1 Nf2 ...

Nf1, Nf2, ... = List of 1 to 6, 2-character, alphanumeric identifiers defining the order and number of kinematic freedoms associated with the displacements input via "dlist" in Records 11 and/or 13. Only the nodal freedoms with non-zero enforced displacements need be included in this list. The i-th identifier "Nfi" in this list must be one of the following:

Nfi -- Either one of the kinematic labels selected via Record 5 for a specific type of analysis frame (REC, CYL or SPH) or one of the words Fj, j=1, 6 which denote the six ordered nodal freedoms (see Record 4) associated with all analysis frame types. If the former type of identifier is used, all nodes with specified displacements must have the same list of analysis-frame kinematic labels. However, if "Fj" words are used, the kinematic labels associated with the referenced nodes are immaterial.

Default Record: The order and number of nodal-freedoms defined by the last record of this type or Fj; j = 1,6. If this record is not used, the order in which displacements are specified via Records 11 and/or 13 must be consistent with the default order discussed under Record 4 (e.g., TX, TY, TZ, RX, RY and RZ for REC analysis frames).

Records of this type may be used any number of times within this data subset.

106.1.1.1 Displacements Grouped by Load Cases

Records 10 and 11 allow specified displacements to be associated with a group of load cases.

Record 10 Load Case Labels

CASF ATlist(c)

ATlist(c) = ATLAS list defining up to 250 user-assigned load case labels (positive integers or alphanumeric words with 1 to 7 characters). The word ALL, the 2-character words Fi, where i=1, 6 and any of the force and kinematic labels selected via Record 5 may not be used.

Displacements specified by subsequent Records 11 are attributed to each of the load cases defined by ATlist(c).

Record 11 Node and Displacement Specification

ATlist(n)	{ dlist Nf1 d1 Nf2 d2 ... }
-----------	--------------------------------

ATlist(n) = ATLAS list of up to 250 node numbers (integers) to which the subsequent displacements apply. Non-existent nodes identified by ATlist(n) are not allowed.

dlist = List of 1 to 6 displacements associated with the sequence and number of kinematic freedoms identified by the last Record 9. Trailing zeros need not be input.

Nfi, di = The i-th kinematic label and displacement in this record denoting which freedom is to have a displacement of "di" imposed. The format of Nri is the same as defined for Record 9. These items are input in pairs in an order that is independent of Record 9.

This record or Records 10 and 11 may be repeated.

Example Use of Records 10 and 11:

For load cases ABC and B20, nodes 35 and 100 through 110 have a TX translation of 5. and an RZ rotation of 0.1 radians. Nodes 36 and 200 through 210 have a TX of 10. and an RZ of 0.2 radians. Additionally, nodes 37 and 300 through 310 have a TX of

15. and an RZ of 0.3 radians. The data records could be input as:

```
CASE ABC B20 /
35 100 TO 110 TX 5. RZ 0.1 /
**2 1 100 0 100 0 5. 0 0.1 /
```

106.1.1.2 Displacements Grouped by Nodes

Records 12 and 13 allow specified displacements to be associated with a group of nodes.

Record 12 Node List

NODE ATlist (n)

ATlist (n) = ATLAS list of up to 250 node numbers (integers) to which subsequent Records 13 apply. Non-existent nodes identified by ATlist (n) are not allowed.

Displacements specified by subsequent Records 13 are applied to all nodes in ATlist (n).

Record 13 Load Case and Displacement Specification

**ATlist (c) { dlist
Nf1 d1 Nf2 d2 ... }**

ATlist (c) = Same as defined for Record 10

dlist
Nfi
di } = Same as defined for Record 11

All displacements specified within this record are attributed to each of the load cases defined by ATlist (c).

This record or Records 12 and 13 may be repeated.

Example Use of Records 12 and 13:

Nodes 5, 7, 9, 11 and 45 for load cases 1, 2, 3, H1, H2 and H3 have a TY displacement equal to 6 times the integer part of the load case label and an RX value equal to twice the integer part of the load case label. The data records could be input as:

```
NODE 5 TO 11 BY 2 45 /
1 H1 TY 6. RX 2. /
**2 1 1 0 6. 0 2. /
```

106.1.1.3 Displacements Grouped by Nodal Freedoms

Records 14 and 15 allow specified displacements to be associated with a group of kinematic freedoms.

Record 14 Kinematic Freedom List

FREEDOM ATlist (fn)

ATlist (fn) = ATLAS list defining up to 250 alphanumeric words--each word is comprised of a kinematic label, as specified by Record 5, followed by a node number. An ATlist (fn) could be TX11 TO TX88 which denotes the X-axis translational freedoms for nodes 11 through 88. Non-existent node numbers referenced by words generated by ATlist (fn) are not allowed.

Record 15 Load Case and Displacement Specification

ATlist (c) dlist

ATlist (c) = Same as defined for Record 10

dlist = List of displacements to be enforced on the freedoms as listed by Record 14. The number and sequence of the displacements in "dlist" must correspond to the number and sequence of freedoms specified by Record 14.

This record or Records 14 and 15 may be repeated.

Example Use of Records 14 and 15:

TX displacements are to be specified at nodes 1 through 5 and 20. For load case 10, the displacements are 1, 2, 13, 4, -5 and 8. For load case JACK, the displacements are 7, 12, 15, -2, 6 and 2. The data records could be input as:

```
FREEDOM TX1 TO TX 20 /  
10 1. 2. 13. 4. -5 8. /  
JACK 7. 12. 15. -2. 6. 2. /
```

Record 16 End Support Displacement Data Subset

END SUPPORT DISPLACEMENT DATA

Additional Records 2-7 or Records 2-16 may be input.

Record 17 End BC Data Set

END BC DATA

Additional BC data sets may be input by repeating Records 1-17.

Table 106-1. Summary of Boundary Condition Data Records

Reference Page	Data Records
106.2	BEGIN BC DATA
106.2	<SET Se> <STAGE st>
106.2	FREEDOM ACTIVITIES a1 a2 a3
106.3	FREEDOM LABELS
106.4	$\left\{ \begin{array}{l} \text{REC} \\ \text{CYL} \\ \text{SPH} \end{array} \right\} \left\{ \langle \text{DISP dlist} \rangle \langle \text{FORCE flist} \rangle \right\}$
106.5	$\text{ORDER RETAIN} \left\{ \begin{array}{l} \text{AS LISTED} \\ \text{BY INTERNALID} \\ \text{BY USERID} \end{array} \right\}$
106.6	$\text{Activity} \left\{ \begin{array}{l} \text{Dlist} \\ \text{ALL} \end{array} \right\} \text{FOR ATlist}(n)$
106.7	$\text{Activity} \left\{ \begin{array}{l} \text{Dlist} \\ \text{ALL} \\ \text{SYMM} \\ \text{ASYM} \end{array} \right\} \text{IN SURFACE} \left\{ \begin{array}{l} 1 \\ 2 \\ 3 \end{array} \right\} \langle \text{THROUGH Nn} \rangle$
106.9	BEGIN SUPPORT DISPLACEMENT DATA
106.9	ORDER Nf1 Nf2 . . .
106.10	CASE ATlist(c)
106.10	$\text{ATlist}(n) \left\{ \begin{array}{l} \text{dlist} \\ \text{Nf1 d1 Nf2 d2} \dots \end{array} \right\}$
106.11	NODE ATlist(n)
106.11	$\text{ATlist}(c) \left\{ \begin{array}{l} \text{dlist} \\ \text{Nf1 d1 Nf2 d2} \dots \end{array} \right\}$
106.12	FREEDOM ATlist(fn)
106.12	ATlist(c) dlist
106.13	END SUPPORT DISPLACEMENT DATA
106.13	END BC DATA

112.0 DESIGN DATA

The stiffness element data required to perform a fully-stressed structural design or regional optimization of composite-built structures are defined by this data set. The design capabilities and analysis techniques of the DESIGN Processor are presented in section 212.0. The element properties of the structural model defined by a stiffness SET (sec. 152.0) are modified (resized) according to user specifications. Resize calculations are performed by design algorithms according to specified geometric and margin-of-safety constraints on selected elements.

The detail and material data sets, if required, and the nodal, stiffness, BC, loads and subset-definition data sets must be input prior to this data set in the input stream.

The design data set is divided into the following data subsets:

- a) Table--shear and compression buckling allowable stresses
- b) Modulus--shear moduli and Young's moduli of elasticity for calculation of local buckling allowables for "plate-like" elements
- c) Property--properties for conversion of element loads as calculated by the STRESS Processor to element stresses
- d) Fixed--selected properties of elements which are to be held at a fixed level by the DESIGN Processor during resizing
- e) Lower-Bound--minimum or lower-bound element property values
- f) Upper-Bound--maximum or upper-bound element property values
- g) Margin--element property margins of safety
- h) Sizing--element material properties and allowable stresses
- i) Restrain-Sizing--selected elements that are not to be resized by the DESIGN Processor

- j) Optimization--problem-definition data for regional optimization of composite structures.
- k) Variable-Constraints--design-variable constraints (lamina-thickness coupling expressions) to be used in solving the Optimization problem (j)
- l) Smoothing--regions of the structural model in which the element properties resulting from design calculations are to be smoothed
- m) Buckling--parameters defining panel-buckling interaction formula for "plate-like" elements
- n) Loads--design load cases for which elements are to be resized
- o) Superposition--design load cases formed by superimposing previously-defined load cases
- p) Thermal--design load cases formed by superimposing one mechanical and one thermal load case previously defined by the loads data set (sec. 134.0).

Input of these data subsets is optional except, as indicated below, certain data are required for the particular type of design activity to be performed:

- a) Fully-Stressed Design--the Sizing data subset is required. Additionally, if BEAM elements are included in the stiffness SET, the Property data subset is required.
- b) Thermal Fully-Stressed Design--the Sizing and Property data subsets are required as discussed for (a). In addition, the Thermal data subset is required.
- c) Regional Composite Optimization--only the Optimization data subset is required.

Printout of the input design data may be requested from the DESIGN Postprocessor (sec. 212.1).

112.1 INPUT DATA

The following notation is used to identify selected stiffness elements within many of the design-data records. The items, "Elid" and "Etype," are always used together.

Elid = Identifier of an element or a subset of elements. The options for "Elid" are:

Exxx--The name of a stiffness element subset previously defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0).

Nxxxxx--The letter N followed by a 1 to 5 digit, user element number (sec. 152.0).

Ixxxxx--The letter I followed by a 1 to 5 digit internal element number assigned via the selected node reorder specification (sec. 146.0).

Default: All elements associated with the specified stiffness SET.

Etype = A key-word or its equivalent integer that denotes a particular type of element included in "Elid." If "Elid" identifies a single element, "Etype" is ignored. The options for "Etype" are:

<u>Key-Word</u>	<u>Integer-Equivalent</u>
ROD	1
BEAM	2
SPAR	3
COVER	4
PLATE	5
GPLATE	6
BRICK	8
SROD	10
SPLATE	11
CPLATE	12
CCOVER	13

Default: ALL. The key-word ALL denotes all element types included in "Elid."

5th of these items, <Elid> <Eltype>, may be defaulted within all the data records except Records 23 and 26. The remaining data defined by the corresponding records apply to all elements denoted by "Elid" and "Eltype."

Record 1 Data Set Identification

BEGIN DESIGN DATA

This record initiates execution of the DESIGN Preprocessor.

Record 2 Data Preprocessing Mode

This record allows the user to specify how data subsets of a design data set are to be treated by the preprocessor. Five different preprocessing modes are available to indicate whether a) particular data subsets are input for the first time, b) selected subsets are being input to replace previously-defined subsets of the same type or c) selected subset data are being input to modify corresponding previously-defined subset data. For all modes, the "Property" and "Sizing" data subsets as required for fully-stressed design must either be included in the data set or they must have been defined previously. Details of the various preprocessing modes are described by table 112-1. Note that the 16 input data subsets are grouped into three blocks:

- a) "Table" data applicable to multiple stiffness SETS and boundary condition (BC) STAGES,
- b) "SET-level" data applicable to multiple STAGES,
- c) "STAGE-level" data associated with a particular SET.

MODE Code

Code = Integer in the range 1 to 5 denoting how the input data subsets are to be preprocessed. The options for "Code" are:

- 1--No input data have been defined previously.
- 2--"Table" data are to be used to modify previously-input "Table" data, whereas the other input data have not been defined previously.

3--"Table" data are to be used to modify previously-input "Table" data, input SET-level data subsets are to replace corresponding data subsets which may have been defined previously and no STAGE-level data have been defined previously.

4--"Table" and SET-level subset data are to be used to modify previously-input subsets of the same type, whereas any STAGE-level subset data are to replace corresponding data which may have been defined previously.

5--All data are to be used to modify the corresponding, previously-input subset data.

Caution: The number of design load cases specified by Records 38-40 for a particular SET and STAGE may not be increased when mode 4 or 5 is selected.

Table 112-1. Design Data Preprocessing Modes

INPUT DATA SUBSET		DATA PREPROCESSING MODE									
		1		2		3		4		5	
		NEW	MODIFY	NEW	MODIFY	NEW	MODIFY	NEW	MODIFY	NEW	MODIFY
Table Data	TABLE	*			***		***		***		***
	MODULUS	*			***		***		***		***
Stiffness SET Level Data	PROPERTY	*		*		**1			***		***
	FIXED	*		*		**1			***		***
	LOWER-BOUND	*		*		**2			***		***
	UPPER-BOUND	*		*		**2			***		***
	MARGIN	*		*		**			***		***
	SIZING	*		*		**1			***		***
	RESTRAIN-SIZING	*		*		**			***		***
	OPTIMIZATION	*		*		**3			***		***
	VARIABLE-CONSTRAINTS	*		*		**3			***		***
	SMOOTHING	*		*		**			***		***
BUCKLING	*		*		**			***		***	
BC STAGE Level Data	LOADS	*		*		*		**4			***
	SUPERPOSITION	*		*		*		**4			***
	THERMAL-LC	*		*		*		**			***

* The input data subset has not been input previously; a new subset is being input. Only the Property and Sizing data subsets, as required, must be input for modes 1 and 2.

** The input data subset may have been input previously. If corresponding data were input previously, they are replaced by the new subset being input. Thus, selected data subsets may be input via modes 3 and 4. If one of the subsets tagged with **1 is to be input, the other two subsets tagged with **1 must also be input. This is also a requirement for the subsets tagged with **2, **3 and **4.

*** The input data subset is to be used only to modify the corresponding, previously-input subset. New data may be added to the old subset data or the old data may be replaced on a one-to-one basis.

112.1.1 Table Design Data

Allowable buckling stresses, shear moduli and Young's moduli or elasticity for local buckling calculations associated with plate-like elements are established by Records 3-8. "Table" data may be used for multiple SET/STAGE models.

112.1.1.1 Table Data Subset (optional)

Buckling allowable stresses applicable to shear and compression plate-like elements (SPAR, COVER, PLATE, GPLATE, SPLATE) are defined by this data subset. Each set of allowable stresses is identified by a material reference code. The code is subsequently associated with selected elements via the element sizing data subset (sec. 112.1.2.4). If element resizing is to be based on the allowable stresses as defined via the MATERIAL Preprocessor (sec. 140.0), this subset is not required. That is, if previously-defined allowable stresses are to be used as constant buckling allowables (not a function of combined stresses) for plate-like elements, Records 3-5 are not input. Special material allowables, defined via the MATERIAL Preprocessor, to be used as buckling allowables may be specified at a maximum of 10 temperature levels.

Record 3 Begin Table Subset

```
BEGIN TABLE DATA
```

Record 4 Material Reference Code, Gages and Allowables

Gage-dependent allowable buckling stresses associated with a particular temperature level are identified and defined by this record. Specified material reference codes are used subsequently as input items of element sizing data records (Record 20).

```
Bcode Tcode q1 q2 ... qn s1 s2 ... sn
```

Bcode = Material reference code of the form BCxx or BSxx where BC and BS denote compression-buckling and shear-buckling codes, respectively. The characters xx denote an integer in the range 51 to 99, inclusive.

Tcode = Temperature code of the form T followed by a signed or unsigned integer in the range -461 < temp < 1000 (e.g., T100, T+80, T-40) which denotes a temperature-level

in degrees Fahrenheit. The specified temperature is associated with the allowable stresses s_i ; $i=1,n$.

- q_1, q_2, \dots, q_n = List of "n" material gages ($n \leq 25$).
- s_1, s_2, \dots, s_n = List of "n" allowable stresses. The number and sequence of these items must correspond with the number and sequence of material gages.

This record may be repeated to define a maximum of 49 compression-allowable material tables and a maximum of 49 shear-allowable material tables. Each set of "TABLE" data is identified by a unique reference code. By repeating this record, allowable data identified by a particular reference code may be defined at a maximum of 10 temperature levels. Extraction of allowable stresses for an element with a particular gage at a particular temperature is performed by linear interpolation of the stress data relative to gage and temperature. If an element gage and/or temperature is less than the corresponding defined "table" values, the allowables defined for the smallest gage and/or temperature are used. If an element gage is greater than the largest "table" value, the allowables defined for the largest gage are used. Element temperatures greater than the largest "table" value are not allowed.

Record 5 End Table Subset (optional)

END TABLE DATA

This record indicates that all buckling allowable stress data required for subsequent referencing to selected elements via Record 20 have been input.

112.1.1.2 Modulus Data Subset (optional)

Stress-dependent moduli of elasticity and shear moduli to be used for calculation of COVER, PLATE and GPLATE element local-buckling allowables are defined by this data subset. Each set of moduli is identified by a material reference code previously established via the Material data (sec. 140.0). This code is subsequently associated with selected elements via the element sizing data subset (Record 20 of sec. 112.1.2.4). If local buckling allowables are to be based on the moduli as defined via the MATERIAL Preprocessor, this subset (Records 6-8) is not required. Special material moduli, defined via the Material input data, to be used for buckling-allowable

calculations may be specified at a maximum of 10 temperature levels.

Record 6 Begin Modulus Subset

BEGIN MODULUS DATA

Record 7 Material Reference Code, Stress-Levels and Moduli

Stress-dependent moduli of elasticity and shear moduli associated with particular temperature levels, as defined by this record, are identified by material reference codes.

Mcode Tcode s1 s2 ... sn m1 m2 ... mn

- Mcode = Material reference code of the form MExx or MGxx where ME and MG denote that Young's moduli or shear moduli, respectively, are defined by this record. The characters xx denote an integer in the range 1 to 99, inclusive. Corresponding Special material reference codes of the form M51-M99, as required, must have been defined previously by the material data (sec. 140.0).
- Tcode = Temperature code of the form T followed by a signed or unsigned integer in the range $-461 < \text{temp} < 1000$ (e.g., T100, T+80, T-50) which denotes a temperature-level in degrees Fahrenheit. The specified temperature is associated with the stress-levels and moduli "si" and "mi"; $i = 1, n$.
- s1, s2...sn = List of "n" stresses in a monotonically-increasing order ($n \leq 25$).
- m1, m2...mn = List of "n" moduli of elasticity or shear moduli as identified by "Mcode." The number and sequence of these items corresponds to the number and sequence of stresses.

This record may be repeated to define a maximum of 99 modulus of elasticity tables and 99 shear modulus tables. Each set of "MODULUS" data is identified by a unique reference code. By repeating this record, moduli identified by a particular reference code may be defined

at a maximum of 10 temperature levels. Extraction of modulus data for an element with a particular stress at a particular temperature is performed by linear interpolation of the modulus data relative to stress and temperature. If an element stress level is less than or greater than any of the specified stress values, the moduli corresponding to the smallest or largest stress "table" values are used accordingly. Element temperatures must be within the range of specified temperature "table" values.

Record 8 End Modulus Subset (optional)

END MODULUS DATA

This record indicates that all modulus-related data required for subsequent referencing to selected elements via Record 20 have been input.

112.1.2 Stiffness Set-Level Design Data

Design data associated with a SET to be resized are defined by Records 9-36.

Record 9 Structural Component Identification

SET Se

Se = Integer identifying a structural model previously-defined as a stiffness SET (sec. 152.0).

This record must precede all SET-level data subsets input by Records 10-36.

112.1.2.1 Property Data Subset

Properties of elements used to convert element loads (stress output as described in appendix B) to elemental stresses are defined by this data subset (Records 10-12). These data are required to convert BEAM element bending moments to axial stresses and to calculate SPAR element stiffener stresses. SPAR element stiffeners, which may be introduced by these data, are used only for calculation of margins of safety for design reference. This data subset is required if BEAM elements are included in the structural model.

Record 10 Begin Property Subset

BEGIN PROPERTY DATA

Record 11 Element Property Data

<Elid> <Eltype> plist

plist = List of properties for a particular type of element. The input-sequence, number and types of properties, and the input variations thereof which may be used for "plist" for each element type are summarized in table 112-2.

This record is repeated to define property data for all BEAM and SPAR elements to be resized.

Record 12 End Property Subset (optional)

END PROPERTY DATA

This record indicates that all property data required for resizing of selected BEAM and SPAR elements have been defined.

112.1.2.2 Fixed, Lower-Bound and Upper-Bound Element Property-Data Subsets (optional)

Constraints on geometrical properties of selected elements may be defined by the Fixed, Lower-Bound and Upper-Bound design data subsets (Records 13-15). "Fixed" property values are held at the specified level in resizing calculations. "Lower-Bound" and "Upper-Bound" property values define minimum and maximum element sizes, respectively, that are not to be exceeded during resizing. The specified property values for selected elements may be different from the corresponding element property values as defined via the stiffness data (sec. 152.0). The user should be cautioned that if the specified properties are different, the stresses used in the resize calculations may be inconsistent with these new properties.

Record 13 Begin Constraint Data Subset

BEGIN	<table border="1"> <tr> <td>FIXED</td> </tr> <tr> <td>LOWER-BOUND</td> </tr> <tr> <td>UPPER-BOUND</td> </tr> </table>	FIXED	LOWER-BOUND	UPPER-BOUND	DATA
FIXED					
LOWER-BOUND					
UPPER-BOUND					

This record denotes whether FIXED, LOWER-BOUND or UPPER-BOUND design-constraint element properties are defined by subsequent Records 14.

Table 112-2. Input Variations of Element Design-Stress Properties

ELEMENT TYPE		ELEMENT PROPERTIES		INPUT PROPERTY EXPANSION KEYS ①							
NAME	NO.	LABEL	DESCRIPTION	NO. INPUT PROPERTY VALUES							
				1	2	3	4	5	6	7	8
ROD	1	none									
BEAM	2	WY/A(1)	WY and WZ are section moduli: e.g. $\sigma = MY/WY$	1	1	1	1	1	1		1
		WY/A(2)		1	1	1	1	2	2		2
		WZ/A(1)		1	2	2	2	3	3		3
		WZ/A(2)		1	2	2	2	4	4		4
		CVZ(1)	Stress concentration factors for shear: e.g. $\tau_{max} = CVZ(1) * \tau_{avg}$	*	*	3	3	5	5		5
		CVZ(2)		*	*	3	3	5	5		6
		CVY(1)		*	*	3	4	5	6		7
		CVY(2)		*	*	3	4	5	6		8
SPAR	3	STSP	Stiffener spacing	1	1						
		ASTI	Stiffener area	**	2						
COVER	4	none									
PLATE	5	none									
GPLATE	6	none									
BRICK	8	none									
SROD	10	none									
SPLATE	11	none									
CPLATE	12	none									
CCOVER	13	none									

① An expansion key is the column of integers associated with a particular number of input property values for a particular type of element. Each integer denotes which one of the ordered property values defined by an input Record 11 is to be assigned to the corresponding element property shown in the table.

* Default value is 1.0

** Default stiffener area is calculated as $0.5 * STSP * (T - WEB)$. The default value for STSP, if not defined for a SPAR element, is 9.0

Record 14 Element Design-Constraint Property Data

<Elid> <Eltype> plist

plist = List of properties for a particular type of element. The input-sequence, number and types of properties that may be specified for various types of elements are summarized in table 112-3.

If a composite element type is identified by "Elid" and "Eltype," "plist" identifies the integer number of layers to be associated with each lamina of the elements. The i-th value in "plist" corresponds to the i-th lamina of each of the elements as defined via the stiffness data (sec. 152.0). If only one value is input, each lamina of each of the elements is constrained to the same number of layers. These constraints are imposed after the optimization problem specified by Records 25-27 is solved.

This record is repeated to define all constraint data of the type specified by Record 13.

Record 15 End Constraint Data Subset (optional)

END	{ FIXED LOWER-BOUND UPPER-BOUND	DATA
-----	--	------

Records 13-15 are repeated to define all FIXED, LOWER-BOUND and UPPER-BOUND element-property design data subsets.

Table 112-3. Fixed, Lower-Bound and Upper-Bound Element Property Data ①

ELEMENT TYPE		ELEMENT PROPERTIES (ref. appendix B)	
NAME	NO.	LABEL	DESCRIPTION
ROD	1	A(1) A(2)	Cross section area at end(1) Cross section area at end(2)
BEAM	2	A(1) A(2)	Cross section area at end(1) Cross section area at end(2)
SPAR	3	T-WEB FAREA1U. FAREA1L FAREA2U FAREA2L ASTI/STSP*(T-WEB)	Web thickness Upper flange area at end(1) Lower flange area at end(1) Upper flange area at end(2) Lower flange area at end(2) Stiffener ratio
COVER	4	T(1)U/T(0)U T(2)U/T(0)U T(0)U T(1)U T(2)U T(1)L/T(0)L T(2)L/T(0)L T(0)L T(1)L T(2)L	Thickness ratio for S1-dir. upper Thickness ratio for S2-dir. upper Upper plate thickness Smearred uniaxial gage in S1-dir. upper Smearred uniaxial gage in S2-dir. upper Data for lower plate analogous to data for upper plate
PLATE	5	TS(1)/T(0) TS(2)/T(0) T(0) TS(1) TS(2)	Thickness ratio for S1-dir. Thickness ratio for S2-dir. Plate thickness Smearred uniaxial gage in S1-dir. Smearred uniaxial gage in S2-dir.
GPLATE	6	T	Average membrane thickness
BRICK	8	none	
SHCD	10	A(1) A(2)	Cross section area at end(1) Cross section area at end(2)
SPLATE	11	T	Plate thickness
CPLATE	12	②	No. of layers for each lamina
CCOVER	13	②	No. of layers for each lamina

① Each of the ordered property values defined by Record 14 is assigned to the corresponding element property as listed in the table. If the number of nonzero properties to be specified for a particular element type is less than the number of corresponding properties shown in the table, an appropriate number of zero integers, in proper sequence, must be included in the input list of property values.

② If only one value is input, each lamina of each of the elements is constrained to the same number of layers. Optionally, a sequence of integers may be input to specify a different number of layers to be associated with each lamina of each of the elements.

112.1.2.3 Margin of Safety Data Subset (optional)

Margins of safety for selected element properties are defined by this data subset (Records 16-18). These data allow the user to specify that selected element property values may be increased or decreased, as required by the resize algorithm, such that nonzero margins of safety may result. A 20 percent margin of safety, for example, would allow the element property value(s) to be increased, as required by the resize algorithm, so that 20 percent larger properties may result as compared to a zero margin of safety. A new property, PROP_N, is calculated from an old (initial) property, PROP_O, such that $PROP_N = PROP_O * (1 + MSI) / (1 + MSC)$, where MSC and MSI are the calculated and input margins of safety, respectively.

Record 16 Begin Margin Subset

BEGIN MARGIN DATA

Record 17 Element-Property Margin of Safety Data

<Elid> <Eltype> plist

plist = List of property margins of safety (≥ -1.0) for a type of element. The input-sequence and number of margins of safety that may be specified for properties associated with various types of elements are summarized in table 112-4.

If a composite element type is identified by "Elid" and "Eltype," "plist" identifies the thickness margin of safety for each lamina of the elements. The i-th value in "plist" corresponds to the i-th lamina of each of the elements as defined via the stiffness data (sec. 152.0). If only one value is input, each lamina of each of the elements will have the same margin of safety. If two values are input for CCOVER elements, the first value corresponds to all lamina in the first surface and the second value corresponds to the second surface of the elements.

This record is repeated to define all element-property margin of safety data.

Table 112-4. Input Variations of Element- Property Margins of Safety

ELEMENT TYPE		ELEMENT PROPERTIES (ref. appendix B)		INPUT PROPERTY EXPANSION KEYS ①					
NAME	NO.	LABEL	DESCRIPTION	NO. INPUT PROPERTY VALUES					
				1	2	3	4	5	6
ROD	1	A(1)	Cross section area at end(1)	1	1				
		A(2)	Cross section area at end(2)	1	2				
BEAM	2	A(1)	Cross section area at end(1)	1	1	1	1		1
		A-VY(1)	Shear area (local y-dir) at end(1)	1	1	2	2		2
		A-VZ(1)	Shear area (local z-dir) at end(1)	1	1	2	2		3
		J(1)	St. Venant's torsion constant at end(1)	1	1	1	1		1
		IY(1)	Mom. of inertia(loc. y-axis) at end(1)	1	1	1	1		1
		IZ(1)	Mom. of inertia(loc. z-axis) at end(1)	1	1	1	1		1
		A(2)		1	2	3	3		4
		A-VY(2)		1	2	2	4		5
		A-VZ(2)		1	2	2	4		6
		J(2)		1	2	3	3		4
		IY(2)		1	2	3	3		4
		IZ(2)		1	2	3	3		4
SPAR	3	T-WFB	Web thickness	1	1	1	1	1	1
		FAREA1U	Upper flange area at end(1)	1	2	2	2	2	2
		FAREA1L	Lower flange area at end(1)	1	2	2	2	3	3
		FAREA2U	Upper flange area at end(2)	1	2	2	3	4	4
		FAREA2L	Lower flange area at end(2)	1	2	2	3	5	5
		ASTI	Stiffener area	1	1	3	4	!	6
COVER	4	T(0)U	Upper plate thickness	1	1	1	1	1	1
		T(1)U	Smearcd uniaxial gage in S1-dir. upper	1	2	2	2	2	2
		T(2)U	Smearcd uniaxial gage in S2-dir. upper	1	2	3	2	3	3
		T(0)L		1	1	1	3	1	4
		T(1)L		1	2	2	4	4	5
		T(2)L		1	2	3	4	5	6
PLATE	5	T(0)	Plate thickness		1	1			
		TS(1)	Smearcd uniaxial gage in S1-dir.	1	2	2			
		TS(2)	Smearcd uniaxial gage in S2-dir.	1	2	3			
GPLATE	6	T	Average membrane thickness	1					
BRICK	8	none							
SROD	10	A(1)	Cross section area at end(1)	1	1				
		A(2)	Cross section area at end(2)	1	2				
SPLATE	11	T	Plate thickness	1					
CPLATE	12	②	Lamina thicknesses	1					
CCOVER	13	②	Lamina thicknesses of first surface	1	1				
		②	Lamina thicknesses of second surface	1	2				

① An expansion key is the column of integers associated with a particular number of input margins of safety for a particular type of element. Each integer denotes which one of the ordered values defined by an input Record 17 is to be assigned to the corresponding element property shown in the table.

② If one value is input, each lamina of each of the elements will have the same thickness margin-of-safety. If two values are input for CCOVER elements, the first value corresponds to all lamina in the first surface and the second value corresponds to the second surface. Optionally, a sequence of integers may be input to specify a different thickness margin-of-safety for each lamina of the elements.

Record 18 End Margin Subset (optional)

END MARGIN DATA

This record indicates that all margin of safety data for resizing of selected elements have been defined.

112.1.2.4 Element Sizing Data Subset

Material properties and allowable stresses to be used in resizing the structural elements are defined by this data subset (Records 19-21). Material reference codes, previously defined via the MATERIAL Preprocessor (sec. 140.0) and the modulus data subset (Records 6-8), are used to identify elastic properties, allowable stresses and constant buckling stresses for selected elements. The material reference codes defined via the TABLE data (Records 3-5) are used to identify thickness-dependent shear and compression buckling allowable stresses for plate-like elements.

Record 19 Begin Sizing Subset

BEGIN SIZING DATA

Record 20 Element Sizing Data

Mcode <Elid> <Eltype> <Code1 Code2> <factor>

Mcode = Material reference code. M1 through M50 denote Standard materials, whereas M51 through M99 denote Special materials as defined via the MATERIAL Preprocessor. This code defines the ultimate and yield allowable stresses for the specified elements.

The material properties identified by "Mcode" for the elements specified via "Elid" and "Eltype" may be different from the material properties associated with the same elements as defined by the stiffness data. All elements to be resized, however, must have an "Mcode" assigned by this record.

Code1, Code2 = Compression and shear buckling material reference codes of the form BCxx and BSxx, respectively (defined previously by Record 4). Alternatively, only the letter B may be input for one or both

of these items. In this case, certain allowable stresses associated with "Mcode" are to be used as the allowable buckling stresses. The particular "Mcode" stresses used as buckling allowables for the various element types when this option is used are shown in table 112-5.

Default: If "Code1" and "Code2" are not input, certain "Mcode" allowable stresses as indicated in table 112-5 are used as the buckling allowable stress values.

factor = A factor greater than zero. This factor relates the buckling allowable stresses for material axis 2 with the allowable stresses for material axis 1. The allowables defined via "Code1," associated with material axis 1, are multiplied by "factor" to establish the allowables associated with material axis 2. This factor is applicable to all PLATE, GPLATE and COVER elements to be resized (see appendix B).
Default: "factor"=1.0--the buckling allowables associated with both material axes are the same.

This record is repeated to define all element sizing data.

Record 21 End Sizing Subset (optional)

END SIZING DATA

This record indicates that property and allowable stress data have been defined for all elements that are to be resized.

Table 112-5. Correspondence of Element Stresses and Element Allowable Stresses ^①

ELEMENT TYPE		ALLOWABLE STRESS (ref. Sec. 140)								
NAME	NO.	FTU1 or FTY1	FTU2 or FTY2	FTU3 or FTY3	FCU1 or FCY1	FCU2 or FCY2	FCU3 or FCY3	FSU1 or FSY1	FSU2 or FSY2	FSU3 or FSY3
ROD	1	P/A(1) P/A(2)			P/A(1) P/A(2)					
BEAM	2	SIGX			SIGX (M)	SIGX (B)		TAUY (M) TAUZ (M)	TAUY (B) TAUZ (B)	
SPAR	3	SIGF			SIGF (M)	SIGF (B)	SIGST	TAUWEB (M)	TAUWEB (B)	
COVER	4	SIGMA1U SIGMA1L	SIGMA2U SIGMA2L	SIG-S1U SIG-S2U SIG-S1L SIG-S2L	SIGMA1U SIGMA1L	SIGMA2U SIGMA2L	SIG-S1U SIG-S2U SIG-S1L SIG-S2L	TAU12U TAU12L (M)	TAU12U TAU12L (B)	
PLATE	5	SIGMA1	SIGMA2	SIGMAS1 SIGMAS2	SIGMA1	SIGMA2	SIGMAS1 SIGMAS2	TAU12 (M)	TAU12 (B)	
GPLATE	6	SIGMA1	SIGMA2		SIGMA1 (M)	SIGMA1 (B) SIGMA2 (M)	SIGMA2 (B)	TAU12 (M)	TAU12 (B)	
BRICK	8	SIGMA1	SIGMA2	SIGMA3	SIGMA1	SIGMA2	SIGMA3	TAU12	TAU13	TAU23
SP. 4	10	P/A(1) P/A(2)			P/A(1) P/A(2)					
SP. 4	11							J-EQUIV/T (M)	Q-EQUIV/T (B)	

^① Stresses tagged with (B) denote which allowable stress values are used as the allowable buckling stresses when a B is input for Code1 or Code2 in Record 20. Stresses tagged with (M) denote which allowable stress values are used as the allowable buckling stresses when Code1 and Code2 in Record 20 are defaulted. Element stress notation not presented in Appendix B are shown in the following table.

ELEMENT TYPE		ELEMENT STRESS	DESCRIPTION
NAME	NO.		
BEAM	2	SIGX	Axial stress due to axial load and bending moment at either end
		TAUY	Average shear stress in the y-direction due to shear force at either end
		TAUZ	Analogous to TAUY except for the z-direction
SPAR	3	SIGF	Axial stress in SPAR flange
		SIGST	Axial stress in SPAR stiffener (only for M.S. calculations)
		TAUWEB	Shear stress in SPAR web

112.1.2.5 Element Restrain-Sizing Data Subset (optional)

Selected elements within a model which are not to be resized by the DESIGN Processor are defined by this data subset (Records 22-24). The properties of all elements identified by Record 23 remain the same as defined via the stiffness input data or as calculated via a previous execution of the DESIGN Processor. This capability allows resizing of only selected regions of a structural model to be effected. It should be noted that these data constrain elements, whereas "Fixed" data constrain certain property values of selected elements.

Record 22 Begin Restrain-Sizing Subset

```
BEGIN RESTRAIN-SIZING DATA
```

Record 23 Restrain-Sizing Data

```
Elid <Eltype>
```

This record is repeated to identify all elements which are not to be resized.

Record 24 End Restrain-Sizing Data (optional)

```
END RESTRAIN-SIZING DATA
```

This record indicates that all elements which are not to be resized have been identified.

112.1.2.6 Element Optimization Data Subset (optional)

Problem-definition data for regional strength-optimization of a composite structure modeled with CPLATE and CCOVER stiffness elements are defined by this data subset (Records 25-27).

The region of the structural model to be strength-resized is identified as a stiffness element subset previously defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0). All composite elements in this subset must have the same number of lamina.

The number of calculations associated with an optimization problem may be reduced by identifying certain "critical" elements within the region. That is, resizing of all the composite elements within a region may be based on satisfaction of the strength/weight design criteria for selected "strength-critical" elements within the region. These elements are identified by an element subset (a sub-region) of the resize-region subset.

ORIGINAL PAGE IS
OF POOR QUALITY

Each optimization problem is solved by the DESIGN Processor according to the following steps:

- a) The stress states of the first lamina of each of the sub-region elements are compared for each design loadcase. The most "strength-critical" first lamina is identified. The most critical second, third, etc. laminas are identified similarly.
- b) The set of "most critical" laminas is used to satisfy the selected strength criterion.
- c) All other laminas of the elements in the resize-region are modified as necessary. The number of layers associated with the i-th lamina of each element is calculated such that each of the i-th laminas of the elements has a total thickness equal to or greater than the i-th optimized lamina.

Only the number of layers associated with these laminas is allowed to vary.

Record 25 Begin Optimization Subset

BEGIN OPTIMIZATION DATA

The region of the structural model to be resized and the composite elements within this region to be used in the optimization calculations are identified by Record 26. One of the two input variations of this record may be used to define an optimization problem.

Record 26 Optimization Problem Data

Variation 1 **Etype <Exxx>**

Etype = A key-word or its integer-equivalent that denotes a particular type of element. The options are:

<u>Key-Word</u>	<u>Integer-Equivalent</u>
CPLATE	12
CCOVER	13

Exxx = The name of a stiffness element subset which contains the elements of type "Etype" that are to be resized. If Exxx contains other types of elements, they are ignored.

Default: All elements of type "Etype"
included in the SET identified
by Record 9.

Variation 2

Etype Exxx <Eyyy>

Etype, Exxx = Same as defined for Variation 1

Eyyy = The name of a stiffness element subset
which contains some of the "Etype"
elements included in Exxx. Eyyy
identifies the sub-region of elements
which are to be used in performing the
optimization calculations. If Eyyy
contains elements other than "Etype,"
only the "Etype" elements are used.
Default: Exxx

This record, Variations 1 or 2, is repeated to define
all optimization problems.

Record 27 End Optimization Subset

END OPTIMIZATION DATA

This record indicates that all optimization problems for
the stiffness SET identified by Record 9 have been defined.

112.1.2.7 Element Design-Variable Constraint Data Subset
(optional)

This subset, Records 28-30, allows the user to specify
that the same number of layers is to be associated with two
lamina within each of the composite elements included in a
regional optimization problem defined via Record 26.

Use of this capability allows "balanced" laminates (i.e.,
equal lamina thicknesses in the $\pm 45^\circ$ directions) to result from
the resize calculations. Constraints specified by this data
subset also reduce the number of design variables associated
with the optimization problem. Regardless of the constraints,
stresses and strains are calculated for each lamina of the
elements in the resize region.

ORIGINAL PAGE IS
OF POOR QUALITY

Record 28 Begin Variable-Constraint Subset

BEGIN VARIABLE-CONSTRAINT DATA

All constraint data for an optimization problem are defined by the following record.

Record 29 Variable Constraint Data

```
Eltype <Exxx> Con1 Con2 . . .
```

- Eltype = Same as defined for Record 26
- Exxx = The name of a stiffness element subset for which lamina constraints are specified by "Coni." This subset must have been previously identified as a resize region via Exxx in Record 26. Default: All elements of type "Eltype" included in the SET identified by Record 9.
- Coni = The i-th constraint input as
 Num1 = Num2
 where "Num1" and "Num2" are integers and at least one space is required before and after the equal sign. "Numi" is a lamina number as established via the element-definition records of the stiffness data. The constraints may not be concatenated (e.g. 1 = 3, 3 = 5) and the number of lamina layers of one CCOVER surface may not be constrained to the number of lamina layers in the second CCOVER surface.

Example Record: CCOVER E10 1 = 4 8 = 10 /
 All CCOVER elements contained in subset E10 will be resized so that the first and fourth laminas will have the same number of layers and the eighth and tenth laminas will have the same number of layers.

This record is repeated to define constraint data for different optimization problems.

Record 30 End Variable-Constraint Subset

END VARIABLE-CONSTRAINT DATA

This record indicates that all variable-constraint data for the optimization problems identified via Records 25-27 have been input.

112.1.2.8 Element Smoothing Data Subset (optional)

This data subset (Records 31-33) allows the user to specify that the same property values are to be associated with selected elements. This capability allows the properties resulting from the automated resize calculations to be "smoothed" such that practical variations of the element properties are established. If the element properties as calculated via the DESIGN Processor are not to be modified, this data subset should not be input.

Record 31 Begin Smoothing Subset

BEGIN SMOOTHING DATA

The elements of a particular type which are to have the same property values are identified by Record 32.

Record 32 Smoothing Data

Elid <Eltype> $\left\langle \begin{array}{l} Nxxxxx \\ Ixxxxx \\ plist \end{array} \right\rangle$

$\left. \begin{array}{l} Nxxxxx \\ Ixxxxx \end{array} \right\}$ = The letter N or the letter I followed by a 1 to 5 digit user (N) or internal (I) element number. The property values of the elements identified by "Elid" will be made equal to the property values of the element identified by this data item. The type of element identified by this item must be the same type of element identified by "Eltype."

plist = A list of element cross-sectional property values for the element type identified by "Eltype." The property values of the elements identified by "Elid" will be made equal to these property values. The properties and the various input variations thereof for each type of element

are summarized in section 152.0 and are presented in detail in appendix B. This input item is the same as the "plist" item described for the element-definition record of section 152.0.

Default: If only "Elid" or "Elid" and "Eltype" are input, smoothing of the element properties is to be based on the maximum property values associated with the elements identified by "Elid." Each property of these elements is assigned the corresponding maximum value as calculated for the "Elid" elements.

This record is repeated to define all smoothing data.

Record 33 End Smoothing Subset

END SMOOTHING DATA

This record indicates that all smoothing data for the stiffness SET identified by Record 9 have been defined.

112.1.2.9 Element Buckling-Interaction Data Subset (optional)

Parameters may be input by this data subset (Records 34-36) to define panel-buckling interaction formula for "plate-like" elements. The general form of each interaction equation is

$$F = a \cdot R1^{e1} + b \cdot R2^{e2} + c \cdot R12^{e3}$$

where each "F" denotes the ratio of the corresponding applied stress in the 1, 2 or 1-2 material-axis directions to the allowable stress in the same direction. If this data subset is not input, the following interaction equation is used:

$$R = [R1 / (1 - R12^2)]^{1/2} + R2 / (1 - R12^2)$$

Record 34 Begin Buckling-Interaction Subset

BEGIN BUCKLING DATA

The parameters a, b, c, e1, e2 and e3 to be used in the foregoing interaction expression for specific elements are defined by Record 35.

Record 35 Interaction-Expression Parameters

<Elid> <Eltype> parlist

parlist = A list of parameters to be associated with the foregoing general buckling-interaction expression. The input-sequence, number and input variations which may be used for "parlist" are summarized in Table 112-6.

This record is repeated to define all buckling interaction formula for the plate-like elements in the mathematical model.

Record 36 End Buckling-Interaction Subset

END BUCKLING DATA

This record indicates that all buckling-interaction data for the stiffness SET identified by Record 9 have been defined.

Table 112-6. Input Variations of Interaction - Expression Parameters

PARAMETER	INPUT PARAMETER EXPANSION KEYS ①					
	NO. OF INPUT PARAMETER VALUES					
	1	2	3	4	5	6
e1	1	1	1	1	1	1
e2	1	0.	2	0.	2	2
e3	1	2	3	2	3	3
a	1.0	1.0	1.0	3	4	4
b	1.0	0.	1.0	0.	4	5
c	1.0	1.0	1.0	4	5	6

① An expansion key is associated with a particular number of input parameters. Each integer denotes which one of the ordered values defined by an input Record 35 is to be assigned to the corresponding parameter shown in the table.

112.1.3 Stage-Level Design Data

Design load cases to be associated with selected STAGES are established by Records 37-46.

Record 37 BC Stage Identification

STAGE St <n1> <n2> <THn3>

- St = A STAGE number (integer) associated with the stiffness SET "Se" (Record 9). Design load cases as defined subsequently are associated with this SET/STAGE structural model.
- n1 = The maximum number (integer) of design load cases defined via Records 38-40 for the specified STAGE.
Default: n1=2
- n2 = The maximum number (integer) of design load cases defined via superposition (Records 41-43) for the specified STAGE.
Default: n2=0
- THn3 = A code of the form TH followed by the maximum number (integer) of thermal design load cases defined via Records 44-46 for the specified STAGE.
Default: THn3=TH0

Caution: (n1 + n2) must be less than 25.
"n1" must be less than 26 and "n3" must be less than 26.

This record must precede all STAGE-level data subsets input by Records 38-46.

112.1.3.1 Loads Data Subset (optional)

Selected load cases associated with SET "Se" and STAGE "St" are defined as design load cases for selected elements via Record 39.

Record 38 Begin Loads Data Subset

BEGIN LOADS DATA

Record 39 Design Loads Data

CASE Case $\begin{matrix} \langle U \\ Y \rangle \end{matrix}$ <factor> <Tcode> <Elid> <Eltype>

Case = A load case label identifying a set of loads for which the model is to be resized.

$\begin{matrix} \langle U \\ Y \rangle \end{matrix}$

= The letter U or Y denoting whether ultimate or yield material allowables, respectively, are to be used when resizing for the specified load case. Default: Ultimate (U) load case.

factor = The factor by which the loads identified via "Case" are to be multiplied. Default: "factor" = 1.0

Tcode = Temperature code of the form T followed by a signed or unsigned integer in the range -461<temp <1000 (e.g., T100, T+80,T-50) which denotes the temperature in degrees Fahrenheit at which material properties (allowable stresses) are to be extracted from the "table" data. Default: T70

This record is repeated to define all design load cases which are subsets of previously-defined load cases.

Record 40 End Loads Data Subset (optional)

END LOADS DATA

This record indicates that the design load-case data associated with the STAGE identified by Record 37 have been input.

112.1.3.2 Superposition Data Subset (optional)

Design load cases defined as the sum of two factored, design load cases are established via Records 41-43. The previously-defined load cases must have been specified via Record 39 for the SET and STAGE identified by Records 9 and 37, respectively.

Record 41 Begin Superposition Data Subset

```
BEGIN SUPERPOSITION DATA
```

Record 42 Design Load Superposition Data

```
CASE case  $\begin{matrix} U \\ Y \end{matrix}$  <fac1> Case1 <fac2> Case2
<Tcode> <Elid> <Eltype>
```

case = A design load-case label (positive integer or alphanumeric word with 1 to 7 characters). A load case label defined by this item may not be used as "Case1" or "Case2" in subsequent records of this type.

 $\begin{matrix} U \\ Y \end{matrix}$

= The letter U or Y denoting whether ultimate or yield material allowables, respectively, are to be used when resizing for the load case defined by this record.
Default: Ultimate (U) load case.

<fac1>, Case1 } = Loads associated with two previously-
<fac2>, Case2 } defined load cases identified by the labels "Case1" and "Case2" are to be factored and summed. Loads associated with "Case1" are factored by "fac1." The result defines the design load case identified by "case."
Defaults: fac1=1.0 and fac2=1.0

Tcode = Same as defined for Record 39.

This record is repeated to define all design load cases generated by superimposing two previously-defined load cases.

Record 43 End Superposition Data Subset (optional)

```
END SUPERPOSITION DATA
```

This record indicates that the load-case superposition data for the STAGE identified by Record 37 have been input.

112.1.3.3 Thermal Design Loads Data Subset (optional)

Thermal design load cases for SET "Se" and STAGE "St" are established via Records 44-46. A thermal design load case is defined as the sum of a factored mechanical load case plus a factored thermal load case previously defined by the loads data (sec. 134.0). Thermal design load cases for selected elements (see Record 45) are used only for a thermal fully-stressed design.

Record 44 Begin Thermal-Design Load Case Subset

```
BEGIN THERMAL DATA
```

Record 45 Thermal Design Load Data

```
CASE case  $\begin{matrix} \langle U \rangle \\ \langle Y \rangle \end{matrix}$  <fac1> Case1 <fac2> Case2  
<Tcode> <Elid> <Eltype>
```

All items in this record are the same as those defined for Record 42 with the following exceptions:

- a) "Case1" is a mechanical load case
- b) "Case2" is a thermal load case
- c) The default for "Tcode" is the element temperatures as defined by the stiffness data (sec. 152.0).

This record is repeated to define all thermal design load cases by superimposing two previously-defined load cases.

Record 46 End Thermal-Design Load Case Subset (optional)

```
END THERMAL DATA
```

This record indicates that the thermal-design load cases for the STAGE identified by Record 37 have been input.

Record 47 End Data Set

```
END DESIGN DATA
```

Additional design data sets may be input by repeating Records 1-47.

Table 112-7. Summary of Design Data Records

Reference Page	Data Records
112.4 112.4	BEGIN <u>DESIGN</u> DATA MODE Code
112.7 112.7 112.8	BEGIN <u>TABLE</u> DATA Bcode Tcode g1 g2 ... gn s1 s2 ... sn END <u>TABLE</u> DATA
112.9 112.9 112.10	BEGIN <u>MODULUS</u> DATA Mcode Tcode s1 s2 ... sn m1 m2 ... mn END <u>MODULUS</u> DATA
112.10	SET Se
112.10 112.11 112.11	BEGIN <u>PROPERTY</u> DATA <Elid><Etype> plist END <u>PROPERTY</u> DATA
112.11 112.13 112.13	BEGIN $\left. \begin{array}{l} \text{FIXED} \\ \text{LOWER-BOUND} \\ \text{UPPER-BOUND} \end{array} \right\}$ DATA <Elid><Etype> plist END $\left. \begin{array}{l} \text{FIXED} \\ \text{LOWER-BOUND} \\ \text{UPPER-BOUND} \end{array} \right\}$ DATA
112.15 112.15 112.17	BEGIN <u>MARGIN</u> DATA <Elid><Etype> plist END <u>MARGIN</u> DATA
112.17 112.17 112.18	BEGIN <u>SIZING</u> DATA Mcode <Elid><Etype><Code1 Code2><factor> END <u>SIZING</u> DATA
112.20 112.20 112.20	BEGIN <u>RESTRAIN-SIZING</u> DATA Elid <Etype> END <u>RESTRAIN-SIZING</u> DATA
112.21 112.21 112.22	BEGIN <u>OPTIMIZATION</u> DATA Etype <<Exxx> Eyyy> END <u>OPTIMIZATION</u> DATA
112.23 112.23 112.24	BEGIN <u>VARIABLE-CONSTRAINT</u> DATA Etype <Exxx> Con1 Con2 ... END <u>VARIABLE-CONSTRAINT</u> DATA

Table continued on next page

Table 112-7. Summary of Design Data Records (Cont'd.)

Reference Page	Data Records
112.24	BEGIN <u>S</u> MOOTHING DATA
112.24	Elid <Eltype> $\left\{ \begin{array}{l} Nxxxxx \\ Ixxxxx \\ plist \end{array} \right\}$
112.25	END <u>S</u> MOOTHING DATA
112.25	BEGIN <u>B</u> UCKLING DATA
112.26	<Elid><Eltype> parlist
112.26	END <u>B</u> UCKLING DATA
112.27	STAGE St <n1><n2><THn3>
112.27	BEGIN <u>L</u> OADS DATA
112.28	CASE Case $\left\langle \begin{array}{c} U \\ Y \end{array} \right\rangle$ <factor><Tcode><Elid><Eltype>
112.28	END <u>L</u> OADS DATA
112.29	BEGIN <u>S</u> UPERPOSITION DATA
112.29	CASE case $\left\langle \begin{array}{c} U \\ Y \end{array} \right\rangle$ <fac1> Case1 <fac2> Case2 <Tcode><Elid><Eltype>
112.29	END <u>S</u> UPERPOSITION DATA
112.30	BEGIN <u>T</u> HERMAL DATA
112.30	CASE case $\left\langle \begin{array}{c} U \\ Y \end{array} \right\rangle$ <fac1> Case1 <fac2> Case2 <Tcode><Elid><Eltype>
112.30	END <u>T</u> HERMAL DATA
112.30	END <u>D</u> ESIGN DATA

114.0 DETAIL DATA

Detailed geometric descriptions of the stiffness finite elements of a structural model are defined by the Detail data set. These data are not required for generation of element stiffness or stress characteristics but may be used for the following types of analysis:

- a) Generation of equivalent nodal loads for thermally-loaded BEAM elements (see sec. 134.0)
- b) Calculation of local buckling allowables for COVER, PLATE and GPLATE elements as required for design constraints (see sec. 112.0)

The Detail data set is divided into the following data subsets:

- a) Cross-Section -- Finite-element, cross-section dimensions to be used for BEAM thermal load calculations. The basic sectional shape of each loaded element is defined via a library of section "concepts" (configurations).
- b) Spacing -- Location of stiffeners for "plate-like" elements. These data define elemental "free spans" to be used for calculation of local buckling allowables.

The Stiffness data set (sec. 152.0) and the Subset-Definition data set (sec. 156.0), as required, must be input prior to the Detail data set in the input stream.

RECORDED PAGE BLANK NOT FILMED

114.1 INPUT DATA

If the default values of the following input data are acceptable, none of the records described in this section should be input.

Record 1 Data Set Identification

BEGIN DETAIL DATA

This record initiates execution of the **DETAIL** Preprocessor.

Record 2 Data Set Number (optional)

Each set of Detail data is identified by an integer number which identifies the associated stiffness data set (structural model). If multiple **SETS** are used in a job, each **SET** must be assigned a unique number by this record. Input data following a record of this type are attributed to this data set until another Record 2 is encountered.

SET Se

Se = Integer identifying a structural model previously defined as a stiffness set (sec. 152.0).

Default Record: **SET 1**

114.1.1 Cross-Section Data Subset (optional)

This data subset, Records 3-5, is used to define finite-element, cross-sectional dimensions to be used for thermal load calculations. If thermally-loaded **BEAM** elements are not included in the structural model, Records 3-5 should not be input. If this data subset is not used, equivalent nodal loads for any thermally-loaded **BEAM** elements are calculated on the basis of unit, element sectional depths.

Record 3 Cross Section Data Subset

BEGIN CROSSECTION DATA

Each basic, sectional shape of the finite elements is defined, as necessary, via a section concept (configuration). The available concepts are illustrated in table 114-1. Dimensions to be associated with a concept are input via Record 4. One or two different concepts, depending on the basic

ORIGINAL PAGE IS
OF POOR QUALITY

geometries of the element, may be used to describe the sectional shapes at the ends of each BEAM element. The depths of the element, relative to the element y and z reference axes, as used for thermal load calculations are denoted by "dy" and "dz" in table 114-1.

Record 4 Cross-Section Dimensions

<code><Elid> <Eltype> Shape1 <Shape2></code>
--

`Elid` = Identifier of a BEAM element or a subset of BEAM elements. The available options are:

`Exxx`--The name of a stiffness element subset previously defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0).

`Nxxxxx`--The letter N followed by a 1 to 5 digit, user element number (sec. 152.0).

Default: All elements associated with the SET identified by Record 2.

`Eltype` = The key-word BEAM or the integer-equivalent 2.
 Default: All BEAM elements included in "Elid."

`Shapei` = $\left\{ \begin{array}{l} \text{Coni,plisti} \\ \text{NOCON} \end{array} \right\}$ = The i-th sectional shape to be used for the elements defined via "Elid" and "Eltype." "Shape1" defines the sectional shape (concept) at "End1" of each BEAM element, whereas "Shape2" defines the concept at "End2" (see appendix B). If a sectional shape is of no significance, the key-word NOCON may be input for this item. Otherwise, a pair of items "Coni" and "plisti" must be input.

`Coni`--Concept reference code--the letter C followed by a one or two-digit integer (e.g., C5, C26). A concept

identifies a basic sectional shape. See table 114-1 for available concept reference codes.

plisti--List of property values (dimensions) to be associated with "Coni." The input sequence, number and type of properties associated with each "Coni" are summarized in table 114-1.

Default: If "Shape2" is not input, "Shape1" is used at "End2" of the BEAM elements.

This record may be repeated.

Record 5 End Cross Section Subset (optional)

END CROSSSECTION DATA

Additional cross-section data may be defined for different stiffness data sets by repeating Records 2-5.

114.1.2 Spacing Data Subset (optional)

This data subset, Records 6-8, is used to define locations of stiffeners (a grid of line supports) for COVER, PLATE and GELATE finite elements. These "free span" constraints on the elements are used to calculate local buckling allowables to be used during element resizing by the Design Processor (sec. 112.0 and 212.0). If support spacing data are not input, the lengths of the element boundaries are used to calculate local buckling allowables, as required.

Record 6 Spacing Data Subset

BEGIN SPACING DATA

Record 7 Spacing Data

<Elid> <Eltype> plist

Elid = Same as defined for Record 4.

Eltype = A key-word or equivalent integer that denotes a particular type of element included in "Elid." If "Elid" identifies a single element type, "Eltype" must conform. "Elid" and "Eltype" must identify

a single element type. The available options for "Eltype" are as follows:

<u>Key-Word</u>	<u>Integer-Equivalent</u>
COVER	4
PLATE	5
GPLATE	6

Default: The element type identified via "Elid."

rlist = List of property values (distances) defining the spacing of supports for the element type specified via "Elid" and "Eltype." The input sequence, number and type of support spacings associated with each element type are summarized in table 114-2.

This record may be repeated.

Record 8 End Spacing Data Subset (optional)

END SPACING DATA

Additional cross-section and spacing data may be defined for different stiffness data sets by repeating Records 2-8.

Record 9 End Data Set

END DETAIL DATA

Additional detail data sets may be input by repeating Records 1-9.

Table 114-1. Input Variations of Properties for Stiffness-Element Cross-Section Concepts

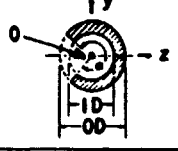
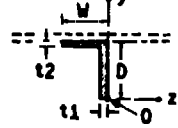
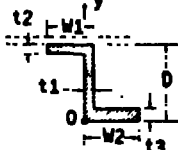
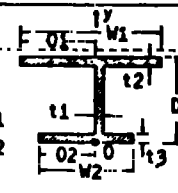
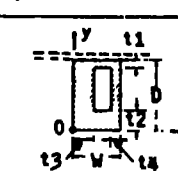
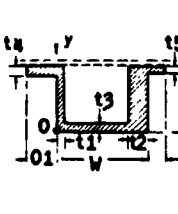
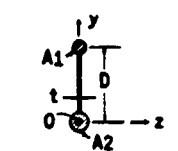
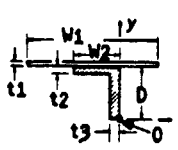
CONCEPT REFERENCE CODE	CONCEPT PROPERTIES (DIMENSIONS)		INPUT PROPERTY EXPANSION KEYS ①																	
			NO. OF INPUT PROPERTIES																	
	DESCRIPTION	LABEL	1	2	3	4	5	6	7	8	9	10	11							
C1	$0 \leq ID < OD$ $0 < \alpha \leq 360^\circ$	 $dy = OD$ $dz = OD$	OD	1	1	1														
		ID	0.	.2	2															
	$0 < t_2 < D$ $0 < t_1 < W$	 $dy = D$ $dz = W$	D		1	1	1													
			W		1	2	2													
	$0 < (t_2 + t_3) < D$ $0 < t_1 < W_1$ $t_1 < W_2$	 $dy = D$ $dz = W_1 + W_2 - t_1$	D		1	1	1	1	1											
			W1		1	2	2	2	2											
	$0 < (O_1 + t_1) \leq W_1$ $0 < (O_2 + t_1) \leq W_2$ $0 < t_1; 0 < t_2$ $0 \leq t_3; 0 < (t_2 + t_3) < D$	 $dy = D$ $dz = \max(W_1, W_2)$	D		1	1	1	1	1	1	1	1								
			W1		1	2	2	2	2	2	2	2	2	2						
	$0 < (t_1 + t_2) \leq D$ $0 < (t_3 + t_4) \leq W$	 $dy = D$ $dz = W$	D	1	1	1	1	1	1											
			W	1	2	2	2	2	2	2	2									
	$0 < (t_1 + t_2) < W$ $0 < (t_3 + t_5) < D$ $0 < (t_3 + t_4) < D$ $0 < t_1; 0 < t_2$ $0 < t_3$	 $dy = D$ $dz = W + O_1 + O_2$	D		1	1	1	1	1	1	1	1	1							
			W		1	2	2	2	2	2	2	2	2	2	2					
	$0 < t$	 $dy = D$ $dz = t$	D			1	1													
			t		1	2	2													
	$0 < t_3 < W_2$ $0 < t_2 < D$ $W_2 \leq W_1$	 $dy = D + t_1$ $dz = W_2$	W1				1	1	1	1										
			D				2	2	2	2	2									
			W2				2	2	2	2										
			t1				3	3	3	3	3									
			t2				3	3	3	3										
			t3				3	3	3	3	3									

Table continued on next page.

Table 114-1. Input Variations of Properties for Stiffness-Element Cross-Section Concepts (Cont'd)

CONCEPT REFERENCE CODE	CONCEPT PROPERTIES (DIMENSIONS)		INPUT PROPERTY EXPANSION KEYS ①											
			NO. OF INPUT PROPERTIES											
	DESCRIPTION	LABEL	1	2	3	4	5	6	7	8	9	10	11	
C23	<p> $dy = D + t1$ $dz = W2 + W3 - t3$ </p> <p> $0. < (t2 + t4) < D$ $0. < t3 < W2$ $0. < t3 < W3$ $W2 \leq W1$ </p>	<p> $dy = D + t1$ $dz = W2 + W3 - t3$ </p>	W1			1	1	1	1	1	1			
			D			2	2	2	2	2	2			
			W2			2	3	3	3	3	3			
			W3			2	3	3	4	4	4			
			t1			3	4	4	5	5	5			
			t2			3	4	5	6	6	6			
			t3			3	4	5	6	7	7			
			t4			3	4	5	6	6	8			
C24	<p> $dy = D + t1$ $dz = \max(W2, W3)$ </p> <p> $0. < (t2 + t4) < D$ $0. < (O1 + t3) \leq W2$ $0. < (O2 + t3) \leq W3$ $0. < t1; 0. < t2$ $0. < t3; 0. \leq t4$ $W2 \leq W1$ </p>	<p> $dy = D + t1$ $dz = \max(W2, W3)$ </p>	W1			1	1	1	1	1	1	1	1	
			D			2	2	2	2	2	2	2	2	
			W2			2	2	3	3	3	3	3	3	3
			W3			2	2	3	3	3	4	4	4	4
			O1			***	***	***	***	***	***	***	5	5
			O2			***	***	***	***	***	***	***	6	6
			t1			3	3	4	4	4	5	6	7	8
			t2			3	4	5	5	5	6	7	8	9
			t3			3	4	5	6	6	7	8	9	10
			t4			3	4	5	5	7	8	9	10	
C26	<p> $dy = D + t1$ $dz = W2$ </p> <p> $0. < (t2 + t3) \leq D$ $0. < (t4 + t5) \leq W2$ $0. < t1; W2 \leq W1$ </p>	<p> $dy = D + t1$ $dz = W2$ </p>	W1			1	1	1	1	1	1			
			D			2	2	2	2	2	2			
			W2			2	3	3	3	3	3			
			t1			3	4	4	4	4	4			
			t2			D/2	D/2	5	5	5	5	5	5	
			t3			D/2	D/2	5	5	6	6	6		
			t4			W/2	W/2	5	6	7	7			
			t5			W/2	W/2	5	6	7	8			
C27	<p> $dy = D + t1$ $dz = W2 + O1 + O2$ </p> <p> $0. < (t4 + t6) < D$ $0. < (t2 + t3) < W2$ $0. < t2; 0. < t3$ $0. < t4$ $(O1 + O2 + W2) \leq W1$ </p>	<p> $dy = D + t1$ $dz = W2 + O1 + O2$ </p>	W1			1	1	1	1	1	1	1	1	
			D			2	2	2	2	2	2	2	2	2
			W2			2	2	3	3	3	3	3	3	3
			O1			2	2	3	4	4	4	4	4	4
			O2			2	2	3	4	4	4	5	5	5
			t1			3	3	4	5	5	5	6	6	6
			t2			3	4	5	6	6	6	7	7	7
			t3			3	4	5	6	6	6	7	8	8
			t4			3	4	5	6	7	7	8	9	9
			t5			3	4	5	6	6	8	9	10	10
t6			3	4	5	6	6	8	9	10	11			

① An expansion key is the column of integers associated with a particular number of input property values for a particular cross section concept. A nonzero integer component of a key denotes which one of the ordered property values defined by an input data record (Record 4) is to be assigned to the corresponding concept property shown in the table.

- Default value is $(W1 - t1)/2$
- ** Default value is $(W2 - t3)/2$
- *** Default value is $(W3 - t3)/2$

The element depths "dy and dz" shown above are used for thermal load calculations.

Table 114-2. Input Variations of Element Support-Spacing Properties

ELEMENT TYPE		SPACING PROPERTIES		INPUT PROPERTY EXPANSION KEYS ①			
NAME	NO.	DESCRIPTION	LABEL	1	2	3	4
COVER	4	<p>Upper Plate d1 measured along S1 d2 measured along S2 in plane of plate</p> <p>Lower Plate d3 measured along S1 d4 measured along S2 in plane of plate</p>	d1	1	1	1	1
			d2	1	1	2	2
			d3	1	2	3	3
			d4	1	2	2	4
PLATE	5	Analogous to half a COVER	d1	1	1		
	d2		1	2			
GPLATE	6	Analogous to PLATE	d1	1	1		
			d2	1	2		

① An expansion key is the column of integers associated with a particular number of input property values for a particular type of element. A non-zero component of a key denotes which one of the ordered property values defined by an input data record (Record 7) is to be assigned to the corresponding element property shown in the table.

Table 114-3. Summary of DETAIL Data Records

Reference Page	Data Records
114.2	BEGIN <u>DETAIL</u> DATA
114.2	SET Se
114.2	BEGIN <u>CROSSECTION</u> DATA
114.3	<Elid><Eltype> Shape1 <Shape2>
114.4	END <u>CROSSECTION</u> DATA
114.4	BEGIN <u>SPACING</u> DATA
114.4	<Elid><Eltype> plist
114.5	END <u>SPACING</u> DATA
114.5	END <u>DETAIL</u> DATA

116.0 DUBLAT AERODYNAMICS DATA

Unsteady aerodynamic analyses of rigid or elastic, nonplanar lifting surfaces and body surfaces defining a general three-dimensional configuration in subsonic flow can be performed by the DUBLAT Processor (sec. 216.0). Calculations performed by this processor are based on the Doublet-Lattice lifting-surface theory published in references 116-1 and 116-2. The input data set supporting Doublet-Lattice aerodynamic analysis is comprised of the following data subsets:

- a) Geometry -- Aerodynamic lifting-surface panel, body-surface panel and body-doublet data which define the aerodynamic model.
- b) Subset -- Aerodynamic strip and box subset definitions used to define Modal and Option input data.
- c) Modal -- Vibration modes or "AIC" mode substitutes (see sec. 232.0) associated with the aerodynamic model.
- d) Option -- Velocity profile and aerodynamic pressure correction data.

The only restriction on the input sequence of these data subsets is that the Geometry subset must be first. If only the quasi-inverse of the normal wash matrix is to be calculated, only the Geometry subset is required. This data set need not be preceded by any other data set within a data deck.

Each set of aerodynamics data is referred to as a CASE. A maximum of 36 DUBLAT data cases may be identified per job. Printout of the input case data may be requested from the DUBLAT Postprocessor (sec. 216.1).

REPRODUCTION FROM BLANK NOT FILMED

116.1 INPUT DATA

Record 1 Data Set Identification

BEGIN DUBLAT DATA

This record initiates execution of the DUBLAT Preprocessor.

Record 2 Data Case Number (optional)

Each set of DUBLAT aerodynamics data is referenced as a case which is identified by an integer number. If multiple sets are used in a job, each case must be assigned a unique number.

CASE n

n = Data case number (integer) in the range 1 to 36, inclusive.

Default Record: CASE 1

116.1.1 Geometry Data Subset

This data subset includes the following geometrical definitions:

- a) Lifting-surface panels with spanwise/streamwise grid lines thereon.
- b) Interference (body) surface panels with spanwise/streamwise grid lines thereon.
- c) Body radii and body-axis, pressure doublet distributions.

All geometry data are defined relative to the GLOBAL reference frame whose positive X-axis is in the aft (streamwise) direction. Configuration symmetry and flow symmetry about the X-Y plane can be accommodated, permitting geometrical data to be input for only the positive-Y half of the model. Only half of the total aerodynamic forces on a vertical tail are calculated by the DUBLAT Processor when X-Z is a plane of symmetry. Thus, the generalized airforces calculated for a complete aircraft model via a symmetrical analysis are different from those calculated via an antisymmetrical analysis (sec. 216.2).

An aerodynamic model may be comprised of any number of spatially-oriented lifting surfaces and bodies to idealize a configuration. The method is capable of analyzing lifting

surfaces with arbitrary planform and dihedral with or without full- or partial-span control surfaces. Intersecting and/or interfering non-coplanar configurations such as T-tail, V-tail and wing-tail combinations may also be analyzed. Lifting-surface/body interference in unsteady flow is handled by placing interference panels on all body surfaces in addition to doublet distributions along the body axes. The doublet distributions account for body incidence and body-diameter effects.

Lifting surfaces and interference body surfaces are idealized by aerodynamic panels. Panels on both types of surfaces are divided into streamwise strips which also have chordwise divisions to define an aerodynamic grid of elements. Each basic trapezoidal element of the grid is referenced as a box. The analysis technique is based on placement of the bound portion of a horseshoe vortex along the quarter-chord line of each box (uniform distribution of acceleration potential doublets along the base quarter-chord) with the control point located at the mid-point of the three-quarter chord line.

The analyst should adhere to the following rules when defining the two types of lifting-surface panel box-grids:

- a) Box boundaries (panel edges) should coincide with surface intersections, surface edges, fold lines and control surface hinge lines.
- b) Panel divisions should be continuous across the boundaries between adjacent panels. See figure 116-1 for an illustration of a DUBLAT aerodynamic model.
- c) If a strip boundary on an upstream surface lies in the plane of a downstream surface, it must coincide with a strip boundary on the downstream surface.
- d) Strips should be concentrated in regions where rapidly varying spanwise loads occur such as near streamwise wing tips and control-surface edges.
- e) Boxes should be concentrated in regions where rapidly varying chordwise loads occur such as near control surface hinge lines.
- f) The aspect ratio of each box (the spanwise box length divided by its mean aerodynamic chord) should be in the range 0.4 to 2.2, inclusive. The ratio of the box length to the basic oscillatory wave length should also be less than or equal to 0.04. Thus, a fine grid of boxes is required for analysis at very high reduced frequencies.

ORIGINAL PAGE IS
OF POOR QUALITY

General oscillatory motion of a closed body with a circular cross section and with different radii along its axis can be analyzed by the slender body aerodynamic theory. However, when multiple bodies and lifting surfaces are introduced into the flow field, interference effects must be considered. These effects are accommodated by placement of interference panels directly on the surface of body segments which are in the interference region(s), (see fig. 116-1). An aerodynamic model of a body is therefore comprised of either the first or both of the following components:

- a) A distribution of pressure doublets along the body axis for isolated slender-body analysis.
- b) Interference panels on the body surface to account for interaction with other surfaces/bodies.

Both components are required, for example, for analysis of closed-body and airfoil-surface interferences and for analysis of an open (flow through) body with an internal closed body (e.g., a cowling and engine). An open body, however, must be idealized by lifting surface panels (not interference panels) placed on its surface (see ref. 116-2). Non-circular slender bodies may be idealized by multiple circular bodies placed in close proximity (see fig. 116-1).

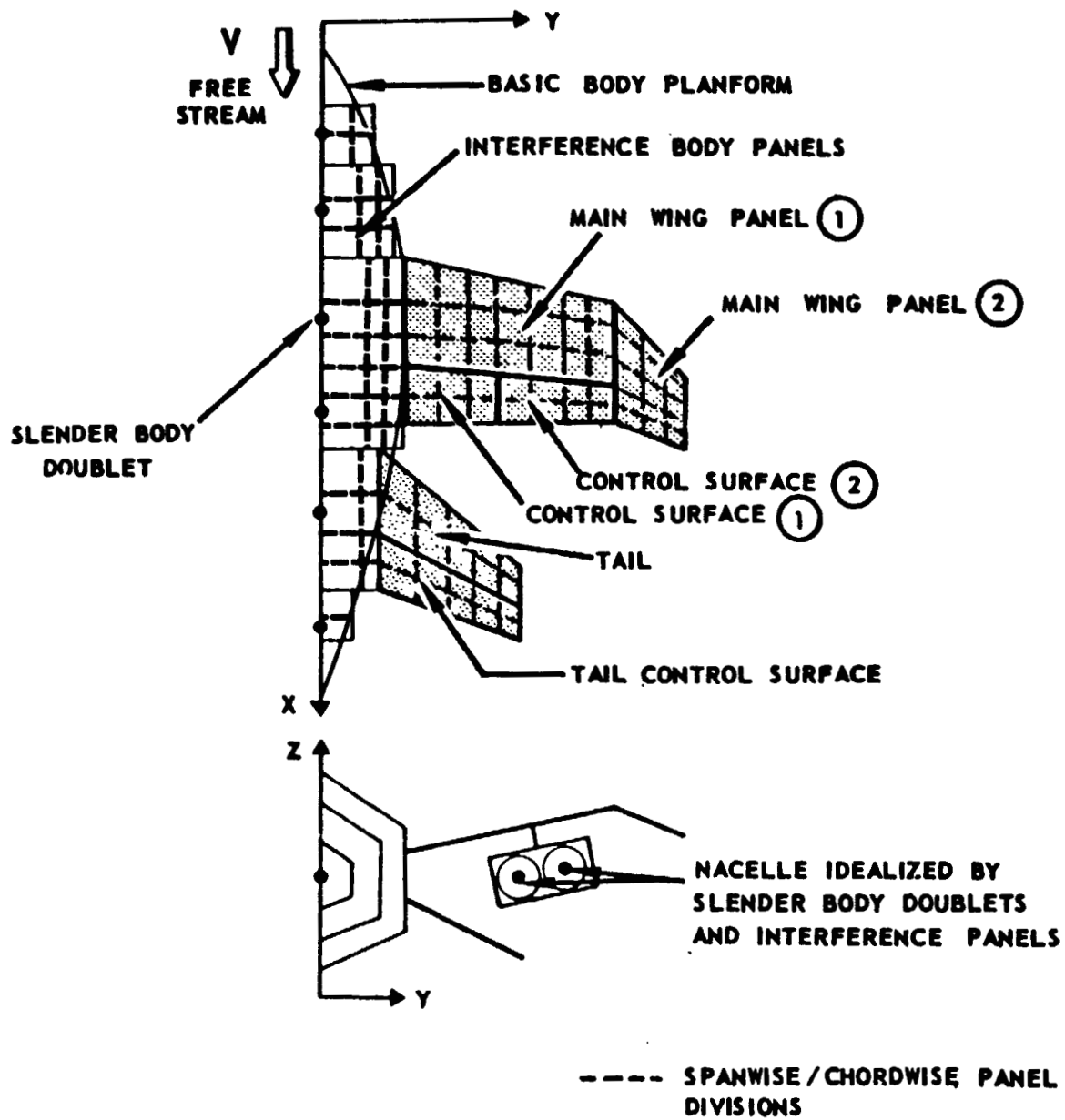


Figure 116-1. Sample DUBLAT Aerodynamic Model

Record 3 Begin Geometry Subset

BEGIN GEOMETRY DATA

Record 4 Lifting Surface Panel Data

LIFTING SURFACE DATA

Records 5-7 define the geometry of a lifting surface panel and the aerodynamic box grid thereon. These records are repeated to define all lifting surfaces.

Record 5 Panel Identification and Geometry

PANEL ident x1 x2 x3 x4 yi yo zi zo

- ident = Positive integer or alphanumeric word of 1 to 5 characters which identifies the panel. The identification of each lifting surface panel associated with a particular data case must be unique.
- x1,x2 = GLOBAL X-coordinates of two points defining the forward and aft locations, respectively, of the inboard edge of the panel ($x1 < x2$).
- x3,x4 = GLOBAL X-coordinates of two points defining the forward and aft locations, respectively, of the outboard edge of the panel ($x3 < x4$).
- yi,yo = GLOBAL Y-coordinates defining the inboard and outboard panel edge locations, respectively.
- zi,zo = GLOBAL Z-coordinates defining the inboard and outboard panel edge locations, respectively.

Each panel is defined by four points such that the inboard and outboard edges of each panel are parallel to the free stream flow. The dihedral of each planar panel is calculated automatically from the specified geometry. A panel in the GLOBAL positive-Y half-space has a positive dihedral if it is located relative to the X-Y plane by a positive right-hand rotation about the X-axis. The normal to a panel is in the direction of positive dihedral measurement.

Record 6 Chordwise Panel Divisions

CHORD DIV c1 c2 ...

ci = Fraction of panel chord in the range $0.0 \leq ci \leq 1.0$. The list of two or more "ci" values must be input in a strictly increasing order beginning with $c1=0.0$ and ending with $cn=1.0$.

Record 7 Spanwise Panel Divisions

SPAN DIV s1 s2 ...

si = Fraction of panel span in the range $0.0 \leq si \leq 1.0$. The list of two or more "si" values must be input in a strictly increasing order beginning with $si=0.0$ and ending with $sn=1.0$. This record defines one or more strips on the panel identified by Record 5.

Records 5-7 are repeated to define all lifting surfaces associated with the data case specified by Record 2.

Records 8-12 define interference body surfaces.

Record 8 Interference Surface Panel Data

INTERFERENCE SURFACE DATA

Record 9 Body Identification

BODY ident

ident = Positive integer or alphanumeric word of 1 to 5 characters which identifies all or part of an aerodynamic body idealized by interference panels (defined by subsequent sets of Records 10-12). The identification of each body associated with a particular data case must be unique. When doublet data are associated with a body, the "Ident" specified by Record 14 must be the same as the "ident" defined by this record.

Records 10-12 are repeated to define the interference panels associated with the body identified by Record 9. At least two panels are required to define a body. Over the length

of each panel, the body is implicitly defined to have a constant axial cross section. See figure 116-1 for an illustration of an aerodynamic model.

Record 10 Panel Identification and Geometry

PANEL ident x1 x2 x3 x4 yi yo zi zo

See Record 5 for a description of the input items. The identification of each interference panel associated with a particular data case must be unique.

Record 11 Chordwise Panel Divisions

CHORD DIV c1 c2 ...

See Record 6 for a description of the input items.

Record 12 Spanwise Panel Divisions

SPAN DIV s1 s2 ...

See Record 7 for a description of the input items.

Records 9-12 are repeated to define all interference bodies associated with the data case specified by Record 2. Following each Record 9 which identifies a particular body, Records 10-12 are repeated to define all interference panels associated with that body.

Records 13-16 are input only if bodies are to be analyzed by the slender body aerodynamic theory or if the aerodynamic model includes interference bodies. If Records 8-12 are input, Records 13-16 are required.

Record 13 Body Doublet Data

DOUBLET DATA

Record 14 Body Identification

RCDY Ident <y z> $\left\langle \begin{matrix} YDOUBLET \\ ZDOUBLET \end{matrix} \right\rangle$

Ident = Positive integer or alphanumeric word of 1 to 5 characters which identifies a complete aerodynamic body doublet or a segment thereof that is idealized by the axial pressure doublets defined by Records 15 and 16. The identification

of each body doublet associated with a particular data case must be unique. Doublet data are associated with an interference body when "Ident" is the same as an "ident" previously-defined by a Record 9.

y,z = GLOBAL Y and Z-coordinates defining the free flow streamline which coincides with the circular slender-body axis. Default Items: $Y=0.0$ and $Z=0.0$ --the body axis coincides with the X-axis.

\langle YDOUBLET
ZDOUBLET \rangle = Key-word denoting that the body is to assume motion in the Y-axis direction (YDOUBLET) or in the Z-axis direction (ZDOUBLET). The key-word defines what type of body structural mode shapes specified by the Modal Data Subset (sec. 116.1.3) are included in the analysis. Default: ZDOUBLET

Record 15 Body Axis Divisions

AXIS DIV x_1 x_2 ... x_n

x_i = The GLOBAL X-coordinate defining the i -th location along the body axis at which the body radius is specified by Record 16. The list of two or more X-coordinates must be input in a strictly increasing order. The length of the body is defined by the difference ($x_n - x_1$).

Record 16 Body Radii

RADII r_1 r_2 ... r_n

r_i = The body radius at the i -th body-axis location specified by Record 15. The number and sequence of items must correspond with the number and sequence of body axis divisions listed in Record 15.

Records 14-16 are repeated to define axial pressure doublet distributions for all aerodynamic bodies for the data case identified by Record 2.

Record 17 End Geometry Subset

END GEOMETRY DATA

This record indicates all geometry data have been defined for the case identified by Record 2.

116.1.2 Lifting-Surface Region Subsets (optional)

This data subset is required if any of the following characteristics is essential to a data case:

- a) Vibration modes calculated by the VIBRATION Processor (see sec. 116.1.3) are to be associated with selected boxes on the lifting surfaces defined by Records 4-7.
- b) Velocity profile modifications associated with lifting surface strips are defined by Records 26-28.
- c) Aerodynamic pressure correction data associated with lifting surface boxes are defined by Records 29 and 30.

Subsets of the strips and/or boxes associated with the aerodynamic panels previously-defined are specified by this data subset. Each panel strip and each box is identified by a unique integer number. Both number sequences begin with unity. Sequential numbers are automatically assigned to panel strips beginning with the most inboard strip to the most outboard strip of each panel. Panels are ordered according to the sequence in which they are defined by Record 5. Boxes are automatically assigned sequential numbers beginning with the most forward box to the aft-most box in each strip in the order of monotonically increasing strip number.

Record 18 Begin Subset-Data Subset

BEGIN SUBSET DATA

Record 19 Identification of Subset Type

SUBSETS OF { STRIPS
BOXES }

{ STRIPS
BOXES }

= Key-word denoting that subsequent Records 20 define subsets of STRIPS or subsets of BOXES.

Record 20 Subset Definition Record

SUBSET ident ATlist

ident = Positive integer or alphanumeric word of 1 to 5 characters which identifies a subset. The identifiers of all STRIP subsets must be unique and the identifiers of all BOX subsets must be unique for a particular data case.

ATlist = ATLAS list of STRIP numbers or BOX numbers (positive integers) which are to be included in subset "ident." Non-existent STRIP numbers or BOX numbers generated by ATlist are not allowed.

All strips included in a subset must be defined by two distinct streamlines. That is, more than one strip may be included in a subset only if the strips are associated with different coplanar panels. One panel must be aft of the other. Each panel must have spanwise divisions defined by Record 7 so that the inboard edge and the outboard edge of each strip included in the subset are on two common streamlines, respectively.

All subsets (STRIPS or BOXES) are defined by repeating Records 19 and 20. A subset "ident" defined by Record 20 may not be the same as the subset "ident" specified by a subsequent Record 20.

Record 21 End Subset-Data Subset

END SUBSET DATA

This record indicates all lifting-surface-region subsets have been defined.

116.1.3 Modal Data Subset (optional)

This subset defines the vibration modes, "AIC" mode substitutes or rigid-body modes to be associated with each component of the aerodynamic model (lifting surfaces, interference surfaces and body doublets). If no modal data are input, only the quasi-inverse matrices would be generated by execution of the DUBLAT Processor.

The unsteady aerodynamic downwash distribution is approximated by a linear combination of modal displacement amplitudes at the lifting-surface control points and body doublet locations. The aerodynamic theory requires translational components of the vibration modes to be normal to the aerodynamic surfaces. Thus, the analysis frame axis associated with the retained nodal freedoms must be normal to the aerodynamic surface. The analysis frame associated with interference bodies, however, must be the GLOBAL triad. In general, both Y and Z nodal translational freedoms must be retained when calculating vibration modal components for interference panels. The modal component normal to each body interference panel is automatically extracted by the DUBLAT Processor. Modes generated via the POLYNOMIAL option of the INTERPOLATION Processor (sec. 232.1) are evaluated in GLOBAL X and Y or X and Z coordinates.

Record 22 Begin Modal Subset

BEGIN MODAL DATA

Record 23 defines the vibration modes or "AIC" mode substitutes that are to be used in the analysis of the case identified by Record 2. Only one of the two variations of this record may be used per data case. Variation 1 must be used when mode shape coefficients generated by the INTERPOLATION Processor are to be employed in the aerodynamic analysis. Modes may be defined directly via Variation 2 for a rigid-body aerodynamic analysis.

Record 23 Modal Data

<u>Variation 1</u>	USE Name WITH	<table style="border: none;"> <tr> <td style="border: none;">{</td> <td style="border: none;">LIFTING SURFACE</td> <td style="border: none;">Bsubsets</td> <td style="border: none;">}</td> </tr> <tr> <td style="border: none;">{</td> <td style="border: none;">INTERF BODY</td> <td style="border: none;">Bodies</td> <td style="border: none;">}</td> </tr> <tr> <td style="border: none;">{</td> <td style="border: none;">BODY DOUBLET</td> <td style="border: none;">Bodies</td> <td style="border: none;">}</td> </tr> </table>	{	LIFTING SURFACE	Bsubsets	}	{	INTERF BODY	Bodies	}	{	BODY DOUBLET	Bodies	}
{	LIFTING SURFACE	Bsubsets	}											
{	INTERF BODY	Bodies	}											
{	BODY DOUBLET	Bodies	}											

Name = The name of an interpolation-coefficient matrix previously generated via the INTERPOLATION Processor (see sec. 232.0).

<table style="border: none;"> <tr> <td style="border: none;">LIFTING SURFACE</td> <td style="border: none;">}</td> <td rowspan="3" style="border: none; vertical-align: middle;"> = Key-word denoting whether "Name" is to be associated with LIFTING SURFACES, interference bodies (INTERF BODY) or BODY DOUBLETs. A particular coefficient matrix may be used with one or more of the three types of aerodynamic model components. It should be noted that each matrix is associated with a subset of nodes all of which have the same analysis frame. A GLOBAL analysis frame must be </td> </tr> <tr> <td style="border: none;">INTERF BODY</td> <td style="border: none;">}</td> </tr> <tr> <td style="border: none;">BODY DOUBLET</td> <td style="border: none;">}</td> </tr> </table>	LIFTING SURFACE	}	= Key-word denoting whether "Name" is to be associated with LIFTING SURFACES, interference bodies (INTERF BODY) or BODY DOUBLETs. A particular coefficient matrix may be used with one or more of the three types of aerodynamic model components. It should be noted that each matrix is associated with a subset of nodes all of which have the same analysis frame. A GLOBAL analysis frame must be	INTERF BODY	}	BODY DOUBLET	}
LIFTING SURFACE	}	= Key-word denoting whether "Name" is to be associated with LIFTING SURFACES, interference bodies (INTERF BODY) or BODY DOUBLETs. A particular coefficient matrix may be used with one or more of the three types of aerodynamic model components. It should be noted that each matrix is associated with a subset of nodes all of which have the same analysis frame. A GLOBAL analysis frame must be					
INTERF BODY	}						
BODY DOUBLET	}						

associated with the coefficient matrix used for interference-body panels.

- Bsubsets** = A list of box subset identifiers defined by Record 20. The vibration modes to be used with all lifting surface boxes included in the listed subsets are identified by the matrix "Name." Geometry compatibility of the corresponding boxes and the region associated with the nodes used to define the matrix "Name" is a user responsibility. A coefficient matrix must be associated with each lifting surface box that is to undergo motion.
- Bodies** = A list of body identifiers defined by Record 9 or Record 14. The vibration modes to be used with the listed bodies are identified by the matrix "Name." Geometry compatibility of the specified bodies and the region associated with the nodes used to define the matrix "Name" is a user responsibility. A coefficient matrix must be associated with each body that is to undergo motion.

This record must be repeated to associate an interpolation coefficient matrix with all lifting surface boxes and bodies for the aerodynamic model.

Variation 2 allows the user to define 1 to 6 rigid body modal freedoms relative to the GLOBAL reference frame.

Variation 2 RIGID BODY <x y z> Rbf1 mag1 Rbf2 mag2 ...

x,y,z = GLOBAL X, Y and Z coordinates defining the reference point to be used in generating the rigid body modes.
Default: The origin of the GLOBAL frame is used as the reference point.

Rbfi = Key-word denoting the i-th rigid body freedom selected from the list TX, TY, TZ, RX, RY and RZ. Within each of these key-words, the letter "T" is associated with a translation freedom and "R" is associated with a rotation freedom relative to the GLOBAL X-Y-Z triad.

magi = The relative magnitude of the i-th rigid body freedom with respect to the reference point. Items "Rbfi" and "magi" are input in pairs 1 to 6 times in an order which defines the sequence of rigid body modes used in the aerodynamic analysis.

Record 24 End Modal Subset

END MODAL DATA

This record indicates all modal data have been defined for the case identified by Record 2.

116.1.4 Option Data Subset (optional)

This data subset contains the following types of information:

- a) Velocity profile modifications to account for lifting-surface thickness aerodynamic effects on selected strip subsets.
- b) Aerodynamic pressure-difference scaling and/or replacement values associated with selected box subsets.

If neither of these analysis refinement options is desired, Records 25-31 should not be input.

Record 25 Begin Option Subset

BEGIN OPTION DATA

Records 26-28 and Records 29-30 are input in groups, only as required, to modify calculated velocity profiles or to scale and/or replace calculated pressure differences, respectively.

Record 26 Velocity Profile Data Identification

VELOCITY PROFILES

Velocity profile modification data are defined and labeled by Record 27, whereas Record 28 defines which profile modification data are to be associated with selected lifting surface STRIP subsets.

Record 27 Velocity Profile Data

A velocity profile is defined as the variation of V_{LOCAL}/V along the local chord where V_{LOCAL} is the local air-flow velocity and V is the free-stream velocity. The first or second derivative of the profile at the forward and/or aft edges of an aerodynamic strip may optionally be specified in addition to velocity profile values at selected points on the aerodynamic strip. The velocity derivatives are calculated relative to the distance along the local chord measured in terms of fraction of the local chord.

PROFILE ident	$\langle DLE1, dle \rangle$	$\langle DTE1, dte \rangle$
	$\langle DLE2, dle \rangle$	$\langle DTE2, dte \rangle$
x1 v1 x2 v2 ... xn vn		

ident = Positive integer or alphanumeric word of 1 to 5 characters which identifies a set of velocity profile data. The identification of each set of profile data associated with a particular data case must be unique.

$\langle DLE1, dle \rangle$
 $\langle DLE2, dle \rangle$ = Key-word DLE1 or DLE2 denoting whether the first or the second velocity-profile derivative, respectively, at the leading (forward) edge of a strip is to assume the value specified by "dle." Only one of the derivatives at the leading edge may be specified.
Default: Derivatives not specified are calculated automatically.

$\langle DTE1, dte \rangle$
 $\langle DTE2, dte \rangle$ = Key-word DTE1 or DTE2 denoting whether the first or the second velocity-profile derivative, respectively, at the trailing (aft) edge of a strip is to assume the value specified by "dte." Only one of the derivatives at the trailing edge may be specified.
Default: Derivatives not specified are calculated automatically.

xi = The fraction of strip chord, $0.0 \leq xi \leq 1.0$, at which "vi" is defined.

vi = Value of the velocity profile (V_{LOCAL}/V) at "xi." Items "xi" and "vi" are input in pairs such that the "xi" values are

strictly increasing. At least four and not more than 26 distinct values of "xi" must be used.

This record is repeated to define all required velocity profile modification data.

Record 23 Velocity Profile Modifications

USE Ident ON Subsets

- Ident = A velocity profile data label defined by Record 27.
- Subsets = A list of STRIP subset identifiers defined by Record 20. The velocity profile modifications identified by "Ident" are applied to each lifting surface strip included in the specified subsets.

This record may be repeated. The user is cautioned that only one profile modification may be specified for a particular strip. Additionally, if more than one strip is included in a subset, the fractions of strip chord specified by Record 27 are based on the combined chord lengths of the included strips.

Record 29 Pressure Correction Data Identification

PRESSURE CORRECTIONS

Record 30 Pressure Data Modifications

USE f1 f2 AS { SCALAR } ON Bsubsets
PRESSURE

- f1, f2 = The real and imaginary components, respectively, of a complex number. This number is either a scale factor (SCALAR) to be applied to the calculated non-dimensional pressure difference or it is the value of the non-dimensional pressure difference (PRESSURE) to be used in the analysis.
- Bsubsets = A list of box subset identifiers defined by Record 20. The specified complex number (f1, f2) will either scale (SCALAR)

DUBLAT

or replace (PRESSURE) the pressure difference value calculated for each box included in the specified subsets.

This record may be repeated. Only one pressure scaling or replacement value may be specified per bcx.

Record 31 End Option Subset

END OPTION DATA

Additional cases may be defined by repeating Records 2-31 with a unique identification number assigned to each case by Record 2.

Record 32 End DUBLAT Data Set

END DUBLAT DATA

Additional data cases may be input by repeating Records 1-32.

Table 116-1. Summary of DUBLAT Data Records

Reference Page	Data Records
116.2	BEGIN DUBLAT DATA
116.2	CASE n
116.6	BEGIN GEOMETRY DATA
116.6	LIFTING SURFACE DATA
116.6	PANEL ident x1 x2 x3 x4 y1 yo zi zo
116.7	CHORD DIV c1 c2 ...
116.7	SPAN DIV s1 s2 ...
116.7	INTERFERENCE SURFACE DATA
116.7	BODY ident
116.8	PANEL ident x1 x2 x3 x4 y1 yo zi zo
116.8	CHORD DIV c1 c2 ...
116.8	SPAN DIV s1 s2 ...
116.8	DOUBLET DATA
116.8	BODY Ident <y z> $\left\langle \begin{array}{l} YDOUBLET \\ ZDOUBLET \end{array} \right\rangle$
116.9	AXIS DIV x1 x2 ...
116.9	RADII r1 r2 ...
116.10	END GEOMETRY DATA
116.10	BEGIN SUBSET DATA
116.10	SUBSETS OF $\left\{ \begin{array}{l} STRIPS \\ BOXES \end{array} \right\}$
116.11	SUBSET ident ATlist
116.11	END SUBSET DATA

Table continued on next page

Table 116-1. Summary of DUBLAT Data Records (Cont'd.)

Reference Page	Data Records
116.12	BEGIN MODAL DATA
116.12	USE Name WITH { LIFTING SURFACE Bsubsets } { INTERF BODY Bodies } { BODY DOUBLET Bodies }
116.13	RIGID BODY < x y z > Rbf1 mag1 Rbf2 mag2 ...
116.14	END MODAL DATA
116.14	BEGIN OPTION DATA
116.14	VELOCITY PROFILES
116.15	PROFILE ident <DLE1 dle> <DTE1 dte> x1 v1... <DLE2 dle> <DTE2 dte>
116.16	USE Ident ON Subsets
116.16	PRESSURE CORRECTIONS
116.16	USE f1 f2 AS { SCALAR } ON Bsubsets { PRESSURE }
116.17	END OPTION DATA
116.17	END DUBLAT DATA

122.0 FLUTTER DATA

The flutter data set described herein defines the altitudes at which V-g flutter solutions are to be performed and the structural damping coefficients which may be essential to the flutter problem. Solution of the flutter equation is performed by the FLUTTER Processor (sec. 222.0) which uses an iterative solution technique based on the automated procedure for computing flutter eigenvalues as published in reference 122-1.

Each set of flutter data is referenced as a CASE. A maximum of 36 FLUTTER data cases may be identified per job. Each data set contains "basic" flutter data and optional groups of modification data which allow variations of the "basic" data to be defined. Each group of modification data is referenced as a change-set (CSET). A maximum of 9 CSETS may be identified per CASE. The degrees of freedom to be used in defining the flutter equation may be selected by the user to be a subset of the freedoms associated with the basic flutter equation. A maximum of 8 different sets of freedoms, each set referenced as a retention-vector set (RSET), may be identified per CASE. These capabilities allow variations of the flutter parameters to be investigated conveniently.

Printout of the flutter input data may be requested from the FLUTTER Postprocessor (sec. 222.1).

~~REPLACES PAGE BLACK NOT FILMED~~

122.1 INPUT DATA

This complete data set is optional. If the default values of the following input data are acceptable, none of the records described in this section should be input.

Record 1 Data Set Identification

BEGIN FLUTTER DATA

This record initiates execution of the FLUTTER Preprocessor.

The "basic" flutter data for the CASE identified by Record 2 are defined by Records 3-5, whereas the CSET (change-set) data are defined by Records 6-9.

Record 2 Data Case Number (optional)

Each set of flutter data is referenced as a case which is identified by an integer number. If multiple sets are used in a job, each case must be assigned a unique number.

CASE n <*/ text */>

n = Data case number (integer) in the range 1 to 36, inclusive.

text = Alphanumeric text of 1 to 80 characters, including blanks, to be associated with the CASE denoted by "n." The "text" information is displayed with the case number when printed or plotted flutter output is requested (see sec. 222.0).
Default: Printed and plotted flutter data are identified by the data case number.

Default Record: CASE 1

If basic CASE data (Records 3-5) are required, they must precede Records 6-9.

Record 3 Altitudes (optional)

ALTITUDE a1 a2 ...

a1,a2,... = List of 1 to 8 altitudes in the range -200,000.0 through 300,000.0 feet (or meters), in any sequence, at which

V-g flutter solutions are desired. The standard atmospheric air density corresponding with each of the specified altitudes is used to factor the air forces associated with the flutter problem. The units of the specified altitudes must correspond with the system of units specified by the EXECUTE FLUTTER statement. If a "matched-point" solution is requested via the MPS parameter, or if the free stream density is specified via the DENSITY parameter of the EXECUTE FLUTTER statement, this record should not be input.

Default Record: ALTITUDE 0.0

Record 4 Structural Damping Coefficients (optional)

DAMPING g1 g2 ...

q_i = The i -th damping coefficient greater than or equal to zero to be associated with the i -th degree of freedom in the flutter equation. The i -th diagonal element of the structural damping matrix is calculated by multiplying the i -th diagonal element of the stiffness matrix by the i -th damping coefficient specified by this record. Trailing zeros need not be input.

Default Record: No structural damping is included in the flutter equation.

Record 5 Retention-Vector Set Definition (optional)

A subset of the NDOF degrees of freedom associated with the basic flutter equation may be identified by this record. A subset of such freedoms is referenced as a retention-vector set (RSET) and is identified by an integer number. If multiple RSETS are required for the CASE identified by Record 2, each RSET must be assigned a unique number.

RSET n f1 f2 . . .

n = Retention-vector set number (integer) in the range 1 to 8, inclusive.

f1,f2,... = List of freedoms to be associated with retention set "n." Each "fi" is the i-th sequential number (integer) of the "basic" sequence of freedoms associated with the flutter equation. Thus, $1 \leq f_i \leq \text{NDOF}$. The freedom numbers may be input in any order.

Default Record: RSET 1 1 2 3 . . . NDOF

This record is repeated to define all RSETS for the CASE identified by Record 2. If a particular RSET is redefined, the last one specified takes precedence.

Change-set modification data for the CASE identified by Record 2 are defined via Records 6-9.

Record 6 Change-Set Data Number (optional)

Variations of the "basic" flutter data established by Records 3-5 may be defined via change-sets (CSETS). Each CSET, which is identified by an integer number, contains modification data to be applied to the "basic" data. If multiple CSETS are required for the CASE identified by Record 2, Records 6-9 are repeated wherein each CSET must be assigned a unique number.

CSET n < *# text #>

n = Change-set data number (integer) in the range 1 to 9, inclusive.

text = Alphanumeric text of 1 to 80 characters, including blanks, to be associated with the CSET identified by "n." The "text" information is displayed with the CASE title (Record 2) when printed or plotted flutter output is requested (see sec. 222.0).

Default: Printed or plotted CSET data are identified by the CSET number.

Default Record: No change-set data are input.

Record 7 Matrix Modifications (optional)

```

{ DAMPING
  MASS
  STIFFNESS } Mod r1 c1 <TO r2 c2 <BY ri1 ci1>> val1

<Mod r3 c3 <TO r4 c4 <BY ri2 ci2>> val2
.
.
.
<Mod rm cm <TO rn cn <BY rin cin>> valn >
  
```

{ DAMPING
MASS
STIFFNESS } = Modifications are specified by this record for the matrix type identified by one of these key-words. These CSET modifications are effected prior to a flutter solution by the FLUTTER Processor (sec. 222.0).

Mod= { RA
 RS
 SA
 SS } = Key-word that identifies what type of modification is to be made to the elements of the specified matrix. The first letter of this key-word denotes that selected elements are to be replaced (R) or scaled (S). The second letter of this key-word denotes that only the matrix elements identified by this record are to be changed (A) or that the changes are to be made to symmetric (S) elements relative to the diagonal of the matrix. If "S" is used, the (i,j) and (j,i) elements are changed accordingly.

r1,c1 = Integer row number and column number, respectively, of the first element to be changed.

r2,c2 = Integer row number and column number, respectively, which identify the last element in a sequence of elements to be changed (r2>=r1 and c2>=c1). A sequence of elements is identified by incrementing "r1" by "ri1" and "c1" by "ci1." If the key-word RS or SS is used, all elements identified by a sequence must be either in the upper or the lower triangular portion of the matrix.

Default: Only the element identified by "r1" and "c1" is to be changed.

ri1,ci1 = Integer row and column number increments, respectively, used to establish a sequence of elements where $0 \leq ri1 \leq (NDOF-1)$ and $0 \leq ci1 \leq (NDOF-1)$. The increments are applied simultaneously to yield the sequence, (r1,c1), (r1+ri1,c1+ci1), . . . , (r1+j*ri1,c1+j*ci1). The maximum value of "j" is such that $r1 + (j+1)*ri1 > r2$ or $c1 + (j+1)*ci1 > c2$.
Defaults: ri1=1 and ci1=0

val1 = The value to be used to replace (R) or scale (S) the matrix elements identified by the sequence.

The remaining items in this record allow other modifications to be made to additional elements of the matrix. The items identify the type of modification to be made, the elements to be changed and value to be used in making the change. Changes are effected in a left-to-right order. Multiple changes for a particular element are cumulated.

Caution: Element identifiers must be in the range 1 to NDOF, inclusive, where NDOF is the order of the basic matrices. Multiple change instructions may be defined by this record provided the maximum number of data items does not exceed 250.

Default Record: The basic matrices are not modified prior to performing a flutter solution.

This record may be repeated to define modifications for the DAMPING, MASS and STIFFNESS matrices. Only one record of this type, however, is allowed for each type of matrix.

Example Records:

MASS SA 1 1 TO 5 1 1.10 SS 5 3 1.30 /

The first 5 elements in column 1 of the mass matrix are scaled (factored) by 1.10 prior to scaling the elements (5,3) and (3,5) by 1.30

STIFF SS 2 1 TO 5 4 BY 1 1 0.99 /

The elements on two sub-diagonals of the stiffness matrix are scaled by 0.99

DAMP PA 3 3 0.00057 /

The third diagonal element of the structural damping matrix is replaced by the value 0.00057

Record 8 Retention Set Identifiers (optional)

A subset of the retention-vector sets defined by Record 5 may be associated with the change-set identified by Record 6.

RSET	Num1	Num2	. . .
------	------	------	-------

Num1, Num2 = List of retention set numbers (integers) to be associated with the CSET identified by Record 6. Each "Numi" must have been defined previously by Record 5.

Default Record: All RSETS defined by Record 5 will be associated with the CSET identified by Record 6.

Record 9 Alternative Altitudes (optional)

A new set of 1 to 8 altitudes may be defined for the change-set identified by Record 6.

ALTITUDE	a1	a2	. . .
----------	----	----	-------

a1, a2, . . . = Same as defined for Record 3

Default Record: The basic altitudes specified by Record 3 are used for this change-set.

Additional cases may be defined by repeating Records 2-9 with a unique identification number assigned to each case by Record 2.

Record 10 End Data Set

END FLUTTER DATA

Additional flutter data cases may be defined by repeating Records 1-10.

Table 122-1. Summary of Flutter Data Records

Reference Page	Data Records
122.2	BEGIN FLUTTER DATA
122.2	CASE n <* ≠ text ≠ >
122.2	<u>ALTITUDE</u> a1 a2 ...
122.3	<u>DAMPING</u> g1 g2 ...
122.3	RSET n f1 f2 ...
122.4	CSET n <* ≠ text ≠ >
122.5	$\left. \begin{array}{l} \text{DAMPING} \\ \text{MASS} \\ \text{STIFFNESS} \end{array} \right\} \text{Mod } r1 \text{ } c1 \text{ } \langle T0 \text{ } r2 \text{ } c2 \text{ } \langle \text{BY } r11 \text{ } c11 \rangle \rangle \text{ } val1$ $\langle \text{Mod } r3 \text{ } c3 \text{ } \dots \rangle$
	$\text{where Mod} = \left\{ \begin{array}{l} \text{RA} \\ \text{RS} \\ \text{SA} \\ \text{SS} \end{array} \right\}$
122.7	RSET Num1 Num2 ...
122.7	<u>ALTITUDE</u> a1 a2 ...
122.8	END FLUTTER DATA

126.0 GEOMETRY DATA

The GEOMETRY Preprocessor allows the user to define and control the three-dimensional geometry of one or more components of a structural model. This preprocessor interrogates and stores the geometry-definition data, develops the component surface geometry and subsequently extracts requested information from the generated data. All interrogations of the generated geometry are initiated by appropriate nodal input data to the NCDAL Preprocessor for definition/extraction of surface or mid-surface nodal coordinates (see sec. 146.0).

Surfaces are generated from the input data by numerical curve-fitting techniques and by enrichment of the geometry input data. The curve-fitting process consists of passing a chain of cubic-polynomial, curve segments (a spline) between adjacent data points such that slope and curvature continuity at the points are maintained. The selected chain of cubics is the one that minimizes the curvature of the entire curve that passes through the corresponding input data points. Optional, user-control of slope and curvature between chains is provided so that surface discontinuities may be defined.

Surface geometries are defined by multiple views of selected control curves. Each control curve is input as a series of curve segments that may be straight, circular or cubic. General, intersecting and non-intersecting curved surfaces may be defined.

A maximum of 61 components may be identified per job. Each component may be used many times in defining a particular model and each component may be associated with different nodal data sets (sec. 146.0).

The geometry data set is comprised of only one data subset by which one or more components is defined. Options are provided for defining components that require detailed geometric data and for components that only require a minimal amount of detail. Direct interrogation of the input and generated geometries cannot be performed by the user. Verification of the extracted nodal coordinates, however, may be performed via printout from the NODAL Postprocessor (sec. 246.1) and via plots generated by the EXTRACT and GRAPHICS Postprocessors (sec. 218.0 and 228.0).

UNRECORDED FILE BLACK NOT FILMED

126.1 INPUT DATA

Geometry data are required only if the user elects to define/generate some or all of the nodal coordinates for a model based on its surface geometries. Input-record data defining component geometries for this purpose must precede the corresponding nodal data. No other data set must precede this data set in the input stream.

Geometry data for a component surface are defined and developed relative to a local, right-handed, rectangular coordinate system x-y-z. Each component is assigned a different label for identification purposes. The same label is used to name the nodal input reference frame (sec. 146.0) when nodal coordinates are extracted from the geometry data for a particular component. Nodes defined in this manner are associated with a unique component (and geometry). Thus, interrogation of a component geometry is initiated by the NODAL Preprocessor by a one-to-one correspondence of the nodal-input frame name and the component name. The orientation of one component relative to another component (and relative to the GLOBAL frame) is established by the selected origins and orientations of the corresponding nodal-data input reference frames.

A set of components may be associated with one or more nodal data sets. A maximum of 61 components may be defined via the GEOMETRY Preprocessor per job and a maximum of 61 components may be referenced by a particular nodal data set.

126.1.1 Control Curve Definitions

The geometry of a component surface is specified by user-selected longitudinal and cross-section control curves that define the surface. Each control curve is defined by points and curve segments in two-dimensional views relative to an x-y-z reference frame.

Longitudinal control curves are used to define locations of surface discontinuities and to identify locations at which nodal coordinates are to be extracted by the NODAL Preprocessor (sec. 146.0). Generally, these curves separate regions on the surface where different curve-fitting rules and constraints are required for surface generation. The longitudinal axis of a component defined by detailed geometry data (sec. 126.1.2.1) must be the x-axis, whereas the longitudinal axis of a component defined by simplified geometry data (sec. 126.1.2.2) must be the y-axis. A longitudinal control curve is defined by its side view and top view as illustrated in figures 126-1 and 126-2. To assure that a smooth, continuous surface is generated,

it is necessary that both views of each longitudinal curve are smooth and continuous.

Cross sections of a component are established at selected longitudinal locations by input of "true-view" cross-section control curves. For a "detailed-geometry" component, these curves are defined in planes that are perpendicular to the x-z plane as shown in figure 126-1. Additionally, the cross-section planes are perpendicular either to the x-axis or to a cross-section-axis control curve defined by the user (see fig. 126-1). A section axis must be contained within the x-z plane. For a "simplified-geometry" component, cross-section planes must be perpendicular to the y-axis as shown in figure 126-2.

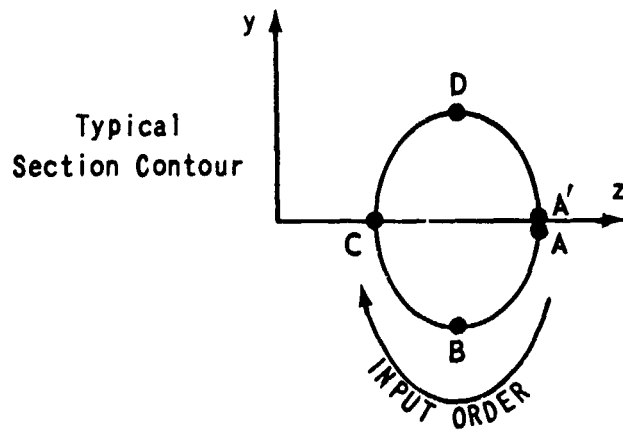
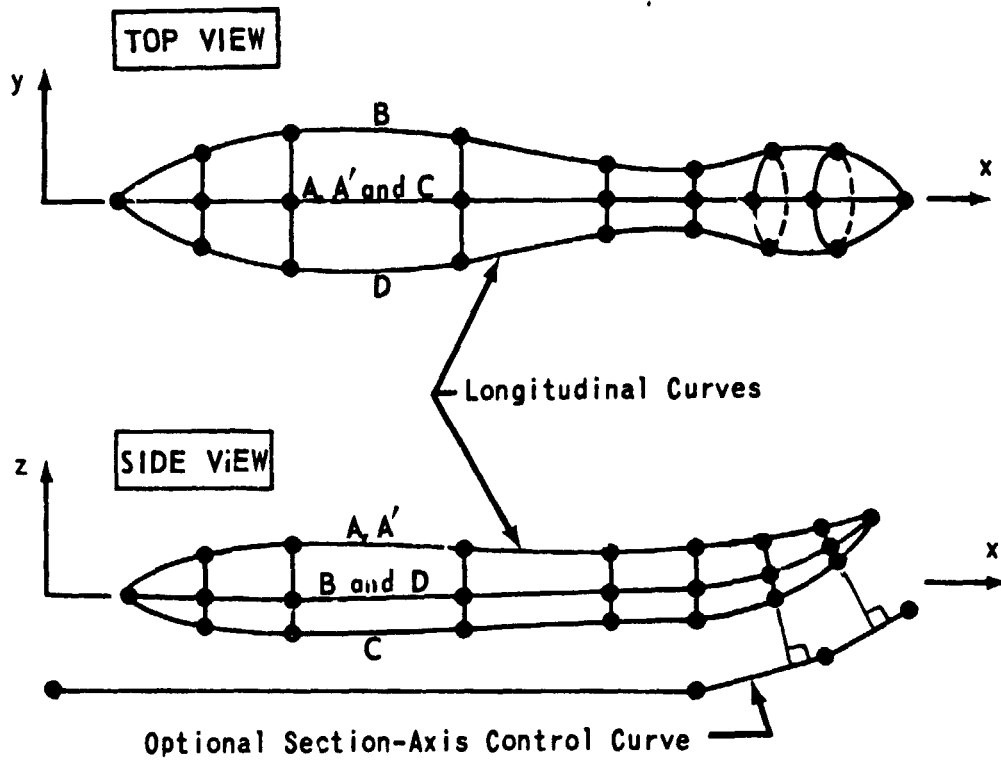
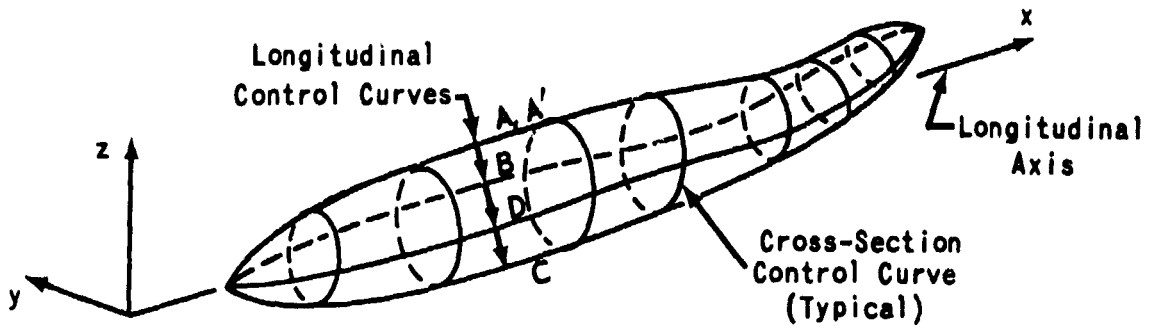
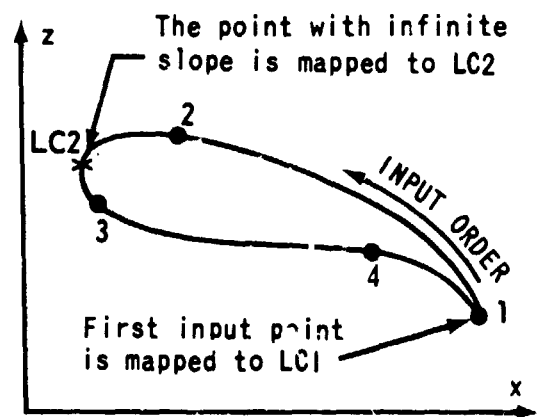
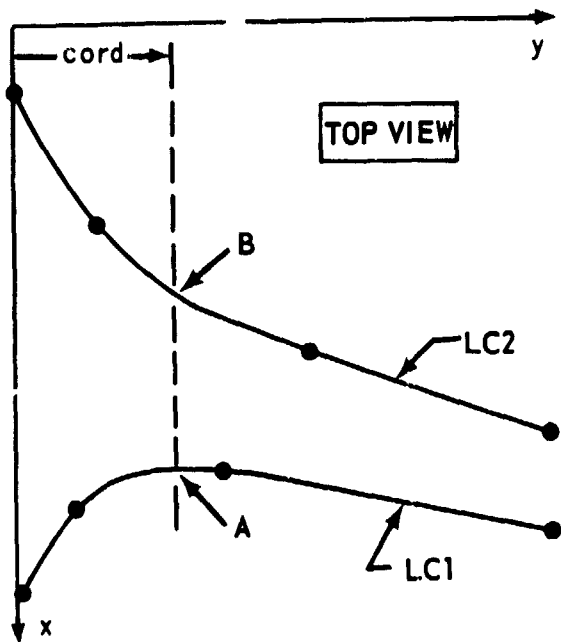
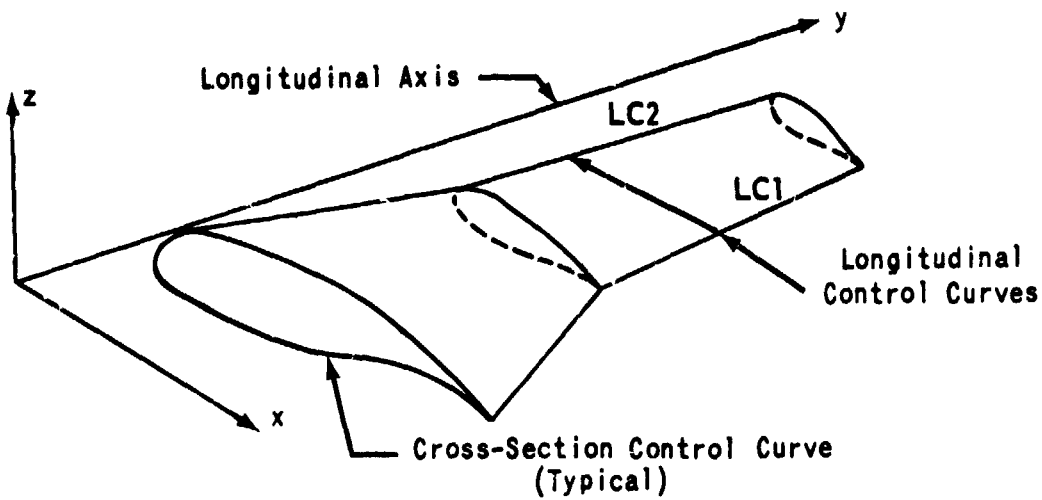


Figure 126-1. Reference Frame and Control Curves for Detailed Geometry Components



Typical Section Contour at "cord"

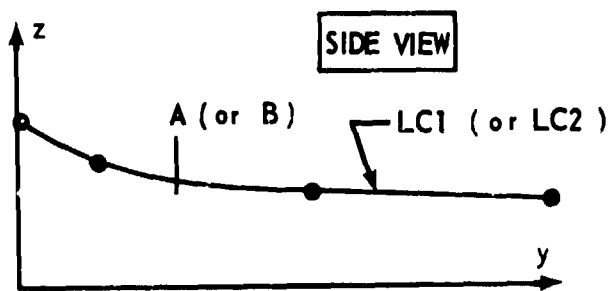


Figure 126-2. Reference Frame and Control Curves for Simplified Geometry Components

Each two-dimensional view of a control curve is defined as a series of straight, circular and/or cubic curve segments with optional slope and curvature constraints. All curve segments are defined by one type of input-data record and all constraint data are specified by another type of record. These two types of records may be used in several places within a geometry data set. Thus, the formats of these records are presented here and referenced as Record A and Record B in the subsequent data-record descriptions. Examples of these records are shown by figure 126-3. Since the control curves are defined relative to different planes associated with the local x-y-z reference frame, the following discussion identifies the input frame as u-v-w wherein the association with the x-y-z triad is established by the type of view.

Record A--Control Curve Definition

This record defines the two-dimensional loci of a control curve in terms of one or more curve segments through selected points established by u-v coordinates.

Seq1 Seq2 ... Seqn

Seqi = The i-th list of 3 or more items defining the i-th curve segment. "Seqi" must be one of the following:

}	<u>STRAIGHT</u> u1 v1 <u2 v2 ... un vn>
	<u>CUBIC</u> u1 v1 u2 v2 <u3 v3 ... un vn>
	<u>CIRCLE</u> u1 v1 <u2 v2> uc vc
	<u>CIRCLE</u> u1 v1 <u2 v2> r
	<u>CIRCLE</u> uc vc r

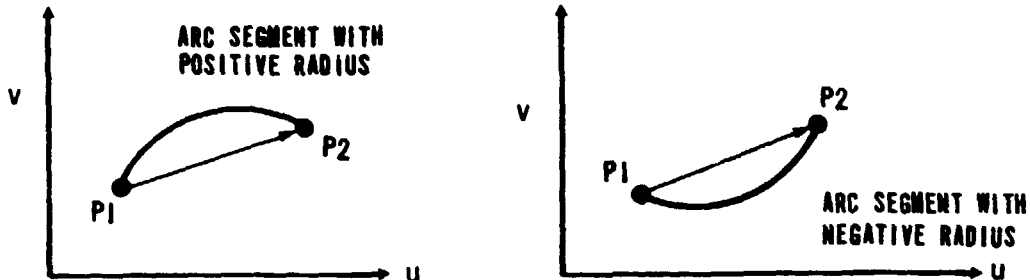
The type of curve segment is identified by the key-word and the remaining items define the geometrical points to be used in fitting the control curve.

ui,vi--The u-v coordinates of a point through which the segment passes

uc,vc--The u-v coordinates of the center of a circular arc

r--Radius of a circular arc. This item may either be positive or negative to orient an arc segment relative to the curve itself and relative to the u-v axes. A positive radius places the center of curvature on the right-hand

side of the vector that goes between two successive points used to define the curve (e.g, points P1 and P2 in the following sketches). A negative radius places the center of curvature on the left-hand side of the vector.



Any combination of up to 99 STRAIGHT, CIRCLE and CUBIC segments may be used to define the trace of a control curve. Each trace must be contiguous and must be specified, in a positive order, from the beginning to the end of the curve. Positive directions of the control curves are identified where Record A is referenced.

The u_1, v_1 coordinates specified for "Seg i " define the beginning point of the i -th segment and the end point of the $(i-1)$ segment. The coordinates of the last point used to define a curve are specified by the items u_2, v_2 or u_n, v_n . Thus, for example, a curve comprised of any combination of segment types that passes through three points requires only six coordinates to be input.

Certain characteristics of each type of curve segment are as follows:

STRAIGHT--May be specified to pass through two or more points.

CUBIC--A chain of cubics may be specified to pass through three or more points.

CIRCLE--Generally, an arc segment is defined by its center (u_c, v_c) or radius (r) and its two end points. The orientation of an arc segment defined by its radius is established by the sign of " r " as described above. A semi-circle must be defined by specifying its radius.

The option for "Segi" denoted by CIRCLE uc vc r is applicable only for the simplified, geometry input-data option (sec. 126.1.2.2). In this case, the control curve is completely defined by one segment.

Circles and arc segments are mapped into chains of cubic equations by the GEOMETRY Preprocessor. Each arc segment is simulated by fitting a cubic spline through 10 equally-spaced points generated by the program. Caution must thereby be exercised when interrogating geometry data defined by CIRCLE segments.

Record E--Curve Segment Constraints

This record is used to enforce slope and curvature constraints at selected points on CIRCLE and CUBIC segments of the control curve defined by the immediately preceding Record A. Constraints may be specified at curve-segment boundary points or at points within curve segments. If Record B is not input, slope continuity at all curve-segment boundary points is automatically enforced. Geometry alterations or constraints may not be applied to STRAIGHT segments.

Con1 Con2 ...

Coni = The i-th list of 3 or more items defining the i-th constraint. "Coni" must be one of the following:

{ SLOPE Cord Vals }
 { CURVATURE Cord Vals }

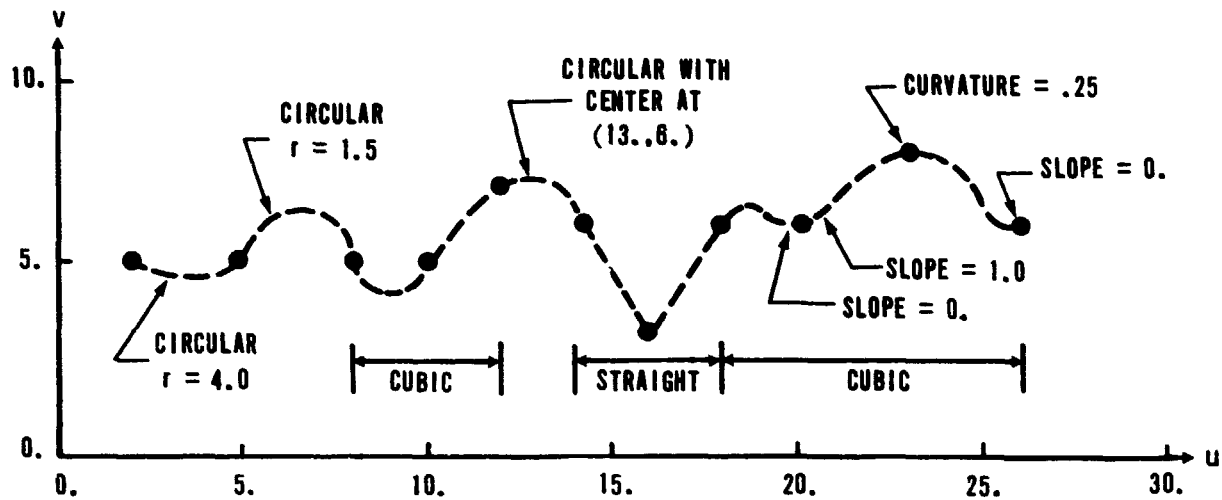
Cord--A u-coordinate that identifies the location of a point on the control curve at which either the SLOPE or the CURVATURE (reciprocal of radius-of-curvature) is specified by "Vals." Constraints for a curve may be input in any order.

Vals--A list of one or more items which must be one of the following:

{	value1			}
	<u>INCOMING</u>	value1		
	<u>OUTGOING</u>	value2		
	<u>INCOMING</u>	value1	<u>OUTGOING</u>	value2

valuei--The value of the SLOPE or CURVATURE to be enforced at the point established by "Cord." A positive slope corresponds to a positive value of dv/du . The keyword INFINITE may be input for "valuei" to denote an infinite slope. Curvatures may be positive or negative to define the proper orientation of the curve relative to the u-v axes. The sign specified for a curvature is treated in the same manner as the sign associated with a CIRCLE radius as discussed above.

INCOMING } --Key-words that identify the INCOMING
OUTGOING } and OUTGOING segments of the curve at a segment-boundary input point. The direction of the curve is established by Record A. SLOPE or CURVATURE values may be enforced on the INCOMING side and/or the OUTGOING side as denoted by the foregoing three options.



Two-Dimensional Loci of a Control Curve

RECCED A--Control Curve Definition

CIR	2.	5.	-4.	CIR	5.	5.	1.5	CUB	8.	5.	10.	5.
CIF	12.	7.	13.	6.	STR	14.	6.	16.	3.			
CUE	18.	6.	20.	6.	23.	8.	26.	6.	/			

RECCED E--Curve Segment Constraints

SLC	20.	INC	0.	OUT	1.0	CUR	23.	.25	SLO	26.	0.	/
-----	-----	-----	----	-----	-----	-----	-----	-----	-----	-----	----	---

Figure 126-3. Example Control-Curve-Definition and Curve-Segment Constraint Records

126.1.2 Component Geometry Input Data

Record 1 Data Set Identification

BEGIN GECMETRY DATA

This record initiates execution of the GECMETRY Preprocessor.

Options are provided for defining components that require detailed geometric data and for components that require only a minimal amount of detail. Either the detailed geometry-data option (Records 2-14) or the simplified data option (Records 15-26) may be used to define the geometry of a component. The primary differences between these options are as follows:

- a) A user-selected number of longitudinal curves are input to define a detailed geometry, whereas only two longitudinal curves are used to define a simple geometry.
- b) There is an option to have the GEOMETRY Preprocessor automatically ENRICH the number of longitudinal curves for a detailed geometry by generation of additional control curves. Enrichment is required for a simple geometry.
- c) A cross-section axis may be defined via Records A and B for a detailed geometry. In this case, the input cross-section control curves lie in planes that are normal to the section axis. If no section axis is defined or if the simple geometry option is selected, all input cross-section curves lie in planes that are normal to the longitudinal axis.
- d) A detailed-geometry component may have open or closed cross-section contours, whereas a simple-geometry component must have closed section contours.
- e) There is an option, only for simple geometries, to rotate the input cross-section control curves after they are defined. Rotations are effected within the plane of the control curve and about a user-selected point within the plane.

The number of control curves used to define the geometry of a component by either option must satisfy the expression

$$12 * NLC * NCS < 12000$$

where NLC is the number of longitudinal control curves and NCS is the number of cross-section control curves.

Non-intersecting and intersecting curved surfaces may be defined via either the detailed geometry data or the simple geometry data. Surface and/or mid-surface nodal coordinates may be subsequently extracted for components defined in either manner. Mid-surface nodes may be defined for components with open or closed section contours (see sec. 146.0).

126.1.2.1 Detailed Geometry Data

If the component is to be defined via simple geometry data, Records 2-14 should not be input.

Record 2 Component Identifier

BEGIN COMPONENT <label>

label = An alphanumeric word of 1 to 7 characters that identifies the component defined by Records 3-14 and identifies the local x-y-z reference frame used to define that component. If more than one component is required for a particular job, each "label" must be different. The word INPUT may not be used for "label."
Default: GLOBAL--this key-word is associated with a component that is defined relative to the GLOBAL frame.

A maximum of 30 longitudinal control curves may be used to define a detailed-geometry component. Each curve is assigned a unique label for subsequent identification. The order in which the curves are described must be sequential from one edge of the surface to the other edge (see fig. 126-1). It should be noted that if the component has closed cross-section contours, two longitudinal control curves defined by similar coordinates must be input. The sequence in which the longitudinal control curves are input establishes the order in which the curve segments for each of the cross-section control curves are input.

The SIDE view of each longitudinal control curve used to describe the component identified by Record 2 is defined by repeating Records 3 and 4. Each curve is identified by Record 3 and the curve segments are described by Record 4.

Record 3 Side View of a Longitudinal Curve

SIDE VIEW name

name = An alphanumeric word of 1 to 10 characters that identifies the longitudinal control curve described by Record 4. The words LC1 and LC2 may not be used. Each "name" associated with a component must be different.

Record 4 Side-View Definition of Longitudinal Curve

Record A or Records A and B

The direction of the curve defined by Record A must be along the x-axis. The curve must also be single-valued with respect to x. Data input by these records are:

u = x-coordinate of a point used to define the curve segments for the side view of the control curve.

v = Corresponding z-coordinate of the point.

value = Positive slopes correspond to positive values of dz/dx .

Records 3 and 4 are repeated to define at least 2 longitudinal curves.

The TOP views of each longitudinal control curve described by Records 3 and 4 are defined by Records 5 and 6. The order in which the curves are described by these records need not correspond to the order in which they are described by Records 3 and 4. Records 5 and 6 need not be input for a curve whose top-view is coincident with the x-axis.

Record 5 Top View of a Longitudinal Curve

TCP VIEW Name

Name = The "name" previously assigned to a control curve by Record 3. The TOP-

view trace of this curve is described by Record 6.

Record 6 Top-View Definition of Longitudinal Curve

Record A or Records A and B

The direction of the curve defined by Record A must be along the x-axis. The curve must also be single-valued with respect to x. Data input by these records are:

- u = x-coordinate of a point used to define the curve segments for the top view of the control curve.
- v = Corresponding y-coordinate of the point.
- value = Positive slopes correspond to positive values of dy/dx .

Records 5 and 6 are repeated to define the top view of each longitudinal control curve. By default, the top view of a curve is assumed to be the x-axis.

Record 7 Enrichment of Longitudinal Curves (optional)

The user may request a maximum of 70 additional longitudinal curves to be generated between selected, user-defined, longitudinal curves. The additional curves are generated via a cubic interpolation technique. These curves are equally-spaced on the section contour between the selected, input, control curves. This option provides a convenient method of increasing the number of control curve points on a sectional cut for extraction of additional nodal coordinates (ref. sec. 146.0).

Variation 1 ENRICH BY num

- num = Integer number of additional control curves to be generated between each pair of user-defined longitudinal curves.

Variation 2 ENRICH Name1 TO Name2 BY num1
<Name3 TO Name4 BY num2
.
.
.
NameM TO NameN BY numK>

- Name_i = The "name" previously assigned to a longitudinal control curve by Record 3.
- num_i = Integer number of curves to be generated between each pair of user-defined control curves included in the i-th sequence of curves identified by "Name1" and "Name2," "Name3" and "Name4," etc.
- Default Items: Additional curves are not generated between the user-defined control curves that are not identified by this record.

Either Variation 1 or Variation 2 of this record may be used for a particular component. Variation 2 of this record may be repeated.

Default Record: No enrichment of the input longitudinal control curves is performed.

Cross-section control curves are defined relative to planes which are normal either to the longitudinal x-axis or to a cross-section axis (a control curve) defined by Records 8 and 9. If any cross-section plane is not perpendicular to the x-axis, a section axis must be defined. A section axis must be contained within the x-z plane. Thus, it is defined wholly by its SIDE VIEW (see fig. 126-1). If all cross-section planes are normal to the x-axis, Records 8 and 9 should not be input.

Record 8 Section Axis Indicator (optional)

SECTION AXIS

Record 9 Side-View Definition of Section-Axis Curve (optional)

Record A or Records A and B

The direction of the curve defined by Record A must be along the x-axis. The curve must span the full length of the component and it must be single-valued with respect to x. Data input by these records are:

- u = x-coordinate of a point used to define the curve segments for the section-axis control curve.
- v = Corresponding z-coordinate of the point.

value = Positive slopes correspond to positive values of dz/dx .

Default Records: If Records 8 and 9 are not input, the cross-section curves defined by Records 10-13 are contained within y-z planes.

Records 10-13 are repeated to define at least two and a maximum of 49 cross-section curves for the component.

Record 10 Cross Section Location

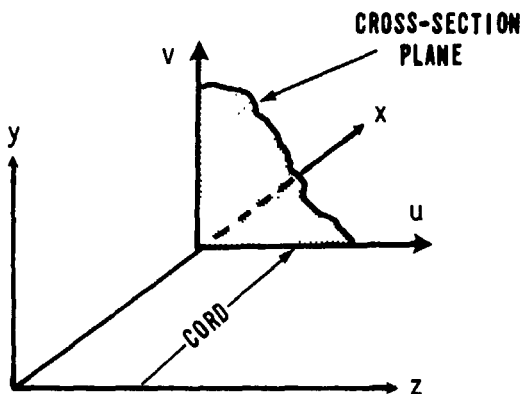
SECTION cord <SAME>

cord = x-coordinate that defines the location at which a cross-section contour is described by Records 11-13.

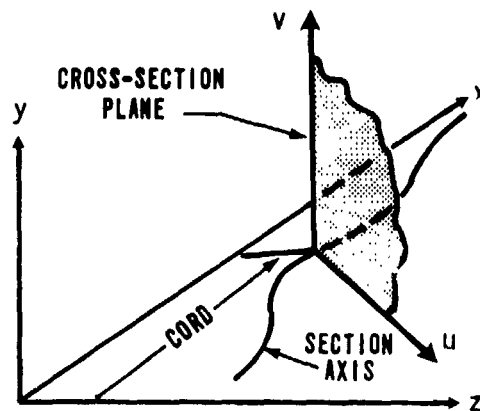
SAME = Key-word denoting that the cross-section contour at "cord" is the same as that defined by the immediately-preceding Records 11-13.

Default: The section contour at "cord" is defined by the next Records 11-13.

Curve segments defining the "true-view" of a cross-section control curve are input via Record 11. The u-v coordinates input by this record correspond to z-y coordinates if a section axis is defined by Records 8 and 9. That is, the origin of the u-v axes is on the x-axis and the u-axis is parallel to the z-axis as shown below. If a section axis is defined, the origin of the u-v axes is on the section axis. For this case, the u-axis is normal to the section axis and the v-axis is parallel to the y-axis.



Default Section Axis



User-Defined Section Axis

Record 11 Cross-Section Control-Curve Definition

Record A

The direction of the curve defined by this record must be as shown in figure 126-1. The order in which a section contour is input must be consistent with the order in which the longitudinal control curves are defined by Record 3. Thus, the first point used to describe a section control curve must correspond to the intersection of the first longitudinal control curve with the cross-section plane, whereas the last point must correspond to the last longitudinal control curve. The u-v coordinates of points input by this record are illustrated above. At least those points where the longitudinal control curves intersect the cross-section plane must be established by this record. These input points must map onto the intersection points with no distortion.

Either cubic or straight curves may be fitted through the section-contour points by Record 12. Slopes of the section contour at points corresponding to the longitudinal control curves may be defined by Record 13.

Record 12 Cross-Section Control-Curve Fitting Data

FIT	Name1	<div style="border-left: 1px solid black; border-right: 1px solid black; padding: 2px;"> <u>CUBIC</u> <u>STRAIGHT</u> </div>	Name2	<div style="border-left: 1px solid black; border-right: 1px solid black; padding: 2px;"> <u>CUBIC</u> <u>STRAIGHT</u> </div>	Name3 ...
-----	-------	---	-------	---	-----------

Name_i = The "name" of a longitudinal control curve. The order in which "Name_i" is input by this record must be consistent with the order in which the control curves are defined by Record 3. "Name1" must be the name of the first curve. The name of the last longitudinal control curve must be identified by this record. Not all of the control curves, however, need be identified by this record.

CUBIC
STRAIGHT

 = Key word denoting that a CUBIC or STRAIGHT curve fit is to be used between "Name1" and "Name2" or between "Name2" and "Name3" etc.
 Default: CUBIC

Example: Longitudinal control curves identified by the names A through G were defined by Record 3. Twenty points were used to define a section contour by Record 11. The coordinates of 7 of these points that correspond

to the longitudinal curves must be defined carefully such that mapping of these points onto curves A-G is performed properly. If a straight line is to be used between curves B-C, C-D and F-G, and a cubic spline is to be used elsewhere, the input record could be
FIT A B STR D CUB F STR G /

**Record 13 Cross-Section Control-Curve Slope Constraints
(optional)**

SLOPE Name1 Vals Name2 Vals ...

Name1 = The "name" of a longitudinal control curve at which the constraint is to be applied.

Vals = Same as defined for Record B for SLOPE constraints.

Default Record: Slopes of the cross-section control-curve are established from Records 11 and 12.

Records 10-13 are repeated to define at least 2 or a maximum of 49 cross-section curves for a component.

Record 14 End Component Data (optional)

END COMPONENT DATA

This record denotes that the geometry data for a component have been defined. Additional components may be defined by additional Records 2-14 or by Records 15-26 described below.

126.1.2.2 Simplified Geometry Data

If a component is defined via detailed geometry data (Records 2-14), Records 15-26 should not be input.

Record 15 Component Identifier

BEGIN COMPONENT <label>

label = An alphanumeric word of 1 to 7 characters that identifies the component defined by Records 16-26 and identifies the local x-y-z reference frame used to define that component. If more than one component is defined for a particular

job, each "label" must be different. The word INPUT may not be used as a label.

Default: GLOBAL--This key-word is associated with a component that is defined relative to the GLOBAL frame.

A "simple geometry" component is defined by two longitudinal control curves identified as "LC1" and "LC2." As illustrated in figure 126-2, these curves should not cross in the TOP view. The TOP views of these curves are described by Records 16-19. The SIDE view of one of these curves, as defined by Records 20 and 21, describes the "shear" characteristic of the component relative to the x-y reference plane. As seen in the TOP view, the edge of the component surface defined by the largest x-coordinates must be identified by the curve LC1.

Record 16 Top View Indicator

TOP VIEW LC1

Record 17 Top-View of Longitudinal Curve LC1

Record A or Records A and B

The direction of the curve defined by Record A must be along the y-axis. The curve must also be single-valued with respect to y. Data input by these records are:

u	=	y-coordinate of a point used to define the curve segments for the top view of the control curve.
v	=	Corresponding x-coordinate of the point.
value	=	Positive slopes correspond to positive values of dx/dy.

Record 18 Top View Indicator

TOP VIEW LC2

Record 19 Top-View of Longitudinal Curve LC2

Record A or Records A and B

Same as Record 17.

Record 20 Side View Indicator

SIDE VIEW { LC1 }
 { LC2 }

Record 21 Side-view of Longitudinal Curve LC1 or LC2

Record A of Records A and B

The direction of the curve defined by Record A must be along the y-axis. The curve must also be single-valued with respect to y. Data input by these records are:

- u = y-coordinate of a point used to define the curve segments for the side view of LC1 or LC2.
- v = Corresponding z-coordinate of the point.
- value Positive slopes correspond to positive values of dz/dy.

Cross-section control curves are defined relative to planes which are normal to the y-axis. The u-v input plane is positioned relative to the longitudinal axis by Record 22 and the control curve is defined by Record 23. Each cross-section control curve is scaled and fitted automatically to the LC1 and LC2 longitudinal curves at the specified longitudinal position. Non-dimensional u-v coordinates may, therefore, be used to define cross-section control curves. "Twist" characteristics of the component may optionally be specified for selected cross-section curves by Record 24.

Record 22 Cross Section Location

SECTION cord

- ccrd = y-coordinate that defines the location at which a cross-section contour is described by Records 23 and 24.

Record 23 Cross-Section Control-Curve Definition

Record A

The direction of the curve defined by this record must be as shown in figure 126-2. The curve must be single-valued for the maximum and minimum x-coordinates defining the section contour. The first and last points of the cross-section curve must correspond to the longitudinal

curve LC1. The point on the section curve with the smallest x-coordinate (infinite slope) is automatically mapped onto curve LC2. The section contour is scaled accordingly. Data input by this record are:

- u = x-coordinate or non-dimensional x-coordinate of a point used to define the curve segments for the section contour.
- v = Corresponding z-coordinate or non-dimensional z-coordinate of the point.

Record 24 Rotation of Input Cross Section (optional)

ROTATE angle ABOUT x z

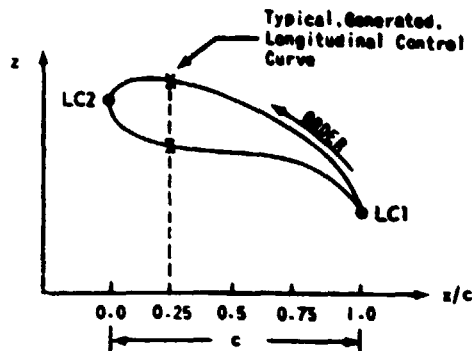
angle = Angle of rotation (degrees) in the plane of the section defined by the immediately-preceding Record 23. A positive "angle" corresponds to a positive, right-handed rotation about the y-axis. The rotated curve is automatically mapped to the longitudinal curves as described above.

x,z = x and z coordinates that define a pivot point contained in the section plane.

Default Record: The cross-section curve as defined by Record 23 is not rotated prior to mapping it to LC1 and LC2.

At least two cross-section curves must be defined by Records 22-24. These records may be repeated to define a maximum of 49 cross-section contours for a component.

Longitudinal control curves through selected points on the cross-section contours are generated by the program according to Record 25. The points on the previously-defined section contours through which longitudinal curves are to be generated are selected for definition of surface discontinuities and/or for identifying locations at which nodal coordinates are to be extracted by the NODAL Preprocessor (sec. 146.0). The geometry of a "simple" component, as saved for subsequent interrogation to extract nodal coordinates, is described strictly by these longitudinal control curves. Point locations on the section contours are specified in terms of the fraction of cross-section chord length measured along the x-axis as shown in the following sketch.



Record 25 Enrichment of Longitudinal Curves

ENRICH AT f1 f2 ...

f_i = The fraction of the x-axis chord, $0.0 \leq f_i \leq 1.0$, at which longitudinal control curves are to be generated. The order in which the "fi" are input must be from LC1 (1.0) along the upper surface to LC2 (0.0) and back to LC1 (1.0) along the lower surface. If control curves are to be generated for the upper surface and the lower surface at the same chord fractions, the first "fi" must be 1.0, whereas the last "fi" could be input as zero. A maximum of 99 fractions may be specified.

Example: If control curves are to be generated for the upper and lower surfaces at the quarter-chord as shown in the foregoing sketch, ($f_1=0.25$), one of the following records should be input.

```
ENRICH AT 1.0 0.25 0.0 0.25 1.0 /
ENRICH AT 1.0 0.25 0.0 /
```

Record 26 End Component Data (optional)

END COMPONENT DATA

This record denotes that geometry data for a component have been defined. Additional components may be input by repeating Records 2-14 or Records 15-26.

Record 27 End Geometry Data Set

END GEOMETRY DATA

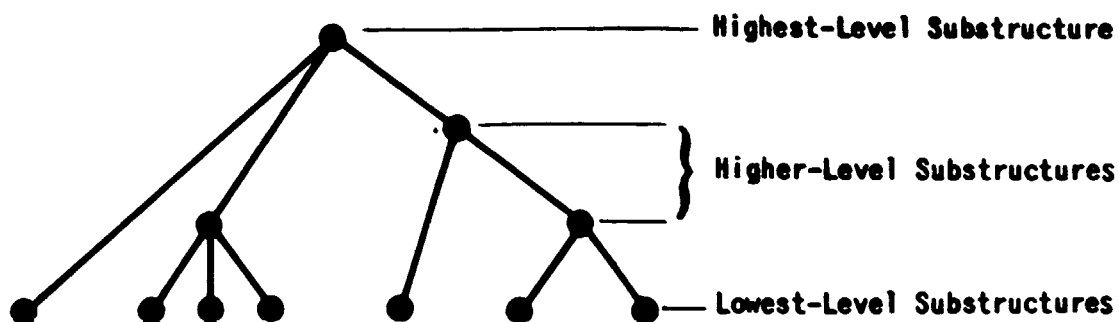
Additional geometry data sets may be input by repeating Records 1-27.

Table 126-1. Summary of Geometry Data Records

Reference Page	Data Records
126.6	<p>Record Type A</p> <p>Seg1 Seg2 ...</p> <p>where</p> $\text{Segi} = \left\{ \begin{array}{ll} \text{STRAIGHT} & u1 \ v1 \ \langle u2 \ v2 \ \dots \ un \ vn \rangle \\ \text{CUBIC} & u1 \ v1 \ u2 \ v2 \ \langle u3 \ v3 \ \dots \ un \ vn \rangle \\ \text{CIRCLE} & u1 \ v1 \ \langle u2 \ v2 \rangle \ uc \ vc \\ \text{CIRCLE} & u1 \ v1 \ \langle u2 \ v2 \rangle \ r \\ \text{CIRCLE} & uc \ vc \ r \end{array} \right\}$
126.8	<p>Record Type B</p> <p>Con1 Con2 ...</p> <p>where</p> $\text{Coni} = \left\{ \begin{array}{ll} \text{SLOPE} & \text{Cord Vals} \\ \text{CURVATURE} & \text{Cord Vals} \end{array} \right\}$ <p>and</p> $\text{Vals} = \left\{ \begin{array}{ll} \text{value1} & \\ \text{INCOMING} & \text{value1} \\ \text{OUTGOING} & \text{value2} \\ \text{INCOMING} & \text{value1} \ \text{OUTGOING} \ \text{value2} \end{array} \right\}$
126.11	BEGIN GEOMETRY DATA
126.12	BEGIN COMPONENT <label>
126.13	SIDE VIEW name
126.13	Record A or Records A and B
126.13	TOP VIEW Name
126.14	Record A or Records A and B
126.14	ENRICH BY num
126.14	ENRICH Name1 TO Name2 BY num1 <Name3 TO Name4 BY num2 ...>
126.15	SECTION AXIS
126.15	Record A or Records A and B
126.16	SECTION cord <SAME>
126.17	Record A
126.17	FIT Name1 <CUBIC STRAIGHT> Name2 <CUBIC STRAIGHT> Name3 ...
126.18	SLOPE Name1 Vals Name2 Vals ...
126.18	END COMPONENT DATA
126.18	BEGIN COMPONENT <label>
126.19	TOP VIEW LC1
126.19	Record A or Records A and B
126.19	TOP VIEW LC2
126.19	Record A or Records A and B
126.20	SIDE VIEW { LC1 } { LC2 }
126.20	Record A or Records A and B
126.20	SECTION cord
126.20	Record A
126.21	ROTATE angle ABOUT x z
126.22	ENRICH AT f1 f2 ...
126.22	END COMPONENT DATA
126.22	END GEOMETRY DATA

130.0 INTERACT (SUBSTRUCTURE) DATA

An ATLAS substructured model and analysis scheme are defined according to an "interact tree" as shown in the graph below.



Each node in this graph represents a substructure component of the interact tree, whereas the branches denote how the substructures are interacted. Each lowest-level substructure is defined as (equivalenced to) a previously-defined stiffness or stiffness/mass data SET and a corresponding boundary-condition STAGE. The manner in which these components are to be interacted to form all higher-level substructures of the interact tree is defined via the interact data.

General rules which must be satisfied for all substructure analyses are as follows:

- a) All structural data for the lowest-level substructures must be defined for the "equivalent" SET/STAGE components.
- b) The same GLOBAL frame must be associated with each of the lowest-level substructures.
- c) Nodes that interact between component substructures must have the same analysis frame and the same geometric location (to within the distance tolerance specified by Record 5).
- d) The kinematic freedoms of nodes that interact between component substructures must be assigned the RETAIN freedom activity (reference sec. 106.0 and the following discussion).
- e) Each substructure is assigned a unique integer number in the range 1 to 999, inclusive. A maximum of 999 substructures may be identified in a single analysis.

PRECEDING PAGE BLANK NOT FILMED

- f) A maximum of 28 substructures (lowest-level, higher-level or a combination thereof) may be interacted to form a new higher-level substructure in the interact tree. There is no explicit limit on the number of interact levels.
- g) The nodes and finite elements associated with a lowest-level substructure are those defined for the "equivalent" SET/STAGE structural model. No finite elements are associated explicitly with the higher-level substructures. The nodes associated with a higher-level substructure include only those nodes with freedoms retained for the next lower-level substructure components thereof.
- h) Loads and displacements may be specified only for the lowest-level substructures. The load case labels associated with each of these components must be unique. These criteria must be satisfied by the loads data (sec. 134.0) for the "equivalent" SET/STAGE components. Loads associated with load cases that have identical labels for two or more lowest-level substructures are cumulated.
- i) Mass data may be specified either for the equivalent SET/STAGE components of the lowest-level substructures or for the highest-level substructure that is "equivalenced" to a SET/STAGE component (Record 17 of the interact data set). If the mass associated with any stiffness elements of lowest-level substructures is to be considered, all mass data for the substructured model must be defined relative to these equivalent SET/STAGE models. In this case, the same mass CONDITION number must be assigned to each of these mass data sets (see sec. 138.0). Mass data may be defined directly for a highest-level substructure strictly by a mass data set for the equivalent SET/STAGE model. In this case, the mass data must be input after the interact data set and the mass data must be defined strictly via the nodes associated with the highest-level substructure. These nodes are the ones with freedoms retained for the substructure components of the highest-level substructure.

Stress analyses and dynamic analyses of substructured models may be performed. Dynamic analyses of a highest-level substructure are performed by using the reduced stiffness and mass matrices for that substructure.

The boundary condition (BC) kinematic-freedom activities (see sec. 106.0) of the nodes associated with a lowest-level substructure are established according to the following distinct steps:

1. The kinematic activities as defined by the BC data (sec. 106.0) for the "equivalent" SET/STAGE model are the basic freedom activities. These activities are modified according to the subsequent steps.
2. The activity label for those freedoms that interact in forming higher-level substructures must be RETAIN. The user may identify these freedoms either directly by the BC data or by a particularly user-convenience which automatically retains all relevant freedoms of interacted nodes (see Record 7). This technique uses the criteria that any FREE or RETAIN freedoms in component substructures which have the same node analysis frame and the same geometric location (to within the distance tolerance specified by Record 5) interact in forming the next higher-level substructure. These freedoms are automatically assigned the RETAIN freedom activity. The additional criterion that interacting nodes are identified by the same user-assigned node numbers is effected by use of Record 2. If the STIFFNESS Processor is executed for the "equivalent" SET/STAGE of a lowest-level substructure prior to input of the Interact data, all inactive freedoms (those with zero stiffness) are ignored by the automatic retention technique. Otherwise, all freedoms which interact according to the foregoing criteria are automatically retained.
3. The foregoing steps, which may be performed automatically, define the default freedom activities. This freedom-activity assignment technique is completely sufficient for most cases. Modifications to the default freedom activities may be defined, as necessary, via the "BC change-data" subset of the Interact data (sec. 130.1.2). Additionally, the user may suppress the automatic retention of interacted freedoms via Record 7 (sec. 130.1.1).

In all cases, the last activity label specified for a particular freedom takes precedence.

The BC kinematic-freedom activities for the nodes associated with a higher-level substructure are established according to the following distinct steps:

1. A freedom that is retained in two or more component substructures and that interacts in forming the higher-level substructure is automatically assigned the FREE freedom activity.
2. The freedom activity labels for all retained freedoms in the component substructures that do not interact remain to be RETAIN.
3. All inactive freedoms of a node associated with a higher-level substructure (freedoms not retained for the component substructures) are assigned the SUPPORT freedom activity.

The kinematic activities defined by the foregoing three steps are the basic freedom activities for a higher-level substructure. These activities may be modified according to steps 4 and 5.

4. If a higher-level substructure is used as a component for another higher-level substructure, the activity label of those freedoms that interact must be RETAIN. These freedoms may be identified by the automatic retention technique described in step (2) above for a lowest-level substructure. In this case, however, all FREE and RETAIN freedoms of the interacting higher-level substructures are used in performing the automatic retention.
5. The foregoing steps, which may be performed automatically, define the default freedom activities. This freedom-activity assignment technique for higher-level substructures is completely sufficient for most cases. Modifications to the default freedom activities may be defined via the "BC change-data" subset (sec. 130.1.2). A freedom, for example, which has interacted in forming a particular substructure, and which has thus been automatically assigned the FREE activity label per step 1 above, must be reassigned the RETAIN freedom activity if it is to interact in forming another higher-level substructure. The user may suppress the automatic retention of interacted freedoms via Record 7 (sec. 130.1.1).

The basic ordered-list of retained freedoms for a lowest-level substructure is defined via the BC-data for the "equivalent" SET/STAGE. Additional retained freedoms identified by the interaction requirements (or automatic interact freedom retention) and by any "BC change data" are added, in an internal node-number order, to the end of the basic list. Additionally, any initially-retained freedoms for which the activity label

changed to FREE or SUPPORT via the "BC change-data" subset are deleted from the list. The resulting list defines the default sequence of retained freedoms for a lowest-level substructure.

The basic ordered-list of retained freedoms for a higher-level substructure is formed from the lists of retained freedoms associated with the component substructures. The list begins with the sequence of freedoms retained for the first component substructure followed by those associated with the second component substructure, etc. Component substructures are identified by Record 6 of the Interact data set. Additional retained freedoms identified by the interaction requirements and by any "BC change data" are added, in an internal node-number order, to the end of the basic list. Additionally, any initially-retained freedoms for which the activity label is changed to FREE or SUPPORT via the "BC change-data" subset are deleted from the list. The resulting list defines the default sequence of retained freedoms for a higher-level substructure.

If the default sequence of retained freedoms for a substructure is not satisfactory, a different sequence may be defined via the "BC change-data" subset. Ordering of the retained freedoms for any substructure within an interact tree may be specified as required (see sec. 130.1.2).

All substructure analyses are most conveniently performed by use of the ATLAS-System, standard, cataloged control statements (see sec. 200.0 and appendix E). All matrix operations performed via these control procedures are described in appendix D.

The interact data set, as described in the next section, is divided into the following data subsets:

- a) Substructure model definition
- b) Boundary-Condition change-data
- c) Highest-Level substructure identification
- d) Load case selection data

Printout of interact data may be requested from the INTERACT Postprocessor as described in section 230.1.

The stiffness data sets (sec. 152.0) and the corresponding boundary-condition (sec. 106.0), loads (sec. 134.0) and mass (sec. 138.0) data sets, as required, must be input prior to the interact data set in the input stream.

130.1 INPUT DATA

Record 1 Data Set Identification

BEGIN INTERACT DATA

This record initiates execution of the INTERACT Preprocessor.

130.1.1 Substructure Model Definition

Substructure-model components are defined via Records 2-8 for subsequent interaction analyses. Each lowest-level substructure is defined as a stiffness or stiffness/mass SET and a corresponding boundary condition STAGE. Higher-level substructures are defined by interaction of these lowest-level substructures. A maximum of 999 substructures (lowest-level and higher-level) may be identified per job.

Record 2 Substructure Interact-Tree Node Numbers (optional)

UNIQUE NODE NUMBERS

This record denotes that unique user-assigned numbers are used for all nodes in the total structure that is defined by the interact tree. In this case, these numbers are used to reference nodes in all higher-level substructures.

Default Record: Unique, user-assigned node numbers are not assumed. Unique numbers are assigned by the program to all nodes in the higher-level substructures. These numbers may be obtained from printout of the interact data (see sec. 230.1).

Record 3 Lowest-Level Substructure Definition

DEFINE SS n AS SET Se STAGE St

n = Substructure number (integer) in the range 1 to 999, inclusive, assigned to a lowest-level substructure of an interact tree.

Se = Integer identifying a SET which is to be equivalenced to "n."

St = Integer identifying a boundary-condition STAGE number previously defined for "Se" (see sec. 106.0).

This record is repeated to define all lowest-level substructures that are referenced by Record 6 to define higher-level substructures. Each lowest-level substructure must be assigned a different number via Record 3.

Higher-level substructures are identified and defined by Records 4 and 6. The numbers assigned via Records 3 and 4 to each of the substructures of an interact tree must be different.

Record 4 Higher-Level Substructure Definition

SS n

n = Substructure number (integer) in the range 1 to 999, inclusive.

The substructure identified by this record is referenced as the index substructure in the descriptions of subsequent records.

The data defined by Records 5 through 8 are attributed to the index substructure identified by Record 4.

Record 5 Distance Tolerance Between Nodes that Interact (optional)

A node in one component substructure is assumed to interact with a node in a different component substructure only if the nodes have the same analysis frame and the distance between the nodes is less than or equal to the tolerance specified by this record.

TOLERANCE tol

tol = The distance tolerance ($tol \geq 0.0$) to be used by the program when comparing the locations of nodes in different component substructures to identify interact nodes.

Default Record: The tolerance specified by the last input record of this type or 10^{-8} units.

Record 6 Interacted (Component) Substructures

INTERACT Sub1 Sub2 ...

Sub1, Sub2,... = A list of 1 to 28 numbers (integers) of component substructures that interact to form the index substructure. Each of the numbers in this list must have been defined previously by a Record 3 or a Record 4. The order in which "Subi" are input is used only in establishing the default sequence of retained freedoms for the index substructure as described in the introduction of this section.

Record 7 Suppression of Automatic Retention of Interacted Freedoms (optional)

NO GEOMETRIC RETAINS

This record denotes that the boundary condition data as defined previously for the substructures identified by Record 6 are to be used in forming the index substructure.

Default Record: Interact nodes are identified by comparing the coordinates and analysis frames of all nodes associated with the substructures identified by Record 6. The freedom activity labels of the FREE kinematic freedoms of an interact node in one substructure are automatically set to RETAIN (see sec. 106.1) if they correspond to FREE or RETAIN freedoms in another substructure.

If unique node numbers are used (Record 2), those nodes identified by the same number, the same analysis frame and the same coordinates (to within the tolerance specified by Record 5) are assumed to interact. For this case, if two nodes are identified by the same number and the same coordinates but have different analysis frames, an error message is issued.

If Record 2 is not input, nodes in different substructures that have the same analysis frame and the same

coordinates (to within the tolerance specified by Record 5) are assumed to interact. If more than one node in a substructure has the same analysis frame and the same coordinates as a node in another substructure, only one of the nodes is interacted.

Record 8 Node Reorder Specification (optional)

This record offers alternate ways of controlling the bandwidth of the gross stiffness matrix by effecting the internal node-number system of the index substructure (see sec. 146.1.1). Only the nodes that have one or more freedoms retained for the component substructures are included in the nodal data for the index substructure.

REORDER FROM	$\left\{ \begin{array}{l} x \ y \ z \\ \text{Node IN Sub} \end{array} \right\}$
--------------	---

x, y, z = GLOBAL X, Y and Z-coordinates of a point. Nodes of the index substructure are assigned internal numbers by the program according to increasing distances from this point.

$\left. \begin{array}{l} \text{Node} \\ \text{Sub} \end{array} \right\}$ = The user-assigned number (integer) of a retained "Node" associated with the lowest-level substructure identified by the integer "Sub" which is a component of the index substructure. Nodes of the index substructure are assigned internal numbers by the program according to increasing distances from "Node."

Default Record: The index-substructure node closest to the origin of the GLOBAL frame is assigned internal number 1. The remaining nodes of the index substructure are assigned internal numbers according to increasing distances from this node.

Records 4-8 may be repeated to define additional higher-level substructures.

130.1.2 Boundary-Condition Change-Data Subset (optional)

This data subset (Records 9-16) allows the user to modify the default kinematic-freedom activities or the default ordered sequence of retained freedoms for any substructure of an interact tree. Reference should be made to the introduction of section 130.0 for descriptions of the default data. If "BC change data" are not required, Records 9-16 should not be input.

BC changes specified for a substructure are effected after its nodal data and default freedom-activities are established. The nodes of an index substructure are those that have been retained in its components. The default activity for all the kinematic freedoms in the index substructure is RETAIN except for the freedoms which interact in forming this substructure. The default activity for these freedoms is FREE.

Specific ordering of the retained degrees-of-freedom for a substructure is particularly useful to order the equations (the reduced structural matrices) to be solved for dynamic analyses.

Record 9 Begin BC Change-Data

BEGIN BC CHANGES

Record 10 Substructure Identification

This record identifies a substructure for which BC change-data are input via subsequent Records 11-16. BC change-data for substructure components must be input prior to change-data for the corresponding index substructure.

SS Sub

Sub = Integer number of a substructure previously defined by a Record 3 or a Record 4.

Record 11 Lowest-Level Substructure Reference (optional)

The BC change-data defined via Record 12 and Record 15 require direct reference to node numbers of "Sub" identified by Record 10. These numbers are user-assigned node numbers only if "Sub" is a lowest-level substructure or if Record 2 is input. Except for these two cases, Record 11 must be input. Otherwise, the node numbers of "Sub" are the default, program-assigned numbers which are not directly known by the user.

This record allows the user to reference nodes of "Sub" in terms of the user node numbers of any of the lowest-level substructures (Rsub) that interact to form "Sub."

REFERENCE SS Rsub

Rsub = Integer number of any one of the lowest-level substructure components of the substructure identified by Record 10.

Default Record: It is assumed that the "Sub" identified by Record 10 is a lowest-level substructure or that Record 2 is used.

Record 12 BC Change-Data Specifications (optional)

This record, Variations 1 and/or 2, may be repeated to define all freedom-activity changes for selected nodes of the substructure identified by Record 10.

<u>Variation 1</u>	Activity	Dlist ALL	FOR ATlist
--------------------	----------	--------------	------------

<u>Variation 2</u>	Activity	Dlist ALL SYMM ASYM	IN SURFACE $\left\{ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} \right\}$ <THROUGH Nn>
--------------------	----------	------------------------------	--

These record formats are the same as those described for BC data (see sec. 106.1). The labels (key-words) specified for "Activity" and "Dlist" must be selected from those labels defined by the BC input data for the reference substructure (Rsub). If the node numbers are unique (Record 2 is input), the "labels" specified by this record for higher-level substructures must be the system-default labels (see sec. 106.1).

Default Record: The default freedom activities as described in the introduction of section 130.0.

Records 11 and 12 are repeated to specify additional freedom-activity changes.

Records 13-15 are input only to specify the order of the freedoms retained for the substructure identified by Record 10.

C-3

Record 13 Retained-Freedom Order Indicator (optional)

RETAIN ORDER

This record denotes that the retained freedoms are to be ordered as specified by Record 15.

Record 14 Lowest-Level Substructure Reference

This record is required only when Record 13 is input and when the user-assigned node numbers in the total model are not unique.

REFERENCE SS Rsub

The function of this record is the same as Record 11.

Record 15 Retained Freedom Order (optional)

{ Dlist }
{ ALL } FOR ATlist

{ Dlist } = List of 1 to 6 kinematic labels
{ ALL } as defined by the BC input data for the reference substructure (Record 11). If the node numbers are unique (Record 2 is input), the system-default labels (see sec. 106.1) must be used. If the word ALL is selected for this item, all six nodal freedoms are implied.

ATlist = ATLAS list of user node numbers (integers).

The retained freedoms are ordered in the sequence specified by Dlist and ATlist. The order is defined as all freedoms in Dlist for the first node in ATlist, followed by the same Dlist freedoms for the second node in ATlist, etc. If ALL is selected, the order of the nodal freedoms 1 through 6 will be as described in section 106.0. Reference to non-retained nodes via ATlist is not allowed.

This record, or Recc.'s 14 and 15 may be repeated. All retained freedoms for the substructure, including the net results of any freedom-activity modifications specified by Record 12, must be identified.

Default Records: If Records 13-15 are not input, the retained freedoms are ordered in the sequence as described in the introduction of section 130.0.

Record 16 End BC Change Data

END BC CHANGES

Records 4-16 may be repeated to define additional data for an interact tree.

130.1.3 Highest-Level Substructure Identification

The highest-level substructure of an interact tree is identified by the following record. If dynamic analyses are to be performed on a substructure model, this record is also used to define an equivalent SET number for the highest-level substructure. This SET number is used subsequently in the parameter lists of the "EXECUTE MASS," "EXECUTE VIBRATION" and "EXECUTE INTERPOLATION" Control Program statements. The boundary condition stage number automatically assigned to this SET is "STAGE=1."

Record 17 Highest-Level Substructure Identifications

DEFINE HIGHEST SS Sub <AS SET n>

Sub = Integer number of a higher-level substructure previously-defined by Record 4.

n = A stiffness or stiffness/mass data set number (integer) in the range 1 to 36, inclusive. This SET number must be different from the SET numbers of the lowest-level substructures. This item is not required if only stress analyses are to be performed on the substructure model.

Default: No SET/STAGE identifiers are assigned to the highest-level substructure.

130.1.4 Load Case Selection Data Subset (optional)

This data subset (Records 18-21) is only applicable to stress analyses of a substructure model. If these data are not input for this case, all load cases defined by the loads data (see sec. 134.0 and 234.0) for all the lowest-level substructures

(SET/STAGE components) are used in all the substructure-analysis back-substitution solutions. If, however, the interact solutions for only selected load cases are to be performed on selected substructures, this data subset must be input.

Record 18 Begin Load Case Selection

BEGIN LOADCASE SELECTION

Record 19 Substructure Identification

This record identifies the substructure for which back-substitution solutions are to be performed for the load cases selected by Record 20.

SS Sub

Sub = Integer number of a substructure previously-defined by a Record 3 or a Record 4.

Record 20 Load Case Selection

Case1 Case2 ...

Case1,Case2,... A list of selected load case labels associated with "Sub" identified by Record 19 for which interact solutions are to be performed. If "Sub" is not the highest-level substructure, the specified load cases must be associated with the index substructure of "Sub."

The interact solution data for the selected load cases are used for the back-substitution solution of the substructure identified by Record 19. Additionally, interact solutions for the substructure components of this substructure are effected only for the selected load cases. If no load cases are selected for a substructure, the interact solution for this substructure will include all load cases associated with the substructure immediately above it in the interact tree.

Records 19 and 20 may be repeated.

INTERACT

Record 21 End Load Case Selection

END LOADCASE SELECTION

This record denotes that all load case selections for a substructure-model stress analysis have been defined.

Record 22 End Interact Data Set

END INTERACT DATA

Additional interact data sets to define different interact trees may be input by repeating Records 1-22.

Table 130-1. Summary of Interact Data Records

Reference Page	Data Records
130.6	BEGIN INTERACT DATA
130.6	UNIQUE NODE NUMBERS
130.6	DEFINE SS n AS SET Se STAGE St
130.7	SS n
130.7	TOLERANCE tol
130.8	INTERACT Sub1 Sub2 . . .
130.8	NO GEOMETRIC RETAINS
130.9	REORDER FROM { x y z Node IN Sub }
130.10	BEGIN BC CHANGES
130.10	SS Sub
130.11	REFERENCE SS Rsub
130.11	Activity { Dlist ALL } FOR ATlist
130.11	Activity { Dlist ALL SYMM ASYM } IN SURFACE { 1 2 3 } <THROUGH Mn>
130.12	RETAIN ORDER
130.12	REFERENCE SS Rsub
130.12	{ Dlist ALL } FOR ATlist
130.12	END BC CHANGES
130.13	DEFINE HIGHEST SS Sub <AS SET n>
130.13	BEGIN LOADCASE SELECTION
130.14	SS Sub
130.14	Case1 Case2 . . .
130.14	END LOADCASE SELECTION
130.14	END INTERACT DATA

134.0 LOADS DATA

Static and quasi-static loads to be applied to a structural model are defined via this data set. Previous input of the associated stiffness and BC data sets is required. Additionally, if BEAM element thermal loads are specified, previous input of the Detail data set (sec. 114.0) is required if cross section "concepts" are used.

The loads data set is divided into the following data subsets:

- a) Discrete nodal loads
- b) Specified support displacements
- c) Distributed element loads
- d) Nodal thermal loads
- e) Element thermal loads
- f) Rotational inertia loads

Discrete nodal loads may be specified relative to any previously-defined nodal data reference frame (sec. 140.0). Distributed element loads, however, may be defined relative to the GLOBAL frame or the element reference frames (see appendix B). Support displacements must be specified relative to the analysis frames of the corresponding nodes. Specified support displacements may be input partially or wholly within the BC data set and/or within the loads data set. Rotational inertia loads are defined by rigid-body motion of the model about an axis of rotation fixed in the GLOBAL reference frame. All these data are subsequently processed by the LOADS Processor (sec. 234.0).

Each group of loads is identified by a unique load case label. The maximum number of load cases that may be defined and processed in a particular job is limited only by the available computer core storage at execution time. All loads associated with a particular node or element are cumulated and all loads associated with a particular load case are cumulated except thermal loads. For this case, only the last specified nodal-temperature increment is effective. If both nodal and thermal loads are specified for a particular point in the model, only the element thermal loads will be used.

Printout of the cumulated nodal and element-distributed applied forces, nodal and element temperature increments and the specified rotational kinematics may be requested per the discussion in section 234.0. Printout of specified displacements may be requested from the STRESS Postprocessor (sec. 254.2).

PRECEDING PAGE BLANK NOT FILMED

134.1 INPUT DATA

The following notation is used in this section.

- ATlist(n) = ATLAS list of node numbers (integers). Non-existent node numbers generated by ATlist(n) are not allowed.
- ATlist(e) = ATLAS list of stiffness element numbers (integers). Non-existent element numbers generated by ATlist(e) are not allowed.
- ATlist(c) = ATLAS list defining load case labels (positive integers or alphanumeric words with 1 to 7 characters). The word ALL, the 2-character words Fi, where i = 1, 6 and any of the kinematic and force labels selected via the BC data (sec. 106.1) may not be used as labels. Multiple load cases associated with a particular stiffness set and BC stage must be identified by unique labels.

Some data records are described by one or more of the foregoing ATLAS lists in addition to other ATLAS lists defined in this section. The total number of items identified via ATlist generation options plus the number of other items input by a single record may not exceed 1000. The general restrictions that a record may contain a maximum of 250 input items or a maximum of 125 items if the *+N record generation option is used also apply (see sec. 100.2).

Record 1 Data Set Identification

BEGIN LOADS DATA

This record initiates execution of the LOADS Preprocessor.

A loads data set may be comprised of one or more subsets or it may be comprised of only Records 1 through 4 and 47. That is, Records 2-4 may be used to assign load case titles (Record 3) and/or combine previously-defined load cases (Record 4). Load cases comprised of specified displacements that are defined wholly by the BC data may be handled in this manner.

Record 2 Set/Stage Identification (optional)

<SET	Se>	<STAGE	St>
------	-----	--------	-----

Se = Integer identifying a stiffness SET to which the subsequent input loads apply.
Default: The SET number specified by the last record of this type or SET 1.

St = Integer identifying a BC STAGE associated with set "Se" to which the subsequent input loads apply.
Default Items: STAGE 1

Default Record: SET 1 STAGE 1

This record may be repeated to identify a SET/STAGE anywhere within this data set except within one of the data subsets.

Record 3 Load Case Title (optional)

Each load case is identified by a user-assigned label. An alphanumeric title may also be assigned to load cases via this record. Both identifiers are displayed when printout is requested.

LOAD CASE ID	label	*#	text	#
--------------	-------	----	------	---

label = Load case label--a positive integer or an alphanumeric word of 1 to 7 characters. The word ALL, the 2-character words Fi, where i = 1, 6 and any of the kinematic freedom and force labels selected via the BC data may not be used. All labels associated with the specified SET/STAGE (Record 2) must be unique.

text = Alphanumeric text of 1 to 100 characters, including blanks, to be associated with "label."

Default Record: Load cases are identified only by the labels specified by the load-case definition records.

This record may be repeated anywhere except within any one of the data subsets.

Record 4 Load Case Combination Record (optional)

This record may be used to generate a load case as a linear combination of one or more other load cases associated with the SET/STAGE specified by Record 2.

CASE label COMBINE f1 Label1 f2 Label2 ...

label = Same as defined for Record 3. This load case label must be different from the labels denoted by Labeli in this record.

fi,Labeli = Loads associated with Labeli are to be multiplied by "fi." One to 10 pairs of these items may be input. If more than 10 load cases are to be combined, this record must be repeated as described below. The Labeli loads need not be input prior to this record. The factor "fi" is applied to all loads associated with Labeli. Note that a factor applied to equivalent nodal loads caused by thermal effects is not, in general, linearly related to the nodal temperature increments (see sec. 134.1.4).

This record may be repeated anywhere except within any one of the data subsets. The "label" defined by one record of this type may be the same as the "label" specified by a subsequent record of this type. In this case, all loads associated with the particular load case are cumulated. A "label" defined by this record, however, may not appear as a Labeli in subsequent records of this type.

134.1.1 Nodal Load Data Subset (optional)

All concentrated nodal forces and/or moments are defined via this data subset. Specified lists of nodal loads may be associated with a group of load cases (Records 6, 7 and 8), a group of Structural nodes (Records 6, 9 and 10) or a group of nodal-force labels (Records 11 and 12).

Record 5 Begin Nodal Load Subset

BEGIN NODAL LOAD DATA

Record 6 Load Reference Frame and Freedom Order (optional)

This record establishes the reference frame relative to which subsequent nodal loads are defined. Any nodal data reference frame may be used for loads input. This record also defines the nodal-force correspondence list associated with lists of load values input via "flist" in Records 8 and/or 10.

Label GLOBAL ANALYSIS	< ORDER Nf1 Nf2 . . . >
-----------------------------	----------------------------------

Label GLOBAL ANALYSIS

 = Alphanumeric "Label" of a nodal data reference frame, the key-word GLOBAL which denotes the GLOBAL frame or the key-word ANALYSIS. The key-word ANALYSIS denotes that loads are defined with respect to the analysis frame of the loaded nodes. Analysis frames of the loaded nodes may be different.
 Default: The reference frame denoted by the last record of this type or ANALYSIS.

Nf1, Nf2, ... = List of 1 to 6, 2-character, alphanumeric identifiers defining the order and number of loads input via "flist" in Records 8 and/or 10. Only the nodal-force identifiers associated with non-zero loads need be included in this list. The i-th identifier Nfi must be one of the following:

Nfi -- Either one of the nodal-force labels selected via the BC data for a specific type of reference frame (REC, CYL or SPH) or one of the words Fj; j = 1, 6 which denote the six ordered nodal forces associated with all reference frame types (see Record 4 of BC data). If the former type of identifier is specified, all loaded nodes must have the same list of nodal-force labels. However, if "Fj" words are used, the nodal-force labels associated with the loaded nodes are immaterial.

Default: The nodal force order defined by the last record of this type or Fj; j=1, 6.

Default Record: The order and number of nodal-forces defined by the last record of this type or Fj; j=1,6. If this record is not used, the order in which nodal loads are specified via Records 8 and/or 10 must be consistent with the default nodal-force order (e.g., FX, FY, FZ, MX, MY and MZ for REC reference frames).

Records of this type may appear any number of times within this data subset.

134.1.1.1 Loads Grouped by Load Cases

Records 7 and 8 allow specified nodal loads to be associated with a group of load cases.

Record 7 Load Case Labels

CASE ATlist (c)

Nodal loads specified via subsequent Records 8 are attributed to each of the load cases defined by ATlist (c).

Record 8 Node and Load Specification

ATlist (n) { flist
Nf1 f1 Nf2 f2 ... }

All loads specified via this record are applied to all nodes in ATlist (n).

flist = List of 1 to 6 nodal loads associated with the sequence and number of nodal-force labels identified by the last Record 6. Trailing zeros need not be input.

Nfi,fi = Nfi is a nodal-force label and "fi" is the corresponding load value. The format of "Nfi" is the same as defined for Record 6. These items are input in pairs in an order which is independent of Record 6.

This record or Records 7 and 8 may be repeated.

Example Use of Records 7 and 8:

For load cases A9C and B20, nodes 35 and 100 through 110 have an FX force of 5. and an MZ of 10. Nodes 36 and 200 through 210 have an FX of 10. and an MZ of 20. Additionally, nodes

37 and 300 through 310 have an FX of 15. and an MZ of 30. The data records could be input as:

```
CASE  ABC  B20  /
35   100  TO   110    FX  5.  MZ  10.  /
**2  1    100  0   100  0   5.  0   10.  /
```

134.1.1.2 Loads Grouped by Nodes

Records 9 and 10 allow specified nodal loads to be associated with a group of Structural nodes.

Record 9 Node List

```
NODE  ATlist (n)
```

Subsequent Records 10 define nodal loads to be applied to all nodes in ATlist(n).

Record 10 Load Case and Nodal Load Specification

```
ATlist (c)  { flist
              { Nf1 f1 Nf2 f2 ... } }
```

All loads specified via this record are attributed to each of the load cases defined by ATlist(c).

```
flist+ }
Nfi    }   = Same as defined for Record 8
fi     }
```

This record or Records 9 and 10 may be repeated.

Example Use of Records 9 and 10:

Nodes 5, 7, 9, 11 and 45 for load cases 1, 2, 3, H1, H2 and H3 have an FY force equal to 6 times the integer part of the load case label and an MX value equal to twice the integer part of the load case label. The data records could be input as:

```
NODE  5  TO  11  BY  2  45  /
1  H1  FY  6.  MX  2.  /
**2  1  1  0  6.  0  2.  /
```

134.1.1.3 Loads Grouped by Nodal Freedoms

Records 11 and 12 allow specified nodal loads to be associated with a group of freedoms.

Record 11 Nodal Freedom List

FREEDOM ATlist(fn)

ATlist(fn) = ATLAS list of alphanumeric words--each word is comprised of a nodal-force label, as specified via the BC data, followed by a node number. An ATlist(fn) could be MY11 TO MY88 which denotes moments about the Y-axis at nodes 11 through 88. Non-existent nodal-force labels and node numbers referenced by words generated by ATLAS(fn) are not allowed.

Record 12 Load Case and Nodal Load Specification

ATlist(c) flist

flist = List of nodal loads to be applied to the freedoms as listed by the last Record 11. The number and sequence of the loads in "flist" must correspond with the number and sequence of freedoms specified by Record 11.

This record or Records 11 and 12 may be repeated.

Example Use of Records 11 and 12:

FX loads are to be specified at nodes 1 through 5 and 20. For load case 10, the forces are 1, 2, 13, 4, -5 and 8. For load case TAXI, the forces are 7, 12, 15, -2, 6 and 2. The data records could be input as:

FREEDOM	FX1	TO	FX5	FX20	/
10	1.	2.	13.	4.	-5. 8. /
TAXI	7.	12.	15.	-2.	6. 2. /

Record 13 End Nodal Load Subset

END NODAL LOAD DATA

Additional Records 2-13 may be input to define nodal loads for different SET/STAGE combinations.

134.1.2 Support Displacement Data Subset (optional)

This data subset, Records 14-22, may be used to input specified displacements. These data may be input partially or wholly via the BC data set (sec. 106.1.1) and/or the loads data set. Displacements input for a particular SET and STAGE (Record 2) are cumulated. Multiple displacements specified for the same freedom are cumulated.

Displacements may be specified only for freedoms which have been previously SUPPORTED. All displacements are defined and effected relative to the analysis frame of the associated node. If no displacement is specified for a SUPPORTED freedom, it is rigidly fixed (zero displacement is enforced).

Specified lists of displacements may be associated with a group of load cases (Records 15, 16 and 17), a group of nodes (Records 15, 18 and 19) or a group of kinematic freedoms (Records 20 and 21). Rotational displacements are input in radians.

Record 14 Specified Displacement Data Subset

```
BEGIN SUPPORT DISPLACEMENT DATA
```

Record 15 Nodal-Freedom Order (optional)

This record defines the nodal-freedom correspondence list associated with lists of displacements input via Records 17 and/or 19.

```
CFDER Nf1 Nf2 . . .
```

Nf1,Nf2,... = List of 1 to 6, 2-character, alphanumeric identifiers defining the order and number of kinematic freedoms associated with the displacements input via "dlist" in Records 17 and/or 19. The i-th identifier Nfi in this list must be one of the following:

Nfi -- Either one of the kinematic labels selected via the BC data for a specific type of analysis frame (REC, CYL or SPH) or one of the words Fj; j = 1, 6 which denote the six ordered nodal freedoms (see Record 4 of BC data) associated with all analysis frame types. If the former type of identifier is used, all nodes with specified displacements must have the same list

of analysis-frame kinematic labels. However, if "Fj" words are used, the kinematic labels associated with the referenced nodes are immaterial.

Default Record: The order and number of nodal-freedoms defined by the last record of this type or Fj; j = 1, 6. If this record is not used, the order in which the 6 displacements are specified via Records 17 and/or 19 must be consistent with the default nodal-freedom order (e.g., TX, TY, TZ, RX, RY and RZ for REC analysis frames).

Records of this type may appear any number of times within this data subset.

134.1.2.1 Displacements Grouped by Load Cases

Records 16 and 17 allow specified displacements to be associated with a group of load cases.

Record 16 Load Case Labels

CASE ATlist (c)

Displacements specified via subsequent Records 17 are attributed to each of the load cases defined by ATlist (c).

Record 17 Node and Displacement Specification

ATlist (n) { dlist
Nf1 d1 Nf2 d2 . . . }

All displacements specified by this record are applied to all nodes in ATlist (n).

dlist = List of 1 to 6 displacements associated with the sequence and number of kinematic freedoms identified by the last Record 15. Trailing zeros need not be input.

Nfi, di = The i-th kinematic label and displacement in this record denoting which freedom is to have a displacement of "di" imposed. The format of Nfi is the same as defined for Record 15. These items are input in pairs in an order which is independent of Record 15.

This record or Records 16 and 17 may be repeated.

Use of Records 16 and 17 is illustrated by the example shown for Records 7 and 8 provided that displacements and kinematic labels are used instead of the force terminology.

134.1.2.2 Displacements Grouped by Nodes

Records 18 and 19 allow specified displacements to be associated with a group of nodes.

Record 18 Node List

```
NCDE ATlist(n)
```

Subsequent Records 19 define displacements to be applied to all nodes in ATlist(n).

Record 19 Load Case and Displacement Specification

```
ATlist(c) { dlist
            Nf1 d1 Nf2 d2 ... }
```

All displacements specified within this record are attributed to each of the load cases defined by ATlist(c).

```
dlist }
Nfi   } = Same as defined for Record 17
di    }
```

This record or Records 18 and 19 may be repeated.

Use of Records 18 and 19 is illustrated by the example shown for Records 9 and 10 provided that displacements and kinematic labels are used instead of the force terminology.

134.1.2.3 Displacements Grouped by Nodal Freedoms

Records 20 and 21 allow specified displacements to be associated with a group of kinematic freedoms.

Record 20 Kinematic Freedom List

```
FREEDOM ATlist(fn)
```

ATlist(fn) = ATLAS list of alphanumeric words--each word is comprised of a kinematic label, as specified via the BC data, followed by a node number. An ATlist(fn) could be TX11 TO TX88 which denotes the X-axis

translational freedoms for nodes 11 through 88. Non-existent node numbers referenced by ATlist(fn) are not allowed.

Record 21 Load Case and Displacement Specification

ATlist(c) dlist

dlist = List of displacements to be imposed on the freedoms as listed by the last Record 20. The number and sequence of the displacements in "dlist" must correspond with the number and sequence of freedoms specified by Record 20.

This record or Records 20 and 21 may be repeated.

Use of Records 20 and 21 is illustrated by the example shown for Records 11 and 12 provided that displacements and kinematic labels are used instead of the force terminology.

Record 22 End Support Displacement Data Subset

END SUPPORT DISPLACEMENT DATA

Additional Records 2-22 may be input to define loads data for different SET/STAGE combinations.

134.1.3 Element Load Data Subset (optional)

This data subset, Records 23-29, is used to specify distributed loading on selected stiffness elements. Uniform or linearly-varying loads on all element types except the SCALAR element may be defined. Distributed loads are defined via load intensities at element nodes.

Element loads may be associated with a group of load cases (Records 25 and 26) or a group of elements (Records 27 and 28). Loads applied to the one and 2-dimensional elements are

prescribed in sections 134.1.3.1 and 134.1.3.2, whereas loads applied to BRICK elements are prescribed according to section 134.1.3.3.

Record 23 Begin Element Load Subset

BEGIN ELEMENT LOAD DATA

Record 24 Load Direction Specification

DIRECTION	⟨GLOBAL ELEMENT⟩	{ Nn x y z }
-----------	---------------------	-----------------

⟨GLOBAL
ELEMENT⟩ = Key-word denoting that either the GLOBAL or each of the ELEMENT reference frames is used to define the load direction vector. The direction of the distributed loads is from the origin of this frame to the point defined by "Nn" or "x", "y" and "z".
Default: The load reference frame is either the one denoted by the last record of this type or the GLOBAL frame.

Nn = Node number (integer).

x,y,z = Coordinates x, y and z of a point located with respect to either the GLOBAL frame or each of the ELEMENT reference frames of the loaded elements.

This record may appear any number of times anywhere within this data subset. The load direction defined via this record remains effective until redefined.

134.1.3.1 Loads Grouped by Load Cases

Records 25 and 26 allow element loads to be associated with a group of load cases.

Record 25 Load Case Labels

CASE ATlist(c)

Element loads specified by subsequent Records 26 are attributed to each of the load cases defined by ATlist(c).

Record 26 Element and Load Specification

ATlist(e)	{ flist blist }
-----------	--------------------

The distributed loads specified by this record are applied to each element in ATlist(e).

flist = List of load intensities at the element nodes. The number and input sequence of items within "flist" that are required for each element type except the BRICK are presented by table 134-1.

blist = Load intensities for BRICK elements only;
see section 123.1.3.3.

This record or Records 25 and 26 may be repeated.

Example Use of Records 25 and 26:

For load cases LIMIT and M30, elements 100 through 200 have a pressure load of 10, elements 300 through 400 have a pressure load of 12, and elements 500 through 600 have a pressure load of 14. The data records could be input as:

```
CASE LIMIT M30 /  
100 TO 200 10. /  
*+2 200 0 200 2. /
```

134.1.3.2 Loads Grouped by Elements

Records 27 and 28 allow element loads to be associated with a group of elements.

Record 27 Element List

ELEMENT ATlist(e)

Subsequent Records 28 define load intensities for each element in ATlist(e).

Record 28 Load Case and Load Specification

ATlist(c) { flist blist }

The distributed loads specified by this record are attributed to each of the load cases defined by ATlist(c).

{ flist
 blist } = Same as defined for Record 26

This record or Records 27 and 28 may be repeated.

Example Use of Records 27 and 28:

Elements 5, 10 and 15 have pressure loads of 6 for load cases P1 and H2; 8 for load cases P2 and H4; and 10 for load cases P3 and H6. The data records could be input as:

```
ELEMENT 5 TO 15 BY 5 /  
P1 H2 6. /  
*+2 1 2 2. /
```

Table 134-1. Input Variations of Nodal Load Intensities ^①

ELEMENT TYPE	LOADED ELEMENT NODE	NUMBER OF INPUT NODAL LOAD INTENSITIES							
		1	2	3	4	5	6	7	8
ROD	N1	1	1						
	N2	1	2						
BEAM	A1	1	1						
	A2	1	2						
SPAR	N1	1	1						
	N2	1	2						
COVER & CCOVER	PIU	.5(1)	1	.5(1)	.5(1)		1		1
	P2U	.5(1)	1	.5(2)	.5(2)		2		2
	P3U	.5(1)	1	.5(3)	.5(3)		3		3
	P4U(quad.)	.5(1)	1		.5(4)				4
	PI L	.5(1)	2	.5(1)	.5(1)		4		5
	P2 L	.5(1)	2	.5(2)	.5(2)		5		6
	P3 L	.5(1)	2	.5(3)	.5(3)		6		7
	P4 L(quad.)	.5(1)	2		.5(4)				8
PLATE & CPLATE	A1	1		1	1				
	A2	1		2	2				
	A3	1		3	3				
	A4(quad.)	1			4				
Triangular GPLATE	A1	1		1					
	A2	1		2					
	A3	1		3					
Quadrilateral GPLATE	A1	1			1				
	A2	1			2				
	A3	1			3				
	A4	1			4				
BRICK	See Section 134.1.3.3								
SROD	N1	1	1						
	N2	1	2						
4-Node SPLATE	N1	1			1				
	N2	1			2				
	N3	1			3				
	N4	1			4				
8-Node SPLATE	A1	1			1				
	A2	1			2				
	A3	1			3				
	A4	1			4				

① Each nonzero integer in this table denotes which one of the nodal load intensities input by Record 26 or 28 is to be applied to a corresponding element node as shown in the table. For the case when 1, 3, or 4 load intensities are input for a COVER or CCOVER, one-half of the input values are applied at the appropriate upper and lower plate corner nodes denoted by PIU and PI L, respectively. N(i) and A(i) denote structural and auxiliary element nodes, respectively, as described in appendix B.

134.1.3.3 Loads on BRICK Elements

Records 24-28 may be used to define distributed loads on BRICK elements via "blist" in Records 26 and 28. "blist" is comprised of one element face number followed by one or more nodal load intensities. A distributed load on one of the six faces of a BRICK element is defined via load intensities at the nodes of the loaded face. Coplanar faces may not be loaded. Coplanar element faces occur, for example, when a BRICK element is degenerated to describe a wedge.

Three variations of "blist" in Record 26 and 28 are presented below. The direction of the distributed pressure loading, as specified by the last Record 24 in the input stream, must be defined relative to the ELEMENT reference frame.

Variation 1 Uniform Pressure on a Face

blist = Face p1
where,

Face = Face number (integer) of the elements.
See appendix B for the definition of BRICK faces.

p1 = Load intensity on the face.

Variation 2 Linear Pressure Distribution on a Face

blist = Face p1 p2 p3 p4
where,

Face = Same as for Variation 1.

p1-p4 = Load intensities at the face corner nodes Ni. The required input order of these values is shown in the table below. Load intensities at the intermediate edge nodes are generated by linear interpolation.

Face Number	Corner Node Load Intensities				Intermediate Edge Node Load Intensities			
	p1	p2	p3	p4	pe1	pe2	pe3	pe4
1	N1	N2	N3	N4	NE1	NE2	NE3	NE4
2	N5	N6	N7	N8	NE5	NE6	NE7	NE8
3	N2	N3	N6	N7	NE2	NE6	NE10	NE11
4	N1	N4	N5	N8	NE4	NE8	NE9	NE12
5	N3	N4	N7	N8	NE3	NE7	NE11	NE12
6	N1	N2	N5	N6	NE1	NE5	NE9	NE10

The nodes defining faces 5 and 6 in the foregoing table are compatible with a BRICK element defined according to the figure in appendix B. If the nodes N1 to N4 are input in the opposite order (clockwise when viewed in the negative x-direction), face numbers 5 and 6 as shown in the table are interchanged.

Variation 3 General Pressure Distribution on a Face

blist = Face p1 p2 p3 p4 pe1 pe2 pe3 pe4
where,

Face = Same as for Variation 1

p1-p4 = Load intensities at the face corner nodes Ni. The required input order of these values is shown in the foregoing table.

pei = List of load intensities for the "nonzero" intermediate nodes on the edge NEi. The load intensities denoted by "pei" are associated with a node order corresponding with traversing of element edge E(i) in its positive direction (see appendix B). The sequence of these lists is shown in the foregoing table. Face numbers 5 and 6 as shown in this table are interchanged if the condition discussed for Variation 2 exists.

Record 29 End Element Load Subset

END ELEMENT LOAD DATA

Additional Records 2-29 may be input to define nodal and element loads for different SET/STAGE combinations.

134.1.4 Nodal Thermal Load Data Subset (optional)

This data subset, Records 30-35, defines incremental nodal temperatures for thermal loading of stiffness elements. Calculation of equivalent nodal loads and initial stresses are performed by the LOADS Processor (sec. 234.0) using the following input data:

- a) The initial (base) element temperature T_0 defined by "Tcode" in the stiffness element data (sec. 152.0) and the final element temperature defined as $T_1 = T_0 + \Delta T$.
- b) Equivalent nodal loads are calculated for the applied thermal strain defined as $\epsilon(T_1) - \epsilon(T_0)$, using $\epsilon(T)$ values associated with the element material "Mcode" specified in the stiffness element data (sec. 152.0).
- c) Initial stresses induced in the elements by the thermal strain are based on the initial T_0 temperature.

Incremental nodal temperatures may be associated with a group of load cases (Records 31 and 32) or a group of nodes (Records 33 and 34).

In this data subset, ATlist(n) may identify nodal subset names "Nxxx" as defined previously via the SUBSET-DEFINITION Preprocessor (sec. 156.0) in addition to node numbers. Non-existent nodes or node subsets identified by ATlist(n) are not allowed.

Record 30 Begin Nodal Thermal Load Subset

BEGIN NODAL THERMAL LOAD DATA

134.1.4.1 Loads Grouped by Load Cases

Records 31 and 32 allow nodal temperature increments to be associated with a group of load cases.

Record 31 Load Case Labels

CASE ATlist(c)

Subsequent Records 32 define temperature increments attributed to each of the load cases specified by ATlist(c).

Record 32 Node and Temperature-Increment Specification

ATlist (n) Δt

Δt = Temperature increment in degrees Fahrenheit which is effective at all nodes in ATlist(n). The last temperature increment specified for a node takes precedence.

This record or Records 31 and 32 may be repeated.

134.1.4.2 Loads Grouped by Nodes

Records 33 and 34 allow temperature increments to be associated with a group of nodes.

Record 33 Node List

NODE ATlist (n)

Subsequent Records 34 define temperature increments for all nodes in ATlist(n).

Record 34 Load Case and Temperature Increment Specification

ATlist (c) Δt

The temperature increment specified by this record is attributed to each of the load cases defined by ATlist(c).

Δt = Same as defined for Record 32

This record or Records 33 and 34 may be repeated.

Record 35 End Nodal Thermal Load Subset

END NODAL THERMAL LOAD DATA

Additional Records 2-35 may be input to define loads for different SET/STAGE combinations.

134.1.5 Element Thermal Load Data Subset (optional)

This data subset, Records 36-41, defines incremental element temperatures for thermal loading of stiffness elements. Calculation of equivalent nodal loads and initial stresses are performed by the LOADS Processor (sec. 234.0) as described in section 134.1.4.

Incremental element temperatures may be associated with a group of load cases (Records 37 and 38) or a group of stiffness elements (Records 39 and 40).

In this data subset, ATlist(e) may identify stiffness element subset names "Exxx" as defined previously via the SUBSET-DEFINITION Preprocessor (sec. 156.0) in addition to element numbers. Non-existent elements or element subsets identified by ATlist(e) are not allowed.

Record 36 Begin Element Thermal Load Subset

BEGIN ELEMENT THERMAL LOAD DATA

134.1.5.1 Loads Grouped by Load Cases

Records 37 and 38 allow element temperature increments to be associated with a group of load cases.

Record 37 Load Case Labels

CASE ATlist(c)

Subsequent Records 38 define temperature increments attributed to each of the load cases specified by ATlist(c).

Record 38 Element and Temperature-Increment Specification

ATlist(e) ATlist(Δt)

ATlist(Δt) = ATLAS list of temperature increments in degrees Fahrenheit at the element nodes. The number and input sequence of items in ATlist(Δt) that are required for each element type identified by ATlist(e) are presented in table 134-2.

This record or Records 37 and 38 may be repeated.

134.1.5.2 Loads Grouped by Elements

Records 39 and 40 allow element temperature increments to be associated with a group of elements.

Record 39 Element List

ELEMENT ATlist(e)

Subsequent Records 40 define temperature increments for all elements in ATlist(e).

Record 40 Load Case and Temperature-Increment Specification

Atlist(c) ATlist(Δt)

The temperature increments specified by this record are attributed to each of the load cases defined by ATlist(c).

ATlist(Δt) = Same as defined for Record 38

This record or Records 39 and 40 may be repeated.

Record 41 End Element Thermal Load Subset

END ELEMENT THERMAL LOAD DATA

Additional Records 2-41 may be input to define loads for different SET/STAGE combinations.

134.1.6 Rotational Inertia Load Data Subset (optional)

This data subset, Records 42-46, defines rigid-body motion of the model about an axis of rotation fixed in the GLOEAL reference frame. These kinematics are used by the LOADS Processor (sec. 234.0) to calculate equivalent nodal forces for the reversed inertia loads ($F=-Ma$). Inertia loads are generated from the mass associated with the stiffness elements (sec. 152.0), the non-structural mass elements and concentrated masses (sec. 138.0). The following rules apply to the use of this data subset per SET/STAGE combination defined by Record 2:

- a) A maximum of 4095 different axes of rotation may be defined (Record 43). Multiple load cases (rotational kinematics) may be associated with each axis via Records 44 and 45.
- b) A load case identified by Record 44 may be associated with only one axis of rotation. Several load cases input by this data subset may be combined via Record 4, however, to create a load case associated with multiple rotation axes.

Record 42 Begin Inertia Load Subset

BEGIN INERTIA LOAD DATA

Record 43 Axis of Rotation Definition

AXIS Nn1 Nn2

Nn1, Nn2 = Use node numbers (integers). The instantaneous axis of rotation is directed from "Nn1" through "Nn2." The sense of the angular velocity and acceleration components

input by Record 45 is defined relative to this axis according to the right-hand rule.

The axis defined by this record remains effective for load cases input via Records 44 and 45 until another Record 43 is encountered.

Record 44 Load Case Labels

CASE ATlist(c)

Inertial loads caused by the rotational kinematics described by Record 45 are attributed to each of the load cases defined by ATlist(c).

Record 45 Rotational Kinematics

omega <alpha>

omega = Angular velocity

alpha = Angular acceleration
Default: Zero acceleration

Records 44 and 45 may be repeated.

Different axes of rotation and load cases for a SET/STAGE combination may be defined by repeating Records 43-45.

Record 46 End Inertia Load Subset

END INERTIA LOAD DATA

Additional Records 2-46 may be input to define inertial loads for different SET/STAGE combinations.

Record 47 End Data Set

END LOADS DATA

Loads to be applied to different SETS and STAGES may be input via additional Records 1-47.

Table 134-2. Input Variations of Element Nodal-Temperature Increments ①

ELEMENT TYPE	LOCATION OF Δt INCREMENT	NO. OF INPUT TEMPERATURE INCREMENTS														COMMENTS			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14				
ROD	N1 N2	1 1	1 2																
BEAM	N1 N2 A1(+y) A1(-y) A1(+z) A1(-z) A2(+y) A2(-y) A2(+z) A2(-z)	1 1 1 1 1 1 1 1 1 1	1 2 1 1 1 1 2 2 2 2	1 1 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	1 2 3 3 3 3 4 4 4 4	0 1 2 3 4 5 6 7 8 9 10	
SPAR	N1(+y) N1(-y) N2(+y) N2(-y) N3	1 1 1 1 1	1 1 2 2 2	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3	1 1 2 2 3		$\triangleright \frac{(1)+(2)}{2}$ $\triangleright \frac{(1)+(2)+(3)+(4)}{4}$
COVER	P1U P2U P3U P4U(quad) P1L P2L P3L P4L(quad)	1 1 1 1 1 1 1 1	1 1 2 2 1 1 2 2	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 2 3 4 5 6 7 8	
PLATE	N1 N2 N3 N4(quad) A1 A2 A3 A4(quad)	1 1 1 1 1 1 1 1	1 1 2 2 1 1 2 2	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 2 3 4 5 6 7 8	
GPLATE	N1 N2 N3 N4(quad) A1(+z) A1(-z) A2(+z) A2(-z) A3(+z) A3(-z) A4(+z)(quad) A4(-z)(quad) A5(+z)(quad) A5(-z)(quad)	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1 1 2 2 1 1 2 2 2 2 2 2 2 2 2	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 1 2 2 3 3 3 3 3 3 4 4 4 4 4	1 2 3 4 5 6 7 8 9 10 11 12 13 14	$\triangleright \frac{(1) + (2)}{2}$ $\triangleright \frac{(3) + (4)}{2}$ $\triangleright \frac{(5) + (6)}{2}$ $\triangleright \frac{(7) + (8)}{2}$
BRICK	ALL NONZERO NODES																		The i-th temperature increment is applied to the i-th non-zero node.
SROD	N1 N2	1 1	1 2																
SPLATE	ALL NODES	1																	
CPLATE	N1 N2 N3 N4(quad) A1 A2 A3 A4(quad)	1 1 1 1 1 1 1 1	1 1 2 2 1 1 2 2	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 2 3 4 5 6 7 8	Additional Options A. Surface Δt's. If 2 increments are input, the first applies to the +z surface and the second applies to the -z surface of the element. If offset, the average of these increments is applied at the structural nodes N(i). B. A Δt for each lamina at each node. The number of input values is 3N (3N+3 if offset) for triangular plates and 4N (4N+4 if offset) for quad plates. The number of laminas, N, must be ≥ 3. The sequence of input values is: Δt at the 3 structural nodes (4 if a quad) followed by the 3 (or 4) corner Δt's for lamina 1, then the 3 (or 4) corner Δt's for lamina 2, etc.
CCOVER	P1U P2U P3U P4U(quad) P1L P2L P3L P4L(quad)	1 1 1 1 1 1 1 1	1 1 2 2 1 1 2 2	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 1 2 2 3 3 3 3	1 2 3 4 5 6 7 8	Additional Options A. Surface Δt's at each corner. The sequence of input values is: Δt at P1U(+z), Δt at P1U(-z), Δt at P2U(+z), etc. (12 values for a triangular CCOVER and 16 values for a quad CCOVER.) B. A Δt at each corner of each lamina. The number of input values is 3N for a triangular CCOVER and 4N for a quad CCOVER. The total number of laminas, N, must be ≥ 3. The sequence of input values is: the 3 (or 4) corner Δt's for lamina 1, then the 3 (or 4) corner Δt's for lamina 2, etc. Laminas 1 through N are those laminas defined for the upper plate followed by those laminas for the lower plate.

① Each nonzero integer in this table denotes which one of the temperature increments input by Record 38 or 39 is to be applied to a corresponding element as shown in the table. A zero integer denotes that the temperature increment at the corresponding point is to be set to zero. Upper and lower plate corners of COVER and CCOVER elements are denoted by P1U and P1L, respectively. N(i) and A(i) denote structural and auxiliary element nodes (see Appendix B). Additionally, A1(-y), for example, denotes the -y face of the element at node A1.

Table 134-3. Summary of Loads Data Records

Reference Page	Data Records
134.2	BEGIN LOADS DATA
134.3	< SET Se > < STAGE St >
134.3	LOAD CASE ID label * / text /
134.4	CASE label COMBINE f1 Label1 f2 Label2 ...
134.4	BEGIN NODAL LOAD DATA
134.5	$\left\langle \begin{array}{l} \text{Label} \\ \text{GLOBAL} \\ \text{ANALYSIS} \end{array} \right\rangle \langle \text{ORDER } n_{f1} \ n_{f2} \ \dots \rangle$
134.6	CASE ATlist(c)
134.6	ATlist(n) $\left\{ \begin{array}{l} \text{flist} \\ n_{f1} \ f1 \ n_{f2} \ f2 \ \dots \end{array} \right\}$
134.7	NODE ATlist(n)
134.7	ATlist(c) $\left\{ \begin{array}{l} \text{flist} \\ n_{f1} \ f1 \ n_{f2} \ f2 \ \dots \end{array} \right\}$
134.8	FREEDOM ATlist(fn)
134.8	ATlist(c) flist
134.8	END NODAL LOAD DATA
134.9	BEGIN SUPPORT DISPLACEMENT DATA
134.9	ORDER n _{f1} n _{f2} ...
134.10	CASE ATlist(c)
134.10	ATlist(n) $\left\{ \begin{array}{l} \text{dlist} \\ n_{f1} \ d1 \ n_{f2} \ d2 \ \dots \end{array} \right\}$
134.11	NODE ATlist(n)
134.11	ATlist(c) $\left\{ \begin{array}{l} \text{dlist} \\ n_{f1} \ d1 \ n_{f2} \ d2 \ \dots \end{array} \right\}$
134.11	FREEDOM ATlist(fn)
134.12	ATlist(c) dlist
134.12	END SUPPORT DISPLACEMENT DATA
134.12	BEGIN ELEMENT LOAD DATA
134.13	DIRECTION $\left\langle \begin{array}{l} \text{GLOBAL} \\ \text{ELEMENT} \end{array} \right\rangle \left\{ \begin{array}{l} n_n \\ x \ y \ z \end{array} \right\}$
134.13	CASE ATlist(c)
134.13	ATlist(e) $\left\{ \begin{array}{l} \text{flist} \\ \text{blist} \end{array} \right\}$
134.14	ELEMENT ATlist(e)
134.14	ATlist(c) $\left\{ \begin{array}{l} \text{flist} \\ \text{blist} \end{array} \right\}$
134.17	END ELEMENT LOAD DATA
134.18	BEGIN NODAL THERMAL LOAD DATA
134.18	CASE ATlist(c)
134.18	ATlist(n) Δt
134.19	NODE ATlist(n)
134.19	ATlist(c) Δt
134.19	END NODAL THERMAL LOAD DATA
134.19	BEGIN ELEMENT THERMAL LOAD DATA
134.20	CASE ATlist(c)
134.20	ATlist(e) ATlist(Δt)
134.20	ELEMENT ATlist(e)
134.20	ATlist(c) ATlist(Δt)
134.21	END ELEMENT THERMAL LOAD DATA
134.21	BEGIN INERTIA LOAD DATA
134.21	AXIS n _{n1} n _{n2}
134.22	CASE ATlist(c)
134.22	omega < alpha >
134.22	END INERTIA LOAD DATA
134.22	END LOADS DATA

136.0 MACHBOX AERODYNAMICS DATA

Unsteady aerodynamic analyses of rigid or elastic planar or non-coplanar wings and wing/tail configurations in supersonic flow can be performed by the MACHBOX Processor (sec. 236.0). Calculations performed by this processor are based on the three-dimensional Mach Box technique published in reference 136-1. The input data set supporting MACHBOX aerodynamic analyses is comprised of the following data subsets:

- a) Geometry -- Aerodynamic surface planform and Mach Box grid data; airfoil thickness-slope pressure correction data and off-planform wash sampling locations.
- b) Modal -- Vibration modes or "AIC" mode substitutes (see sec. 232.0) associated with the aerodynamic surfaces.

Both data subsets are required. This data set need not be preceded by any other data set within a data deck.

Each set of aerodynamics data is referenced as a CASE. A maximum of 36 MACHBOX data cases may be identified per job. Printout of the aerodynamic case data may be requested from the MACHBOX Postprocessor (sec. 236.1).

PRECEDING PAGE BLANK NOT FILMED

136.1 INPUT DATA

Record 1 Data Set Identification

BEGIN MACHBOX DATA

This record initiates execution of the MACHBOX Preprocessor.

Record 2 Data Case Number (optional)

Each set of MACHBOX aerodynamics data is referenced as a case which is identified by an integer number. If multiple sets are used in a job, each case must be assigned a unique number.

CASE n

n = Data case number (integer) in the range 1 to 36, inclusive.

Default Record: CASE 1

Record 3 Data Case Title (optional)

A title provides additional user-identification of a data case. This title and the case number are printed within the page headings of MACHBOX printout (see sec. 236.0).

LABEL label

label = Alphanumeric text of 1 to 75 characters including embedded blanks.

Default Record: MACHBOX printout is identified only by the data case number.

136.1.1 Geometry Data Subset

This data subset includes the following geometrical information:

- a) Aerodynamic surface planform definitions
- b) Mach Box grid data
- c) Airfoil thickness-slope aerodynamic correction data
- d) Locations at which off-planform sampling of the perturbation velocity is desired.

An aerodynamic model may be comprised of one or two surfaces whose planforms are defined relative to local, right-handed, rectangular coordinate systems x-y-z oriented with respect to the GLOBAL reference frame. The origin of each surface reference-frame triad x-y-z must lie in the GLOBAL X-Z plane. Additionally, these triads must be oriented such that the positive x-axis is in the aft direction and the positive y-axis is directed from the root to the tip of the surface(s). The positive X-axis must be in the aft (streamwise) direction and the X-Z plane must be the plane of wing symmetry.

Geometric configurations accommodated by the MACHBOX aerodynamic analysis may be summarized as follows:

- a) Single surface with dihedral in the range $\pm 45^\circ$
- b) Two coplanar surfaces
- c) Two surfaces with or without longitudinal and vertical separations each having dihedral in the range $\pm 45^\circ$
- d) One surface with dihedral in the range $\pm 45^\circ$ and a second non-intersecting surface with a dihedral of 90° to model a symmetrical wing with an aft vertical tail.

Except for a vertical tail, each surface is considered to be symmetrical about the GLOBAL X-Z plane. Geometrical data are input only for half of the symmetrical aerodynamic surfaces. Each surface planform is defined via straight line segments and may have a streamwise or a raked tip. Two surfaces may physically overlap, however, the aerodynamic Mach cone of the aft surface should not intersect the forward planform. Aerodynamic interference effects between wing and tail planforms are accommodated in the following manner:

- a) For coplanar surfaces, the interference of the wing on the tail and the effect of the tail on the wing are both included in the analysis.
- b) For non-coplanar surfaces, only the influence of the wing on the tail is included in the analysis.

The numerical solution technique employed by the MACHBOX Processor overlays the aerodynamic surfaces and diaphragms with a uniform grid of rectangular Mach Boxes whose diagonals lie along the Mach rays (see ref. 136-1). Each box is identified by a span number and a chord number as illustrated by figure 136-1. The size of the boxes as defined by the user must be

sufficiently small so that the assumption of constant source strength on each box is valid. Centers of the boxes define the locations of the aerodynamic control points.

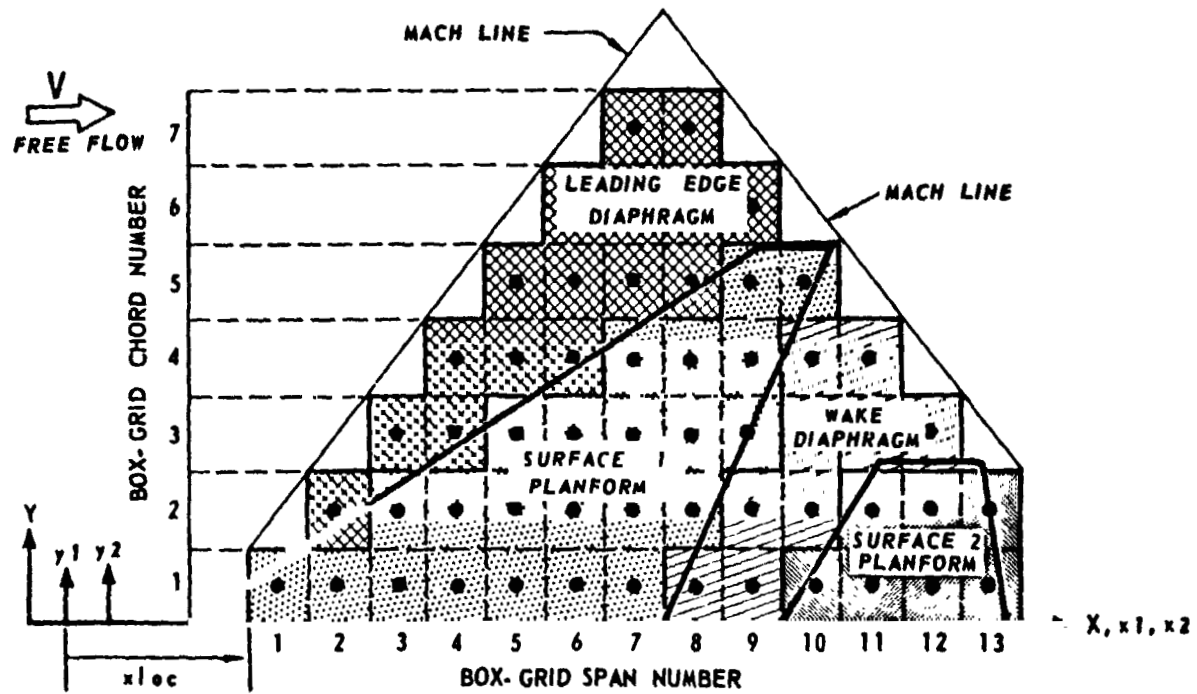


Figure 136-1. Mach Box Aerodynamic Grid

Record 4 Begin Geometry Subset

BEGIN GEOMETRY DATA

Records 5-9 define an aerodynamic surface planform and associated optional pressure correction data. These records are repeated if two surfaces are required.

Record 5 Surface Orientation

SURFACE Num x z dih

- Num = $\begin{Bmatrix} 1 \\ 2 \end{Bmatrix}$ = Integer identification number of the surface. If only one surface is defined, it must be identified as Surface 1. If two surfaces are required, the downstream surface must be identified as Surface 2.
- x,z = GLOBAL X and Z coordinates defining the origin of the local x-y-z coordinate system for surface "Num."
- dih = Dihedral angle in degrees which positions the surface relative to the GLOBAL X-Y plane. A surface in the GLOBAL positive-Y half-space has a positive dihedral if it is located by a positive right-hand rotation about the X-axis. The value of "dih" must always be in the range $\pm 45.0^\circ$ except a Surface 2 may have a dihedral of 90.0° .

Trailing zeros need not be input.

Records 6 and 7 define the leading and trailing edges, respectively, of surface "Num." Each edge is defined by coordinates of 2 to 10 points located with respect to the surface reference frame specified by Record 5. These points must be input in a root-to-tip order. Each edge must begin at $y=0.0$ and the tip-most points of the two edges must have the same local y-coordinate. Segments of an edge may be parallel to the local x-axis, however, the Mach Box grid boundaries, defined via Record 10, must match streamwise planform breaks of this type.

Record 6 Leading Edge Coordinates

LEADING EDGE x1 y1 x2 y2 ...

x_i, y_i = The x and y local coordinates of the i-th point defining the leading edge. A leading edge may not have forward swept segments.

Record 7 Trailing Edge Coordinates

TRAILING EDGE x1 y1 x2 y2 ...

x_i, y_i = The x and y local coordinates of the i-th point defining the trailing edge. A trailing edge may have forward and backward swept segments.

Airfoil thickness effects on interference calculations are ignored by the basic analysis technique. However, correction factors to account for these effects may be applied to the calculated aerodynamic pressure distribution via the data input by Records 8 and 9. These corrections are effected locally at Mach Box centers by employing second-order piston theory (ref. 136-1).

Record 8 Pressure Correction Indicator (optional)

Records 8 and 9 are input only if thickness correction factors for surface "Num" are to be applied.

THICKNESS mach

mach = Free stream Mach number greater than 1.0 and ≤ 5.0 . The correction factors are applied by the MACHBOX Processor (sec. 236.0) only when the Mach number specified in the EXECUTE MACHBOX parameter list for the associated CASE number is within $\pm 10^{-5}$ of "mach."

Record 9 Airfoil Thickness-Slope Correction Data

c1 c2 ... cn

c_i = The thickness slope (variation of the airfoil half-thickness with respect to the x local axis) at the i-th box center on surface "Num." The number and sequence of items input by this

record must be the same as the number and sequence of boxes on surface "Num." The number of boxes on a surface is dependent on the Mach number specified by Record 8 and the Mach Box grid data specified by Record 10. The boxes on a surface are sequentially ordered beginning with the leading-edge root box and proceeding fore to aft on each chord through the aft box on the tip chord.

Records 5-9 are repeated to define a second aerodynamic surface with optional pressure correction data.

Record 10 Mach Box Grid Generation Data

BOX nchords	{ XCENTER XEDGE }	xloc
-------------	----------------------	------

nchords = Number of aerodynamic box-grid chords (integer) to be established on Surface 1. For the example illustrated by figure 136-1, "nchords" is 5. The dimension of each box in the x-direction is calculated as the square root of $(M^2 - 1)$, times the dimension of each box in the y-direction, where M is the free stream Mach number.

For a single surface or coplanar surfaces with zero dihedral and without box subdivision refinement (see SUBD parameter of EXECUTE MACHBOX statement), the box grid may extend up to 50 boxes in each direction. For any spatial configuration or off-planform sampling case (see Record 11), the box grid is limited to a maximum of 40 boxes in each direction. In all cases, the total number of on-planform boxes is limited to 1000 and the total number of planform plus diaphragm boxes is limited to 1275.

{ XCENTER XEDGE }

= Key-word denoting whether "xloc" defines the center (XCENTER) or the leading edge (XEDGE) of any box within the Mach Box grid. The complete box grid for one or two surfaces is automatically

generated relative to the location of this box.

xloc = x-coordinate relative to the local reference frame of Surface 1 which locates a box center or a spanwise box edge. This coordinate may be anywhere near or on Surface 1 (see fig. 136-1 which illustrates "xloc" for an XEDGE).

Record 11 Off-Planform Sampling Locations (optional)

This record allows the user to sample the perturbation velocity in the flow-field resulting from surface motions. Samples of this type may be requested only for a single surface. Velocity components are calculated by the MACHBOX Processor and sampled components may be printed via the MACHBOX Postprocessor (see sec. 236.0).

```
SAMPLING CHORD c1 FBOX f1 LBOX l1 ZLOC z1
          CHORD c2 FBOX f2 LBOX l2 ZLOC z2
          ...
```

ci = Integer greater than zero and ≤ 40 identifying the i-th box-grid chord in which sampling is to be done. A maximum of 10 chords may be specified.

fi } = Box-grid span numbers (integers) identifying the first (most forward) and the
li } last (aft-most) boxes of a sequence of boxes on chord "ci." These items may be in the range $1 \leq fi \leq li \leq 40$ for any sampling chord.

zi = The GLOBAL Z-coordinate at which off-planform velocity sampling is to be effected relative to the foregoing sequence of boxes within chord "ci." The Z-coordinate may not lie in the surface.

Default Record: No off-planform washes are calculated.

Record 12 End Geometry Subset

```
END GEOMETRY DATA
```

This record indicates all geometry data have been defined for the case identified by Record 2.

136.1.2 Modal Data Subset (optional)

This subset defines the vibration modes, "AIC" mode substitutes or rigid-body modes to be associated with each aerodynamic surface. If no modal data are input, only the Mach box grid would be generated by execution of the MACHBOX Processor.

Flow disturbances are modeled by distributed pulsating sources whose strengths are related to modal displacement amplitudes at the box centers. The aerodynamic theory requires translational components of the vibration modes to be normal to the surface planform(s). Thus, the analysis frame axis associated with the retained nodal freedoms must be normal to the aerodynamic surface.

Record 13 Begin Modal Subset

BEGIN MODAL DATA

Record 14 defines the vibration modes or "AIC" mode substitutes that are to be used in the analysis of the case identified by Record 2. Only one of the two variations of this record may be used per data case. Variation 1 must be used when mode shape coefficients generated by the INTERPOLATION Processor are to be employed in the aerodynamic analysis. Mode shapes may be defined directly via Variation 2 for a rigid-body aerodynamic analysis.

Record 14 Modal Data

Variation 1 USE Name WITH { SURFACE 1 }
 { SURFACE 2 }
 BOTH }

Name = The name of an interpolation-coefficient matrix previously generated via the INTERPOLATION Processor (see sec. 232.0).

{ SURFACE 1 }
 { SURFACE 2 }
 BOTH } = Key-word denoting whether "Name" is to be associated with "SURFACE 1," "SURFACE 2" or "BOTH" aerodynamic surfaces. A particular coefficient matrix may be used with BOTH surfaces only if the same analysis frame is associated with all the retained nodes on both surfaces. Thus, the two surfaces must have the same dihedral.

This record must be repeated if different interpolation coefficient matrices are to be associated with two surfaces.

Variation 2 allows the user to define 1 to 6 rigid body modal freedoms relative to the GLOBAL reference frame. This variation may be used only if the dihedral angles of the surfaces are zero.

Variation 2 FIGID BODY <x y z> Rbf1 mag1 Rbf2 mag2 ...

x,y,z = GLOBAL X, Y and Z coordinates defining the reference point to be used in generating the rigid body modes. Default: The origin of the GLOBAL frame is used as the reference point.

Rbfi = Key-word denoting the i-th rigid body freedom selected from the list TX, TY, TZ, FX, RY and RZ. Within each of these key-words, the letter "T" is associated with a translation freedom and "R" is associated with a rotation freedom relative to the GLOBAL X-Y-Z triad.

magi = The relative magnitude of the i-th rigid body freedom with respect to the reference point. Items "Rbfi" and "magi" are input in pairs 1 to 6 times in an order which defines the sequence of rigid body modes used in the aerodynamic analysis.

Record 15 End Modal Subset

END MODAL DATA

Additional cases may be defined by repeating Records 2-15 with a unique identification number assigned to each case by Record 2.

Record 16 End MACHBOX Data Set

END MACHBOX DATA

Additional cases may be input by repeating Records 1-16.

Table 136-1. Summary of MACHBOX Data Records

Reference Page	Data Records
136.2	BEGIN MACHBOX DATA
136.2	CASE n
136.2	LABEL label
136.5	BEGIN GEOMETRY DATA
136.5	<u>SURFACE</u> Num x z dih
136.6	<u>LEADING EDGE</u> x1 y1 x2 y2 ...
136.6	<u>TRAILING EDGE</u> x1 y1 x2 y2 ...
136.6	<u>THICKNESS</u> mach
136.6	c1 c2 ...
136.7	BOX nchords { <u>XCENTER</u> } xloc <u>XEDGE</u>
136.8	<u>SAMPLING CHORD</u> c1 FBOX f1 LBOX l1 ZLOC z1 CHORD c2 ...
136.8	END GEOMETRY DATA
136.9	BEGIN MODAL DATA
136.9	USE Name WITH { <u>SURFACE 1</u> } <u>SURFACE 2</u> } BOTH
136.10	RIGID BODY <x y z> Rbf1 mag1 Rbf2 mag2 ...
136.10	END MODAL DATA
136.10	END MACHBOX DATA

138.0 MASS DATA

An ATLAS mass model may be comprised of any combination of the following components.

- a) Structural mass defined via the stiffness elements associated with a stiffness data set (sec. 152.0) that is identified by the same number assigned to the corresponding mass data set (sec. 138.1).
- b) Non-structural mass defined via mass finite elements (sec. 138.1.2).
- c) Fuel-distribution conditions (sec. 138.1.3). A maximum of 99 fuel distributions may be defined.
- d) Payload-distribution conditions. A maximum of 99 payload-distribution conditions may be defined (sec. 138.1.4).
- e) Concentrated-mass subsets defined in terms of mass input directly as concentrated masses at nodes. A maximum of 9 subsets may be defined (sec. 138.1.5).

Overall mass-distribution conditions for a model, as defined in section 138.1.1, are comprised of user-selected combinations of the foregoing mass model components. A condition must be defined to calculate reduced mass matrices associated with retained nodal freedoms (sec. 106.0) and to calculate panel-weight matrices. All geometrical and mass element data associated with weight panels and the foregoing items b) through e) are defined via the mass data set. Labeling and output format information required to assemble a detailed weight-statement are also defined by this data set.

The mass data set is divided into the following optional data subsets:

- a) Mass Distribution Condition
- b) Mass Element
- c) Fuel
- d) Payload
- e) Concentrated Mass
- f) Mass Lumping
- g) Auxiliary Weight Panel
- n) Weight-Statement Label
- i) Factor

PRECEDING PAGE BLANK NOT FILLED

A mass data set defines the non-structural mass associated with one or more structural models each identified by a SET number. The mass associated with a particular model may be defined partially or wholly by the mass data set and/or by the stiffness data set (sec. 152.0). All mass included in a mass-distribution condition for a particular SET is cumulated when a reduced mass matrix or panel-weight matrix is generated. The geometrical data associated with a SET are defined via the Nodal data set (sec. 146.0) identified by the same SET number. A maximum of 36 node/mass and node/stiffness/mass data SETS may be defined per job.

Printout of the mass data may be requested from the MASS Postprocessor (sec. 238.0). Plots of the mass-element data can also be effected via the EXTRACT and GRAPHICS Postprocessors as discussed in sections 218.0 and 228.0.

138.1 INPUT DATA

Record 1 Data Set Identification

BEGIN MASS DATA

This record initiates execution of the MASS Preprocessor.

Record 2 Data Set Number (optional)

Each set of mass data is identified by an integer number. If multiple sets are used in a job, each set must be assigned a unique number.

SET n

n = Data set number (integer) in the range 1 to 36, inclusive. If a mass data set is to be associated with a structural model, the corresponding stiffness and mass data sets must be identified by the same SET number.

Default Record: SET 1

138.1.1 Mass-Distribution Condition Data Subset (optional)

This subset, Records 3-5, defines which mass model components, a) through e) of section 138.0, are essential to an overall mass-distribution condition. A maximum of 100 different mass conditions may be associated with a mass data set. This data subset is required only if reduced mass matrices and/or panel-weight matrices are to be calculated.

Record 3 Begin Mass Condition Subset

BEGIN CONDITION DATA

Record 4 Mass Condition Data

{ STAGE St
PANEL DATA Sub } CONDITION cond Fuel Payload Conmass

St = The boundary condition (BC) stage number (integer) previously-defined by the BC data (sec. 106.0) for a nodal data set identified by the same number as this data set. The number and order of the retained kinematic freedoms associated with the BC stage correspond

with the size and row sequence of the mass matrix.

- Sub = An auxiliary panel subset number defined by Record 42. This number denotes a particular set of weight panels to be used in calculation of a panel-weight matrix.
- cond = Mass-distribution CONDITION number (integer) in the range 1 to 999, inclusive. Each CONDITION associated with a particular SET must be assigned a unique number. This number, right-adjusted and zero-filled, is denoted by xxx in the 7-character name of the generated mass or weight matrix MDCxxxY. Y is the display-code equivalent (1=A, 2=B, ...) of the mass set number. For example, a mass matrix for CONDITION 3 and set 5 (5=E) is named MDC003E. This 7-character name would be specified in the parameter list of an EXECUTE VIBRATION statement.
- Fuel = A fuel condition number (integer) defined by Record 20 to be associated with the mass CONDITION. A zero denotes no fuel distribution is to be included.
- Payload = A payload condition number (integer) defined by Record 34 to be associated with the mass CONDITION. A zero denotes no payload distribution is to be included.
- Conmass = A concentrated mass subset number (integer) defined by Record 36 to be associated with the mass CONDITION. A zero denotes no concentrated masses are to be included.

If Fuel and Payload or if Fuel, Payload and Conmass are all zero, they need not be input.

This record may be repeated to define a maximum of 100 mass/weight distribution CONDITIONS. All structural mass associated with a stiffness set (sec. 152.0) that is identified by the same number assigned to the mass set (Record 2) is automatically cumulated with all non-

structural mass defined by Records 6-8 in the analysis of a mass CONDITION. The user may, however, exclude the mass and/or stiffness elements from a CONDITION via Record 55 of the Factor data subset (sec. 138.1.9)

Record 5 End Mass Condition Subset

END CONDITION DATA

This record indicates all mass distribution CONDITIONS have been defined for the SET identified by Record 2.

138.1.2 Mass Element Data Subset (optional)

All elements in addition to those defined via the stiffness element data (sec. 152.0) required to describe the structural and non-structural mass of a model identified by a particular SET number are defined by this data subset. If additional non-structural mass element data are not required, Records 6-8 should not be input.

Record 6 Begin Mass Element Subset

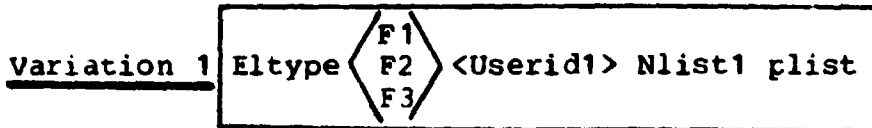
BEGIN MASS ELEMENT DATA

Element data may be defined by any combination of the three element-definition record variations presented below. These variations provide the following capabilities:

- 1) Explicit definition of an element;
- 2) Generation of a sequence of elements with node-definition lists and user element numbers incremented by one;
- 3) Generation of a sequence of elements with node-definition lists and user element numbers incremented as specified.

Records of this type are repeated as necessary. A maximum of 32,767 mass elements may be defined per SET. Detailed descriptions of the ATLAS mass element types are presented in appendix C. If OPTION=4 of the MASS Processor is used, all mass elements should be defined by Structural nodes which have non-zero stiffness components.

Record 7 Mass Element Data



Eltype = A key-word or its equivalent integer which denotes the type of element defined by this record. The available options for Eltype are:

<u>Key-word</u>	or	<u>Integer-Equivalent</u>
ROD		1
EEAM		2
SPAR		3
COVER		4
PLATE		5
SCALAR		9



= Key-word denoting whether the element material density (F1), total weight (F2), or c.g. (F3) is input by "plist." Only F1 properties may be input for SPAR elements and only F2 properties may be input for SCALAR elements. Default: The last specified property type or property type F1.

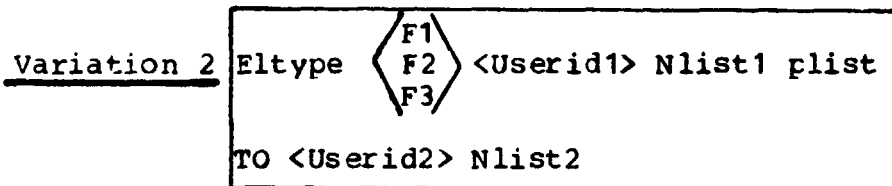
Userid1 = User-assigned element identification. A user element number or a word may be specified by this item. The options are:

- a) Number--The letter N followed by the user element number (integer in the range 1-32767). The element number is used when printing element data (sec. 238.0) and when generating mass element subsets via the SUBSET-DEFINITION Preprocessor (sec. 156.0).
- b) Word--Alphanumeric word of 1 to 10 characters to replace the "Eltype" identification in printout requested from the MASS Postprocessor (sec. 238.0).

Default: Mass elements are assigned the input sequence number of the element within this data subset and they are identified by "Etype."

- Nlist1 = List of node numbers (integers) which define the element. The required list of nodes for each of the element types is shown in table 138-1 and is presented in detail in appendix C.
- plist = List of element properties. The properties and variations of the number and order of input property values for each of the element types are summarized in table 138-1 and are presented in detail in appendix C.

Variations 2 and 3 allow the user to specify generation of a sequence of elements. All elements in the sequence will be of the same type and will have the same property values. The user element numbers and node numbers are generated by incrementing the first list until the second list is reached. Each of the items denoted by Userid1 and Nlist1, when incremented, must reach the corresponding items denoted by Userid2 and Nlist2 simultaneously. A maximum of 100 elements may be defined by one record of this type.



Items 1-5 are the same as defined for Variation 1.

- Userid2 = Number of the last element in the sequence defined in the same manner as option a) for Userid1. This item is required only if Userid1 is specified as N followed by an integer.
- Nlist2 = Node numbers (integers) defining the last element in the sequence. The number of items in Nlist1 and Nlist2 must be equal.

Variation 3

F1 F2 F3	Eltype <Userid1> Nlist1 plist
TO <Userid2> Nlist2 BY <Inc1> inc2	

Items 1-8 are the same as defined for Variation 2.

Inc1 = The letter N followed by a signed or unsigned integer (e.g., N5, N+10, N-85) which denotes the user element number increment. This item is required only if Userid1 and Userid2 are input as N followed by an integer.

inc2 = List of positive and/or negative node-number increments (integers) to be used with Nlist1. The number of items in Nlist1 and "inc2" must be equal.

Example: An example record to generate a set of ROD elements with input-sequence number identifiers is:

ROD F2 1 9 5.6 TO 7 6 BY 2 -1 /

This record is equivalent to the following four records:

ROD	F2	1	9	5.6	/
ROD	F2	3	8	5.6	/
ROD	F2	5	7	5.6	/
ROD	F2	7	6	5.6	/

If user element numbers 10 through 13 are desired, the single record would assume the form:

ROD F2 N10 1 9 5.6 TO N13 7 6 BY N1 2 -1 /

Record 8 End Mass Element Subset

END MASS ELEMENT DATA

This record indicates that all element data have been defined for the SET identified by Record 2.

Table 138-1. Input Variations of Mass Element Data ①

ELEMENT TYPE	NO. OF	F1-DENSITY	NUMBER OF INPUT PROPERTY VALUES							F2-WEIGHT	NUMBER OF INPUT PROPERTY VALUES									F3 - C.G.	NUMBER OF INPUT PROPERTY VALUES												
			PROPERTIES	1	2	3	4	5	6		7	8	9	PROPERTIES	1	2	3	4	5		6	PROPERTIES	1	2	3	4	5	6					
NAME	NO.	NODES																															
ROD	1	2	DENSITY A1 A2	1 1 2 2 2 3							WEIGHT A1 A2	1 1 1 * 2 2 * 2 3											WEIGHT NODE X		1 2 0 0 2 1								
BEAM	2	2 - 5	DENSITY A1 J1 IY1 IZ1 A2 J2 IY2 IZ2	1 1 2 2 * * * * * * * * * * * * 2 3 * * * * * * * * * * * *		1				1	WEIGHT A1 J1 IY1 IZ1 A2 J2 IY2 IZ2	1 1 1 * 2 2 * * * * * * * * * * * * * 2 3 * * * * * * * * * * * *		1				1							WEIGHT NODE X		1 2 0 0 2 1 1 1 1 1 1						
SPAR	3	2	t-web DENSITY A1-U A1-L A2-U A2-L	1 1 2 2 3 3 3 4 3 3 3 4		1			2																								
COVER	4	3 - 9	DENSITY-U DENSITY-L t-U t-L	1 1 1 1 1 2 2 2 3 2 3 4							WEIGHT-U WEIGHT-L t-U t-L	1 1 1 1 1 2 2 2 * * 3 3 * * 3 4											WEIGHT-U WEIGHT-L NODE-U X-U Y-U NODE-L X-L Y-L				1 1 1 2 2 2 3 3 0 0 0 3 0 0 4 4 0 5 0 4 5 0 5 0						
PLATE	5	3 - 9	DENSITY t	1 2							WEIGHT t	1 1 * 2										WEIGHT NODE X Y		1 1 2 0 0 2 0 3									
SCALAR	9	1									WEIGHT I=Label IXX IYY IZZ IXY IXZ IYZ	1 1 1 1 1 1 1 0 2 2 2 2 2 2 0 0 3 3 3 3 3 0 0 0 4 4 4 4 0 0 0 0 5 5 5 0 0 0 0 0 6 6 0 0 0 0 0 0 7																					

- ①
- Each integer tabulated under the "number of input property values" denotes which one of the listed property values input by a data Record 7 is to be assigned to the j-th element property listed in this table or whether the property value is to be set to zero.
 - An asterisk (*) denotes a property value is automatically set to 1.0.
 - Two asterisks (**) denote the property value is calculated using the other input property values.

138.1.3 Fuel Data Subset (optional)

The data subset, Records 9-21, defines the fuel tank configurations, the fuel-management sequences and the fuel-tank orientations. Selected combinations of these mass model components are used to define a maximum of 99 fuel mass-distribution conditions by Record 20 for the SET identified by Record 2. If fuel data are not required, Records 9-21 should not be input.

Record 9 Begin Fuel Subset

BEGIN FUEL DATA

138.1.3.1 Fuel Tank Data

The geometry of a fuel tank is, in general, defined by multiple tank sections. All sections used to define a fuel tank must have the same basic shape. This shape is identified as TYPE1 or TYPE2 as shown in figure 138-1.

The configuration and general data for a fuel tank are defined by Records 10-12. These records may be repeated to define a maximum of 500 fuel tanks.

Record 10 Tank Identification

TANK n

n = Fuel-tank, identification number (integer) in the range 1-32767, inclusive. If multiple tanks are required, each tank must be assigned a unique number.

Record 11 General Tank Data (optional)

⟨TYPE1⟩ <levels> <density> <useable PERCENT>
⟨TYPE2⟩

⟨TYPE1⟩ = Key-word denoting that either
⟨TYPE2⟩ TYPE1 or TYPE2 geometric shapes are defined by Record 12 to describe the tank configuration.
Default: TYPE1

levels = Number of equally-spaced fuel levels to be calculated for this tank. An integer in the range 2-63, inclusive.

Default: 10 fuel levels or the number of levels specified for this item in the immediately-preceding record of this type.

- density = Fuel weight density greater than zero for this tank.
 Default: 0.0294 lbs/in.³ or the density specified for this item in the immediately-preceding record of this type.
- useable = Percent of the tank volume that can be used for fuel (0.<useable≤100.)
 Default: 100.0 percent or the value specified for this item in the immediately-preceding record of this type.

Default Record: TYPE1 10 .0294 100. PERCENT /

Record 12 Tank Section-Definition Data

Each tank section is defined by a list of node numbers which establishes the geometries of a sequence of TYPE1 or TYPE2 shapes (fig. 138-1). Any combination of the three input-record variations presented below may be used to define a maximum of 63 sections per fuel tank.

Variation 1 <Id1> Nlist1

- Id1 = Alphanumeric word of the form Sxx where xx is an integer number in the range 1-63, inclusive. This word is used to identify the tank section described by this record.
 Default: Sxx where xx is the i-th, sequential section defined for a tank by a record of this type.
- Nlist1 = List of node numbers (integers) which define a section. 3 or 4 mid-surface nodes must be specified if TYPE1 shapes are used; 8 or 20 nodes must be specified if TYPE2 shapes are used.

Variations 2 and 3 allow the user to specify generation of a sequence of tank sections. The section identification numbers and the node numbers are generated by incrementing the first list until the second list is reached. Each of the items denoted by

Id1 and Nlist1, when incremented, must reach the corresponding items denoted by Id2 and Nlist2 simultaneously. A maximum of 63 sections may be defined by one record of this type.

Variation 2 `<Id1> Nlist1 TO <Id2> Nlist2`

Items 1 and 2 are the same as defined for Variation 1.

Id2 = Identifier for the last section in the sequence. This item is defined in the same manner as "Id1" and is required only if "Id1" is specified.

Nlist2 = Node numbers (integers) defining the last section in the sequence. The number of items in Nlist1 and Nlist2 must be equal.

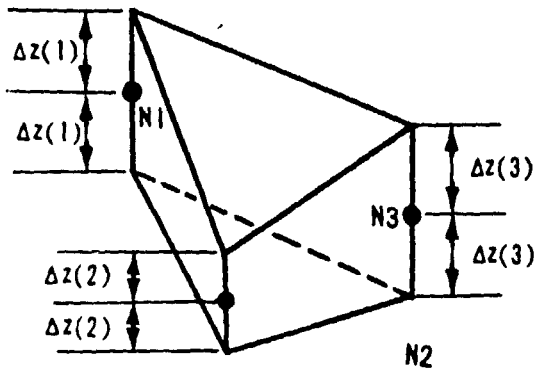
Variation 3 `<Id1> Nlist1 TO <Id2> Nlist2 BY
<Inc1> inc2`

Items 1-5 are the same as defined for Variation 2.

Inc1 = The letter S followed by a signed or unsigned integer (e.g., S5, S-20, S+12) which denotes the section, identification number increment. This item is required only if "Id1" and "Id2" are specified.

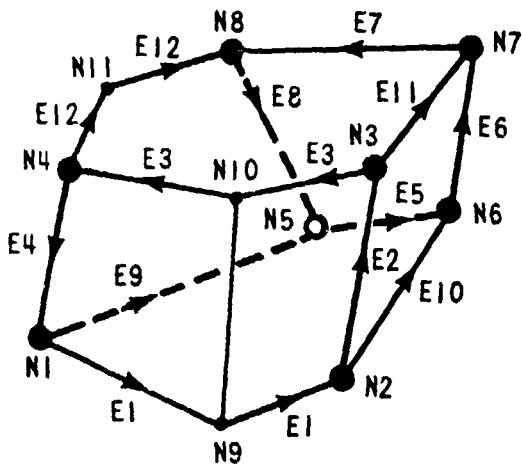
inc2 = List of positive and/or negative node-number increments (integers) to be used with Nlist1. The number of items in Nlist1 and "inc2" must be equal.

Records 10-12 may be repeated.



TYPE1 Shape

Three or four mid-surface nodes N1,N2,N3 (and N4) are used. Addition of the respective Δz coordinates to the nodal z-coordinates defines the upper surface, whereas subtraction defines the lower surface. The Δz directions are defined by the input nodal z-axes which need not be parallel. If 4 nodes are used, they must be input sequentially in either a clockwise or counterclockwise direction.



TYPE2 Shape

The basic solid is defined by 8 nodes. These nodes define 12 edges (E1-E12) and directions thereof as shown in the figure. An intermediate node may be specified on each edge. The 8 unique corner nodes must be input first followed by any edge nodes. The first 8 nodes define two opposite sides of the solid. N1-N4 may be input clockwise or counterclockwise, however, N5-N8 must be ordered so that each of the node pairs (N1,N5), (N2,N6), (N3,N7) and (N4,N8) lies on a separate edge. The 1 to 12 edge nodes must be input in an order consistent with the edge numbers. If only some of the edges have nodes, zeros must be input in the appropriate locations so that a total of 20 integers are input. The nodal input for the example shown would be N1,N2,...,N8,N9,0,N10,0,0,0,0,0,0,0,0,N11.

Figure 138-1. TYPE1 and TYPE2 Geometric Shapes

138.1.3.2 Fuel Management Data

Management data which define how fuel is to be loaded and/or used to establish a desired fuel condition are input via Records 13-16. These records may be repeated to define a maximum of 99 fuel-management sequences.

Record 13 Management Sequence Identification

MANAGEMENT SEQUENCE n

n = Fuel-management, sequence identification number (integer) in the range 1-99, inclusive. If multiple sequences are required, each sequence must be assigned a unique number.

The sequence in which fuel is loaded, used and/or transferred is defined by the input order of Records 14-16.

Record 14 Fuel Loading Data

LCAD TANK T1 <T2> <T3> <RATIO r1 r2 r3> UNTIL List

T1,T2,T3 = Fuel-tank, identification numbers (integers) previously-defined by Record 10. One to three tanks may be loaded via this data record.

ri = The relative rate at which the i-th tank identified by "Ti" is to be loaded. If, for example, r1=1.0 and r2=2.0, the tank identified by T2 will be loaded twice as fast as the tank identified by T1. The number of specified rates must equal the number of tanks identified by this record. If rates are specified, at least two tanks must be identified by this record.
Default: r1=r2=r3=1.0

List = List of items which describe the end condition of the fuel-loading steps defined by this record. One of the following three variations of "List" may be used.

Variation 1: Tn = One of the fuel-tank, identification numbers identified by "T1, T2 or T3"

in this record. Fuel loading will stop when tank "Tn" is full.

Variation 2: wt = Weight of fuel to be loaded into the tanks identified by "T1, T2 and T3."

Variation 3: wt TOTAL = Fuel loading will stop when the total weight of fuel in all tanks associated with the mass SET reaches "wt."

This record may be repeated.

Record 15 Fuel Usage Data

USE TANK T1 <T2> <T3> <RATIO r1 r2 r3> UNTIL List

Items 1-10 are the same as defined for Record 14.

List = List of items which describe the end condition of the fuel usage defined by this record. One of the following four variations of "List" may be used.

Variation 1: wt = Weight of fuel to be used from the tanks identified by "T1, T2 and T3."

Variation 2: wt TOTAL = Fuel usage will stop when the total weight of fuel remaining in all tanks associated with the mass SET equals "wt."

Variation 3: Ti <Tj> <Tk> wt = Fuel will be used until the total weight of fuel remaining in the tanks identified by "Ti, Tj and Tk" equals "wt." At least one of these tank identification numbers (integers) must be the same as T1, T2 or T3 specified by this record.

Variation 4: <fi> Ti <fj> Tj > EQUALS <fk> Tk <fl> Tl >

= Fuel will be used from the tanks identified by "Ti" and "Tj" until the following equation is satisfied.
 $fi*WTi + fj*WTj = fk*WTK + fl*WTl$
The weight factors for tanks "Ti" and "Tj" are specified via "fi" where $0.0 < fi$. Either "Ti" or "Tj" must be the same as one of the tank identification numbers (integers) specified by T1, T2 or T3 in this record.
Default: fi=1.0

This record may be repeated.

Record 16 Fuel Transfer Data

TRANSFER	{ wt FROM per PERCENT }	T1 TO T2
----------	----------------------------	----------

wt = Weight of fuel to be transferred from tank "T1" to tank "T2."
per = Percent ($0.0 < per \leq 100.$) of fuel in tank "T1" to be transferred to tank "T2."
T1,T2 = Fuel-tank, identification numbers (integers) previously-defined by Record 10.

This record, Records 14-16 or Records 13-16 may be repeated.

138.1.3.3 Fuel Orientation Data

The orientation of the fuel tanks associated with a mass SET and the fuel-surface plane to be used with these tanks may be defined via Records 17-19. The fuel-tank data may be calculated for a maximum of 99 different attitudes as defined by repeating Records 17-19. These records should not be input if the fuel-tank orientations as defined relative to the GLOBAL frame by Record 12 are to be used and if the fuel-surface plane is parallel to the GLOBAL X-Y plane.

Record 17 Attitude Identification

ATTITUDE n

n = Attitude identification number (integer) in the range 1-99, inclusive. If multiple attitudes are required, each attitude must be assigned a unique number.

Record 18 Fuel-Tank Orientation Data (optional)

The orientation of the fuel tanks is established by a reference frame which is oriented relative to the GLOBAL axes and which has its origin at a specified node.

Ncde rx ry rz

Ncde = Node number (integer) which identifies the origin of the orientation reference frame.

rx,ry,rz = Sequential, rotation angles in degrees about the X, Y and Z GLOBAL axes, respectively. These rotations establish the orientation of the fuel tanks for c.g. calculations.

Default Record: Fuel tanks are oriented relative to the GLOBAL reference frame as defined by Record 12.

Record 19 Fuel-Surface Plane Definition (optional)

A vector which is normal to the plane of the fuel-surface and which has a direction toward the empty region of a partially-full tank may be defined by this record.

FUEL PLANE p1 <p2>

p1,p2 = Either node numbers (integers) or a list of three GLOBAL X, Y and Z coordinates (decimal numbers). "p1" and "p2" define two points relative to the GLOBAL frame. A vector from "p1" to "p2" defines the fuel surface and the direction toward the empty region of the tanks.

Default: If "p2" is not input, the normal vector is from the GLOBAL origin through "p1."

Default Record: The normal vector is parallel to the GLOBAL Z-axis.

138.1.3.4 Fuel Condition Data

A fuel condition is defined by Record 20 by the total fuel weight, a fuel-management sequence and a fuel-tank attitude. A fuel condition is associated with an overall mass-distribution condition by Record 4. Record 20 may be repeated to define a maximum of 99 fuel conditions.

Record 20 Fuel-Condition Definition

CONDITION	n	Weight	Seq	<Attitude>
-----------	---	--------	-----	------------

n = Fuel-condition identification number (integer) in the range 1-99, inclusive. If multiple conditions are required, each condition must be assigned a unique number.

Weight = The total weight of fuel loaded for "Seq." If zero is input for this item, the weight used for this fuel condition will be the total fuel weight associated with "Seq."

Seq = A fuel-management, sequence identification number (integer) which was previously defined by Record 13.

Attitude = An attitude identification number (integer) which was previously defined by Record 17.
Default: The attitude defined by the GLOBAL reference frame.

Record 21 End Fuel Subset

END FUEL DATA

This record indicates that all fuel data have been defined for the SET identified by Record 2.

138.1.4 Payload Data Subset (optional)

This data subset, Records 22-35, defines the passenger seat locations, the cargo-hold geometries and payload loading sequences. Selected combinations of these mass model components are used to define a maximum of 99 payload mass-distribution conditions (Record 34) for the SET identified by Record 2. If payload data are not required, Records 22-35 should not be input.

Record 22 Begin Payload Subset

BEGIN PAYLOAD DATA

138.1.4.1 Passenger Data

The following record defines the average weight and weight moments of inertia for a passenger and seat. It is assumed that the c.g. of the passenger and the c.g. of the seat, as defined in the next section, are coincident.

Record 23 Passenger Weight Definition (optional)

PASSENGER Wt Ixx Iyy Izz Ixy Ixz Iyz

Wt = Average weight of each passenger

Ixx, Iyy, Izz } = Average weight moments of inertia of each passenger. These quantities are measured relative to a coordinate system that has the same orientation as the GLOBAL frame and has its origin at the c.g. of the passenger.
Ixy, Ixz, Iyz }

Default Record: PASSENGER 170. 0. 0. 0. 0. 0. 0.

138.1.4.2 Seat Location Data

The passenger-seat locations (the c.g. of each passenger and corresponding seat) are defined by points at which passengers are to be placed.

Coordinates of a seat location are defined relative to the GLOBAL or a local, right-handed, rectangular coordinate system (see sec. 100.5). All input coordinates are automatically transformed to GLOBAL values. Only the GLOBAL values are printed by the MASS Postprocessor (sec. 238.1).

Record 24 Begin Seat Location Data

BEGIN SEAT LOCATION DATA

If local reference frames are not used, Records 25 and 26 are not required. In this case, all seat location data are defined relative to the GLOBAL frame.

A maximum of 10 unique local frames may be defined by Record 25 placed sequentially or interspersed within a seat-location subset. All seat data defined via Record 26 are assumed to be input with respect to the local coordinate system defined by the immediately preceding Record 25. Seat data input prior to the first Record 25 are assumed to be defined with respect to the GLOBAL coordinate system. Record 26 allows a reference frame previously-defined within a seat-location subset to be identified as the input frame until another Record 25 or 26 is encountered.

Record 25 Definition of Local Input Frame (optional)

REC label Origin X-axis Z-axis

label = An alphanumeric word of 1 to 7 characters which identifies the particular local coordinate system. The word GLOBAL may not be used as a label.

Each of the items "Origin," "X-axis" and "Z-axis" denotes a list of three GLOBAL X, Y and Z coordinates (decimal numbers). Each item defines a point relative to the GLOBAL reference frame.

Origin = A point defining the origin of the local coordinate system.

X-axis = A point different from the one denoted by "Origin" which lies on the local positive x-axis.

Z-axis = A point different from "Origin" and "X-axis" which defines the local positive z-axis. This point may be any point with a positive z-coordinate in the x-z plane.

Records of this type may appear any number of times anywhere within a seat-location data subset. A "label" defined by one record of this type may be the "label" specified by a subsequent record of this type. The last defined origin and orientation of a particular triad

remains in effect until redefined. A maximum of 10 unique local frames may be effective at one time.

Record 26 A Previously-Defined Reference Frame to be Used as the Input Frame (optional)

RESUME	{ Label } { GLOBAL }
--------	-------------------------

{ Label }
{ GLOBAL } = The label assigned to a local coordinate system previously-defined by a Record 25 or the key-word GLOBAL. This identifies the input frame of all seat data defined by Record 27 immediately following this record.

Records of this type may appear any number of times within a seat-location data subset.

Record 27 Seat-Location Data

Seat numbers and coordinates may be defined by any combination of the three input-record variations presented below. These variations provide the following capabilities:

- 1) Explicit definition of a seat;
- 2) Generation of a sequence of equally-spaced seats;
- 3) Generation of a sequence of variably-spaced seats.

The input-record formats for seat data defined with respect to the GLOBAL or local coordinate systems are identical.

Variation 1 seat1 list1

seat1 = User seat number (integer) in the range 1-99999, inclusive.

list1 = List of 3 coordinates of "seat1."

Variations 2 and 3 allow the user to specify generation of a sequence of intermediate seats (seat numbers and coordinates) between two seats denoted by "seat1" and "seat2."

Variation 2 seat1 <list1> TO seat2 <list2>

seat1 }
seat2 } = Same as "seat1" in Variation 1

list1 }
list2 } = Lists of 3 coordinates associated with "seat1" and "seat2," respectively.

In general, coordinates are input as decimal numbers. However, if "seat1" and/or "seat2" were defined or generated previously, the associated coordinate list or lists may be omitted from these input records. In addition to simplifying the input required for seat-location data generation relative to a particular coordinate system, this feature allows the user to generate seat data spanning two different coordinate systems (GLOBAL/local or local/local).

Default: Previously-defined coordinates of either specified seat are automatically retrieved.

Equally-spaced seats between "seat1" and "seat2" are generated. Seat numbers are established by incrementing "seat1" by ± 1 depending on whether "seat1" or "seat2" is larger. Coordinates of the generated seats are obtained by linear interpolation between "list1" and "list2." A maximum of 101 seats may be defined by one record of this type.

Variation 3

seat1 <list1> TO seat2 <list2>
 <BY inc> <OF d1 d2 ... dn>

seat1 }
 seat2 }
 list1 }
 list2 }

= Same as defined for Variation 2

inc

= The seat-number increment (integer) to be used to generate seat numbers between "seat1" and "seat2." The quantity $[(seat2-seat1)/inc]$ must be a positive integer ≤ 100 .
 Default: "inc" is automatically set to ± 1 depending on whether "seat1" or "seat2" is larger.

d1,d2,...,dn

= List of proportionate spacing distances between each pair of seats in the sequence.
 Default: Seats generated from "seat1" to "seat2" are equally spaced via linear interpolation between "list1" and "list2."

Record 28 End Seat-Location Subset

END SEAT LOCATION DATA

This record indicates that all seat-location data have been defined for the SET identified by Record 2.

138.1.4.3 Cargo Hold Data

The geometry of a cargo hold is, in general, defined by multiple cargo-hold sections. The configuration of each section is defined by one or more 3-dimensional segments. All segments used to define a cargo hold must have the same basic shape. This shape is identified as TYPE1 or TYPE2 as shown by figure 138-1.

A cargo hold is defined by Records 29 and 30. These records may be repeated to define a maximum of 500 cargo holds.

Record 29 Cargo Hold Identification

HOLD n {TYPE1
TYPE2} <density>

n = Cargo-hold, identification number (integer) in the range 1-32767, inclusive. If multiple holds are required, each hold must be assigned a unique number.

{TYPE1
TYPE2} = Key-word denoting that either TYPE1 or TYPE2 geometric shapes are defined by Record 30 to describe the cargo-hold configuration.

density = Weight density of the contents of this cargo hold.
Default: 1.0 lbs/in.³ or the density specified for this item in the immediately-preceding record of this type.

Record 30 Cargo-Hold Section-Definition Data

Each hold section is defined by a list of node numbers which establishes the geometries of a sequence of TYPE1 or TYPE2 shapes (fig. 138-1). Any combination of the three input-record variations presented below may be used to define a maximum of 63 sections per cargo hold.

Variation 1 `<Id1> Nlist1`

Id1 = Alphanumeric word of the form Sxx where xx is an integer number in the range 1-63, inclusive. This word is used to identify the cargo-hold section described by this record.
Default: Sxx where xx is the i-th, sequential section defined for a hold by a record of this type.

Nlist1 = List of node numbers (integers) which define a section. 3 or 4 mid-surface nodes must be specified if TYPE1 shapes are used; 8 or 20 nodes must be specified if TYPE2 shapes are used.

Variations 2 and 3 allow the user to specify generation of a sequence of cargo-hold sections. The section identification numbers and the node numbers are generated by incrementing the first list until the second list is reached. Each of the items denoted by Id1 and Nlist1, when incremented, must reach the corresponding items denoted by Id2 and Nlist2 simultaneously. A maximum of 63 sections may be defined by one record of this type.

Variation 2 `<Id1> Nlist1 TO <Id2> Nlist2`

Items 1 and 2 are the same as defined for Variation 1.

Id2 = Identifier for the last section in the sequence. This item is defined in the same manner as "Id1" and is required only if "Id1" is specified.

Nlist2 = Node numbers (integers) defining the last section in the sequence. The number of items in Nlist1 and Nlist2 must be equal.

Variation 3 `<Id1> Nlist1 TO <Id2> Nlist2 BY <Inc1> inc2`

Items 1-5 are the same as defined for Variation 2.

Inc1 = The letter S followed by a signed or unsigned integer (e.g., S5, S-20, S+12)

which denotes the section, identification number increment. This item is required only if "Id1" and "Id2" are specified.

inc2 = List of positive and/or negative node-number increments (integers) to be used with Nlist1. The number of items in Nlist1 and "inc2" must be equal.

This record or Records 29 and 30 may be repeated.

138.1.4.4 Payload Management Data

Management data which define how passengers and cargo are to be loaded to establish a desired payload condition are input via Records 31-33. These records may be repeated to define a maximum of 99 payload-management sequences.

Record 31 Loading Sequence Identification

LOADING SEQUENCE n

n = Payload, loading-sequence, identification number (integer) in the range 1-99, inclusive. If multiple sequences are required, each sequence must be assigned a unique number.

Record 32 Seat Loading Data

The order in which particular seats are to be loaded is defined by any combination of the three input-record variations presented below. These variations allow seats to be identified in the following ways:

- 1) Explicit list of seat numbers;
- 2) Generation of a sequence of seat numbers by incrementing a list of numbers by one;
- 3) Generation of a sequence of seat numbers by incrementing a list of numbers by specified increments.

Variation 1

LOAD SEATS List1

List1 = List of up to 20 seat numbers (integers). The seats are loaded in the specified order.

Variations 2 and 3 allow the user to specify generation of seat numbers between two lists denoted by "List1" and "List2."

Variation 2

LOAD SEATS List1 TO List2

- List1 = Same as defined for Variation 1
- List2 = List of up to 20 seat numbers (integers) which identify the last seats to be loaded in the sequence. The number of items in "List1" and "List2" must be equal.

Intermediate seat numbers are established by incrementing the numbers in "List1" by ±1 until the numbers in "List2" are reached. Each of the numbers in "List1", when incremented, must reach the corresponding numbers in "List2" simultaneously.

Variation 3

LOAD SEATS List1 TO List2 BY inc

- List1 } = Same as defined for Variation 2
- List2 }
- inc = List of positive and/or negative seat-number increments (integers) to be used to increment "List1." The number of items in "inc" and "List1" must be equal.

This record may be repeated.

Record 33 Cargo Loading Data

The order in which the cargo holds are to be loaded for a particular payload-loading sequence is defined by this record.

LOAD CARGO HOLD Num IN DIRECTION	$\begin{Bmatrix} \pm X \\ \pm Y \\ \pm Z \end{Bmatrix}$
UNTIL	$\begin{Bmatrix} \text{FULL} \\ \text{wt1. LOADED} \\ \text{wt2 TOTAL} \end{Bmatrix}$

Num = Cargo-hold, identification number (integer) previously-defined by Record 29.

$\begin{Bmatrix} \pm X \\ \pm Y \\ \pm Z \end{Bmatrix}$ = The cargo-hold is filled in the GLOBAL reference-frame direction identified by this item. For example, the item "+Z" denotes that the hold

is loaded such that the vector which is normal to the cargo surface and which is directed toward the empty region of the hold is parallel to the GLOBAL Z-axis.

{ FULL
wt1 LOADED } = Cargo loading will stop when cargo-hold
wt2 TOTAL "Num" is FULL, when the weight of cargo
LOADED into "Num" equals "wt1," or when
the TOTAL weight of all loaded cargo
equals "wt2."

This record may be repeated.

138.1.4.5 Payload Condition Data

A payload condition is defined by Record 34 by a payload-loading sequence, the total number of passengers to be loaded and the total cargo weight to be loaded. A payload condition is associated with an overall mass-distribution condition by Record 4. Record 34 may be repeated to define a maximum of 99 payload conditions.

Record 34 Payload-Condition Definition

CONDITION	n	SEQUENCE	Seq	Pass	Weight
-----------	---	----------	-----	------	--------

n = Payload-condition identification number (integer) in the range 1-99, inclusive. If multiple conditions are required, each condition must be assigned a unique number.

Seq = A payload-loading sequence number (integer) which was previously defined by Record 31.

Pass = The total number (integer ≥ 0) of passengers loaded for "Seq."

Weight = The total weight, ≥ 0 , of cargo loaded for "Seq."

Record 35 End Payload Subset

END PAYLOAD DATA

This record indicates that all payload data have been defined for the SET identified by Record 2.

138.1.5 Concentrated Mass Data Subset (optional)

Concentrated masses are defined by this data subset. When diagonal or non-diagonal mass matrices or panel-weight matrices are calculated by the MASS Processor, no additional mass (weight) of the model is lumped at a node for which a concentrated mass is defined. Additionally, a concentrated mass is treated as an independent auxiliary panel when panel weight matrices are requested (see sec. 138.1.7). A concentrated mass subset to be included in a mass matrix generated by OPTION 4 of the MASS Processor is identified by the CONMASS parameter. Multiple concentrated masses defined at a node are cumulated.

Record 36 Begin Concentrated Mass Subset

```
BEGIN CONCENTRATED MASS DATA n
```

n = Concentrated mass subset number (integer) in the range 1 to 9, inclusive. Each concentrated mass subset for a SET must be identified by a unique number.

Record 37 Concentrated Mass Data

This record is repeated to define concentrated masses associated with the subset identified by Record 36.

```
<name> <Node> Cnode <O=Label1> <I=Label2>  
wt <i11 i22 i33 i12 i13 i23>
```

name = Alphanumeric word of 1 to 10 characters which identifies the concentrated mass. Default: The label CON-MASS is assigned to the concentrated mass.

Ncde = The number (integer) of a Structural node. This retained node defines the point to which the corresponding mass is to be transferred when calculating mass matrices. Default: "Cnode", if it is a retained Structural node. Concentrated masses may only be transferred to active Structural nodes.

Cnode = A node number (integer). This node defines the c.g. of the concentrated mass.

Label1 = The label assigned to a local coordinate system previously-defined by the nodal data (sec. 146.0). The output reference frame for the calculated mass inertias has its origin at "Node" and has the same orientation as the "Label1" triad. Default: If "Node" is specified, the output frame is the analysis frame of "Node." If "Node" is not specified and "Cnode" is an inactive node, the output frame has the same orientation as the GLOBAL frame.

Label2 = The label assigned to a local coordinate system previously-defined by the nodal data (sec. 146.0). The input reference frame for the inertia values has its origin at "Cnode" and has the same orientation as the "Label2" triad. Default: The input frame has its origin at "Cnode" and has the same orientation as the GLOBAL frame.

wt = Weight of the concentrated mass.

$\langle \begin{matrix} i11, i22, i33 \\ i12, i13, 23 \end{matrix} \rangle$ = Weight moments of inertia of the concentrated mass relative to the "Label2" input triad 1-2-3. The products of inertia are defined as follows:
 $i12 = -\rho \int x(1) * x(2) dv,$
 $i13 = -\rho \int x(1) * x(3) dv$
and $i23 = -\rho \int x(2) * x(3) dv$
where ρ is the mass density. Trailing zeros need not be input.
Default: Zero weight moments of inertia.

Record 38 End Concentrated Mass Subset

END CONCENTRATED MASS DATA

This record indicates that a concentrated mass subset has been defined. Records 36-38 may be repeated to define a maximum of 9 concentrated mass subsets for the SET identified by Record 35.

138.1.6 Mass Lumping Data Subset (optional)

When the MASS Processor (sec. 238.0) is executed with either the parameter OPTION=2 or OPTION=3, each incremental mass associated with a mass distribution CONDITION (Record 4) is lumped at the closest retained node. If the mass within a geometric region, defined via an element subset, is to be lumped at selected retained nodes within the model, mass lumping data must be defined by this data subset. If mass lumping data are not required, Records 39-41 should not be input.

Record 39 Begin Lumping Subset

BEGIN LUMPING DATA

Record 40 Mass Lumping Data

The mass associated with either stiffness or mass element subsets or with the fuel and payload conditions may be lumped at selected retained nodes included in a node subset.

LUMP	{ STIFFNESS SUBSETS ATlist MASS SUBSETS ATlist FUEL PAYLOAD }	AT NODE SUBSET Num
------	--	--------------------

ATlist = ATLAS list of Stiffness or Mass element subset numbers (integers) previously-defined via the SUBSET-DEFINITION data (sec. 156.0). The mass associated with the identified elements is to be lumped at the retained nodes included in "Num."

Num = Integer number of a node subset previously-defined by the SUBSET-DEFINITION data. Non-retained nodes included in this subset are ignored.

This record may be repeated to define all distinct mass lumping regions and to identify which mass model components must be lumped at selected retained nodes.

Record 41 End Lumping Subset

END LUMPING DATA

This record indicates that all mass lumping data have been defined for the SET identified by Record 2.

138.1.7 Auxiliary Panel Data Subset (optional)

The weight contained within cylindrical regions of a mass model may be calculated from the information defined by this data subset. Each cylinder is generated by translation of a polygon in a specified direction. The weight and inertias of selected elements and portions thereof contained within a cylinder are lumped at the center of gravity of the contained mass. These data are the components of the panel-weight matrix generated by the MASS Processor. Each concentrated mass subset associated with a mass-distribution CONDITION defined via Record 4 is treated as an additional panel when calculating a panel-weight matrix. If panel-weight matrices are not required, Records 42-46 should not be input.

Record 42 Begin Auxiliary Panel Subset

BEGIN PANEL DATA n

n = Panel subset number (integer) in the range 1 to 9, inclusive. Each panel subset associated with a mass data set must be identified by a unique number.

Records 43-45 are input to define all weight panels within one panel subset.

Record 43 allows the user to specify certain mass model components (subsets) to be associated with selected panels in the calculation of panel-weight matrices. Selected components are identified as payload subsets, fuel subsets/conditions or as mass and/or stiffness element subsets.

Record 43 Panel Element Subsets (optional)

Any combination of the following three variations of this record may be input provided a particular variation of this record appears only once prior to Record 45. Subsets defined by this record are associated with the data of Record 45 until redefined by another Record 43.

Variation 1 MASS SUBSETS n1 n2 ...

n1,n2,... = Mass element subset numbers (integers) defined via the Subset-Definition data (sec. 156.0). This type of subset includes only non-structural elements defined by Records 16-18.

Variation 2 STIFFNESS SUBSETS n1 n2 ...

n1,n2,... = Stiffness element subset numbers (integers) defined via the Subset-Definition data.

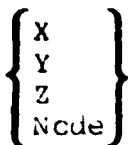
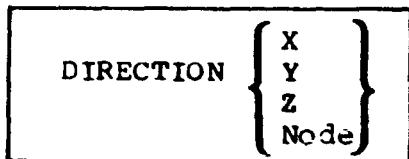
Variation 3 <PAYLOAD> <FUEL>

Key-words that denote whether the PAYLOAD and/or FUEL conditions are to be included in the panel weight calculations.

Default Record: All stiffness, mass, fuel and payload elements will be included in the panel weight calculations. If any form of this record is input, only the specified conditions will be included.

Record 44 Direction of Cylinder Axis (optional)

This record defines the direction of the cylinder axis for the panels defined by Record 45. This direction remains in effect until redefined by another Record 44. If a Record 43 is input, Record 44 may be input again only after input of another Record 43. If no Record 43 is input, this record may be input to define different directions of cylinder axes for the panels defined by Records 45.



= The cylinder axis is parallel to either the GLOBAL X, Y or Z axis or it is parallel to the line connecting the origin of the GLOBAL frame to the node identified by the number "Node" (integer).

Default Record: The direction specified by the last Record 44 or DIRECTION Z.

Record 45 Auxiliary Panel Data

Auxiliary panels may be defined by any combination of the three variations presented below. Records of this type may be repeated.

Variation 1 ident1 <O=Label> Nlist

- ident1 = Panel number (integer) in the range 1 to 999, inclusive. Each panel associated with a data set must be identified by a unique number.

- Label = The label assigned to a local coordinate system previously-defined by the nodal data (sec. 146.0). The output reference frame for the calculated mass inertias has its origin at the c.g. of the mass contained in the polygonal cylinder and has the same orientation as the "Label" triad.
 Default: The output frame has its origin at the c.g. of the mass contained in the cylinder and has the same orientation as the GLOBAL frame.

- Nlist = List of 3 to 8 node numbers (integers) which define a panel. The nodes must be coplanar and must be listed consecutively around the panel. A panel must have interior angles less than 180°.

Variations 2 and 3 allow the user to specify a sequence of panels to be generated between panels identified by "ident1" and "ident2."

Variation 2 ident1 <O=Label> Nlist TO ident2

- | | | |
|--------|---|------------------------------------|
| ident1 | } | = Same as defined for Variation 1. |
| Label | | |
| Nlist | | |

- ident2 = Panel number (integer) in the range 2-999, inclusive. This number must be greater than "ident1."

A series of panels identified by "ident1" through "ident2" in increments of one is generated. The nodes defining

each of these panels are established by incrementing the list of nodes in "Nlist" by one until the sequence is completed.

Variation 3 ident1 <O=Label> Nlist TO ident2
BY inc <d1 d2 ...>

ident1 }
Label } = Same as defined for Variation 2.
Nlist }
ident2 }

inc = The increment (integer) to be used to generate panel numbers between "ident1" and "ident2." The quantity [(ident2-ident1)/inc] must be greater than zero.

d1,d2,... = List of 3 to 8 positive, negative or zero integers used to increment the list of nodes in "Nlist." Trailing ones need not be input.
Default: Node lists generated from "Nlist" are established by incrementing the list of nodes in "Nlist" by one.

Record 46 End Auxiliary Panel Subset

END PANEL DATA

This record indicates that a panel subset has been defined. Records 42-46 may be repeated to define a maximum of 9 panel subsets for the SET identified by Record 2.

Example: Example records to illustrate an auxiliary panel data subset follow:

```

BEGIN PANEL DATA           3 /
1    10    11    12        /
2    20    21    22        /
DIRECTION Y                 /
3    30    31    32        /
MASS SUBSETS                2 3 /
FUEL                        /
DIRECTION Z                 /
4    40    41    42        /
5    50    51    52        /
END PANEL DATA            /

```

Panel subset 3 is comprised of 5 auxiliary panels. The cylinders associated with panels 1 and 2 are parallel to the GLOBAL Z-axis and all mass elements are paneled. Panel 3 is associated with a cylinder which is parallel to the Y-axis with all elements paneled, whereas panels 3 and 4 are associated with cylinders which are parallel to the Z-axis with only fuel condition data and mass element subsets 2 and 3 paneled.

138.1.8 Weight-Statement Label Data Subset (optional)

This data subset, Records 47-49, defines a detailed, weight-statement printout format with 1 to 3 levels of weight summation.

Level 3 Weights-- Weights of elements in specified subsets are summed.

Level 2 Weights-- The Level 3 weights are selectively summed.

Level 1 Weights-- The Level 2 weights are selectively summed.

Record 47 Begin Label Subset

BEGIN LABEL DATA

Record 48 Label Data

Ident * label *

Ident = A key-word denoting a weight summation level or an element subset identification. The options for "Ident" are as follows:

LEVEL1-- Key-word denoting that all LEVEL2 records between this record and the next LEVEL1 record (if any) are to be summed.

LEVEL2-- Key-word denoting that all LEVEL3 records between this record and the next LEVEL2 record (if any) are to be summed.

EKxxx<a> } -- Alphabetic word where xxx is the
EMxxx<a> } number of the stiffness (K) or non-
structural mass (M) element subset
denoted by EKxxx and EMxxx, respectively,
to be associated with "label." The
characters xxx denote an element subset

number defined by the SUBSET-DEFINITION Preprocessor (sec. 156.0). The optional <a> is effective only for any SPAR, COVER and CCOVER element types (see appendices B and C) which may be included in the specified subset. This portion of the key-word is input as the 3 characters (U), (L) or (W). These characters denote that only the upper (U) surface and flange data, only the lower (L) surface and flange data, or only the SPAR web (W) data for the corresponding elements in the subset are to be included. The weight of the elements in the subset is assigned to a level one lower than denoted by the last LEVELi record. If no LEVELi record has been input previously, the subset weight is assigned to Level 1.

$\left\{ \begin{array}{l} \text{Fxx} \\ \text{Pxx} \\ \text{Cxx} \end{array} \right\}$ -- Alphanumeric word where xx is the integer number of a fuel (F) condition, a payload (P) condition or a concentrated mass (C) subset defined by Records 20, 34 and 36, respectively, to be associated with "label." The weight of the elements in the specified subset is assigned to Level 1.

label = 40 characters of alphanumeric text to be used as a label in the weight statement.

This record is repeated to define which subset element weights are to be summed for each level and to define labels for each printed line of a weight statement (see examples in appendix F).

Record 49 End Label Subset

END LABEL DATA

This record indicates that all weight-statement data have been defined for the SET identified by Record 2.

138.1.9 Factor Data Subset (optional)

Record 50 Begin Factor Subset

BEGIN FACTOR DATA

Any combination of Records 51-56 may be input.

Record 51 Subset Weight Factors (optional)

This record allows the weight of selected subsets of elements to be factored.

MULTIPLY { MASS STIFFNESS } SUBSET Num<a> BY { factor TABLE Ident }

Num<a> = The number (integer) of the subset whose weight is to be factored. This number is either a STIFFNESS or non-structural MASS element subset number defined by the SUBSET-DEFINITION Preprocessor (sec. 156.0). The optional <a> is effective only for any SPAR, COVER and CCOVER element types (see appendices B and C) which may be included in the specified subset. This portion of the data item is input as the 3 characters (U), (L) or (W). These characters denote that only the upper (U) surface and flange area, only the lower (L) surface and flange area, or only the SPAR web (W) weight is to be factored by "factor."

factor = A factor greater than or equal to zero by which the specified subset weights are to be multiplied.

Ident = The identification number (integer) of a factor table as defined by subsequent Records 52-54. Weight factors for the components of the specified Mass or Stiffness subset are extracted from the factor table.

Default Record: Weights are calculated with a factor of one.

This record may be repeated to factor the weights associated with a maximum of 100 subsets.

Weight-factor tables as required for Record 51 are defined by Records 52-54. These tables allow weight factors to be defined as functions of the geometric properties of selected elements.

Record 52 Factor Table Identification and Function Definition

TABLE ident *# Function #

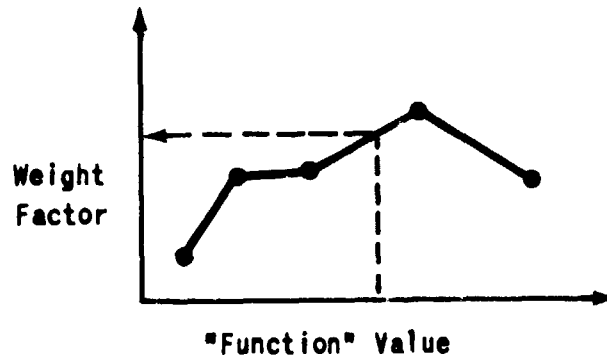
ident = Weight-factor table identification number (integer) in the range 1 to 36, inclusive. Each table defined by Records 53 and 54 for a mass SET must be identified by a unique number.

Function = An expression that is a function of 1 to 3 geometric, elemental properties. The value of this function, when evaluated for a particular element, is to be used to "look up" its weight factor. The general form of "Function" is $c_1 \text{ op}_1 P_1 \text{ op}_2 P_2 + c_2 \text{ op}_3 P_3 + c_3$ where "ci" is a positive or negative constant, "opi" is the multiply or divide operator denoted by * and /, respectively, and "Pi" is one of the key-words in the following table used to identify a particular property for one or more element types.

Example Record: TABLE 4 *# 1.5*L/A-.05#

ELEMENT PROPERTY		ELEMENT TYPE											
KEY-WORD	DESCRIPTION	ROD	BEAM	SPAR	COVER	PLATE	GPLATE	BRICK	SCALAR	SRUD	SPLATE	CPLATE	COVER
A	CROSS-SECTION AREA	✓	✓						✓				
A(L)	LOWER FLANGE AREA			✓									
A(U)	UPPER FLANGE AREA			✓									
AS	SURFACE AREA			✓	✓	✓	✓				✓	✓	✓
D	DEPTH (2 * Δz)			✓									
L	LENGTH	✓	✓	✓					✓				
P	PERIMETER			✓	✓	✓	✓				✓	✓	✓
T	PLATE THICKNESS					✓	✓						
T(L)	LOWER PLATE THICKNESS				✓								✓
T(U)	UPPER PLATE THICKNESS				✓								✓
T(W)	WEB THICKNESS			✓									

Discrete values of the "Function" and corresponding weight-factor values are defined by Records 53 and 54, respectively. A maximum of 250 function values may be specified for a particular table. Linear interpolation of these values is generally performed to extract the weight factor for a particular element. Extrapolation of the table values is not permitted. The following figure illustrates a weight-factor table.



Record 53 Function Value Data

val1 val2 . . . valn

vali = The i-th "Function" value ($1 \leq i \leq 250$) for which a corresponding weight-factor value is specified by Record 54. The list of "vali" values must be input in a strictly increasing or decreasing order. The "Function" is assumed to be single-valued as illustrated in the foregoing figure.

Record 54 Weight Factor Data

fac1 fac2 . . . facn

faci = The value of the weight-factor associated with "vali" specified by Record 53. The number and sequence of items input by this record must correspond to the number and sequence of items input by Record 53.

Records 52-54 may be repeated to define a maximum of 36 weight-factor tables.

Record 55 Element Exclusion Data (optional)

EXCLUDE { MASS
 STIFFNESS } ELEMENTS

This record allows the MASS elements defined by Records 6-8 or the STIFFNESS elements to be excluded from all mass/weight calculations. Both variations of this record may be input.

Default Record: All elements are included.

Record 56 Mass Matrix Divisor (optional)

MASS FACTOR factor

factor = The value greater than zero by which weights are to be divided to obtain mass properties.

Default Record: Weights are divided by 386.04 to convert them to masses.

Record 57 End Factor Subset

END FACTOR DATA

This record indicates that all Factor data have been defined for the SET identified by Record 2.

Record 58 End Data Set

END MASS DATA

Additional mass data sets may be input by repeating Records 1-58.

Table 138-2. Summary of Mass Data Records

Reference Page	Data Records
138.3	BEGIN MASS DATA
138.3	SET n
138.3	BEGIN CONDITION DATA
138.3	{ STAGE St { PANEL DATA Sub } } CONDITION cond Fuel Payload Conmass
138.5	END CONDITION DATA
138.5	BEGIN MASS ELEMENT DATA
138.6	EI type $\left\langle \begin{matrix} F1 \\ F2 \\ F3 \end{matrix} \right\rangle$ <Userid> Nlist1 plist <TO <Userid> Nlist2 <BY <Inc'>inc2>>
138.8	END MASS ELEMENT DATA
138.10	BEGIN FUEL DATA
138.10	TANK n
138.10	$\left\langle \begin{matrix} TYPE1 \\ TYPE2 \end{matrix} \right\rangle$ <levels><density><useable PERCENT>
138.11	<Id1> Nlist1 <TO <Id2> Nlist2 <BY <Incj> inc2>>
138.14	MANAGEMENT SEQUENCE n
138.14	LOAD TANK T1 <T2><T3><RATIO r1 r2 r3> UNTIL List
138.15	USE TANK T1 <T2><T3><RATIO r1 r2 r3> UNTIL List
138.16	TRANSFER $\left\{ \begin{matrix} wt FROM \\ per PERCENT \end{matrix} \right\}$ T1 TO T2
138.17	ATTITUDE n
138.17	Node rx ry rz
138.17	FUEL PLANE pl <p2>
138.18	CONDITION n Weight Seq <Attitude>
138.18	END FUEL DATA
138.19	BEGIN PAYLOAD DATA
138.19	PASSENGER wt lxx lyy lzz lxy lxz lyz
138.20	BEGIN SEAT LOCATION DATA
138.20	REC label Origin X-axis Z-axis
138.21	RESUME $\left\{ \begin{matrix} Label \\ GLOBAL \end{matrix} \right\}$
138.21	seat1<list1><TO seat2<list2><<BY inc> <OF d1 d2 ...>>>
138.23	END SEAT LOCATION DATA
138.23	HOLD n $\left\{ \begin{matrix} TYPE1 \\ TYPE2 \end{matrix} \right\}$ <density>
138.24	<Id1> Nlist1 <TO <Id2> Nlist2 <BY <Inc1> inc2>>
138.25	LOADING SEQUENCE n
138.25	LOAD SEATS List1 <TO List2 <BY inc>>
138.26	LOAD CARGO HOLD Num IN DIRECTION $\left\{ \begin{matrix} \pm X \\ \pm Y \\ \pm Z \end{matrix} \right\}$ UNTIL $\left\{ \begin{matrix} FULL \\ wt1 LOADED \\ wt2 TOTAL \end{matrix} \right\}$
138.27	CONDITION n SEQUENCE Seq Pass Weight
138.27	END PAYLOAD DATA

Table continued on next page.

Table 138-2. Summary of Mass Data Records (Cont'd.)

Reference Page	Data Records
138.28	BEGIN CONCENTRATED MASS DATA n
138.28	<name><Ncde> Cnode <0=Label1><1=Label2> wt <i11 i22 i33 i12 i13 i23>
138.29	END CONCENTRATED MASS DATA
138.30	BEGIN LUMPING DATA
138.30	LUMP { STIFFNESS SUBSETS ATlist MASS SUBSETS ATlist FUEL PAYLOAD } AT NODE SUBSET num
138.30	END LUMPING DATA
138.31	BEGIN PANEL DATA n
138.31	{ MASS STIFFNESS } SUBSETS n1 n2 ...
138.32	<PAYLOAD><FUEL>
138.32	DIRECTION { X Y Z Node }
138.33	ident1 <0=Label> Nlist <TO ident2 <BY inc <d1 d2 ...>>>
138.34	END PANEL DATA
138.35	BEGIN LABEL DATA
138.35	Ident * ≠ label ≠
138.36	END LABEL DATA
138.37	BEGIN FACTOR DATA
138.37	MULTIPLY { MASS STIFFNESS } SUBSET Num<a> BY { factor TABLE Ident }
138.38	TABLE ident * ≠ Function ≠
138.39	val1 val2 ...
138.39	fac1 fac2 ...
138.40	EXCLUDE { MASS STIFFNESS } ELEMENTS
138.40	MASS FACTOR factor
138.40	END FACTOR DATA
138.40	END MASS DATA

140.0 MATERIAL DATA

Material property data for the ATLAS stiffness elements are comprised of the following:

- a) A reference code for use in the finite element definition records (sec. 152.1);
- b) Material density for mass analyses of a stiffness model;
- c) Elastic constants for stiffness/stress analyses;
- d) Allowable stress levels for design studies.

The following types of linearly elastic materials may be used to define a finite element model.

- a) Standard Commonly-used, isotropic materials included in the standard-material catalog (sec. 140.1 and appendix A)
- b) Special -- Generally, three-dimensional orthotropic materials defined by the user
- c) Composite -- Two-dimensional orthotropic materials defined by the user for analysis of composite elements (ref. 140-1)

Data must be input to the MATERIAL Preprocessor to define Special and Composite material properties. Standard material properties are loaded automatically by the ATLAS System and stored internally for direct reference in the element-definition records.

The units indicated for the Special and Composite material properties (sec. 140.2.1 and 140.2.2) are the same as those used for the Standard materials and are consistent with the data units required by the interfacing preprocessors.

Printout of the Material data may be requested from the MATERIAL Postprocessor (sec. 240.0).

PRECEDING PAGE BLANK NOT FILMED

140.1 STANDARD MATERIAL DATA

Material input data are not required to access the properties for Standard materials. These data are stored internally for immediate reference by a material code (see table below) in the Stiffness and Design data sets.

The standard-material catalog of linearly elastic, isotropic materials is currently comprised of the following:

REFERENCE CODE	MATERIAL	TEMPERATURE RANGE, °F
M1	2024-T351 Aluminum--Clad Plate (0.50-1.0 in.)	-200 to +500
M2	7075-T6 Aluminum--Clad Plate and Sheet (0.188-0.499 in.)	-200 to +500
M3	7075-T73 Aluminum--Forging (≤ 3.0 in.)	-200 to +500
M4	6AL-4V Annealed Titanium--Bar and Forging	-400 to +1100
M5	6AL-4V STA1000 Titanium--Sheet and Plate (≤ 0.750 in.)	-300 to +1100
M6	6AL-6V-2SN STA1100 Titanium--Forging (≤ 2.0 in.)	-100 to +1100
M7	AISI 321 Annealed Stainless Steel--Sheet and Bar	-400 to +1600
M8	AISI 4130, 4135 or 4140 Steel (150-170 ksi H.T.) Wrought Products	-200 to +1000
M9	AISI 4330M Steel (200 ksi H.T.)--Wrought Products	-200 to +1000
M10	AISI 4340M Steel (275 ksi H.T.)--Bar and Forging	-200 to +1000

The property values for each of these materials are defined at discrete temperature levels as illustrated in Appendix A. A sufficient number of temperature intervals are defined for each material property such that linear interpolation within the selected temperature range yields reasonable values. Reference to property data outside the specified temperature range is not allowed. The density of each Standard material was selected at 70°F.

The single set of nominal property data and forms selected for each alloy are representative of that material (ref. 140-2). However, since property data for a specific form or thickness may differ significantly from these nominal values, caution must be exercised.

140.2 INPUT DATA

A maximum of 49 Special and 31 Composite materials may be defined per job execution. These data must be defined prior to referencing them by another data set. No other data set must be input prior to this data set in the input stream.

Record 1 Data Set Identification

BEGIN MATERIAL DATA

This record initiates execution of the MATERIAL Preprocessor.

140.2.1 Special Material Data (optional)

Special materials are defined via Records 2-4. Records 3 and 4 are repeated for each Special material.

Record 2 Material Type Identification (optional)

SPECIAL MATERIAL DATA

This record indicates that subsequent records define Special materials.

Default Record: The type of material is identified by the reference code specified in the next record.

Record 3 Special Material Identification

Mcode <md>

Mcode = Special material reference code--the character M followed by an integer in the range 51 to 99, (i.e., M51-M99). A particular material code and associated property values may not be redefined.

md = Material density, ρ , (lbs/in.³)
Default: md = 0.0

Record 4 Special Material Properties

All the property values for a Special material must be defined via one of the three input record variations presented below. In each case, zeros (in proper format) must be input for items not used. One record of this type is required for each temperature to be specified for a

material. These records must define temperature values in a numerically increasing order. Property values may be specified at 1-50 temperature levels. The interpolation techniques used for the Standard material properties are also used to generate Special material properties at unspecified temperatures.

140.2.1.1 Variation 1--Isotropic Materials

p1	p2	p3	p4	p5
----	----	----	----	----

- p1 = Temperature (°F as an integer) associated with the properties that follow:
- p2 = Young's modulus, $E1$, (lbs/in.²)
- p3 = Poisson's ratio, $\nu1$
- p4 = Shear modulus, $G1$, (lbs/in.²)
- p5 = Linear thermal strain, ϵ_{T1} , for thermal stress analyses.

The capability of defining the three dependent elastic constants ($E1$, $G1$, and $\nu1$) for a material allows the user to define one of the following conditions: (1) the element is isotropic if the expression $E1 = 2G1*(1+\nu1)$ is satisfied; (2) the element behaves in a nonisotropic sense if the foregoing equation is not satisfied by the specified elastic constants. This feature, for example, allows the user to model more accurately the behavior of buckled panels or beam webs using, for instance, the SPAR or PLATE elements (see appendix B).

140.2.1.2 Variation 2--Orthotropic Materials

Orthotropic material property values are defined relative to the orthogonal, principal (natural) axes, denoted by 1-2-3, of the material. The principal axes of a material are assumed to be coincident with the reference axes x-y-z, respectively, of each element composed of that material unless indicated otherwise in the element-definition record. An element that is configured from a Special, isotropic material possesses the properties defined via Variation 1 in all directions within the material. This characteristic is handled automatically by the program. The user can, however, define explicitly the same property values for an isotropic material via Variation 2.

p1 p2 p3, ..., p13

- p1 = Temperature ($^{\circ}$ F as an integer) associated with the properties that follow:
- p2 = Young's modulus, E_1 , associated with the first principal direction of the material (lbs/in. 2)
- p3 = Poisson's ratio, ν_{12} , associated with the 1-2 plane
- p4 = Shear modulus, G_{12} , associated with the 1-2 plane (lbs/in. 2)
- p5 = Linear thermal strain, ϵ_{T1} , in direction 1
- p6-p9 = Properties p2 to p5 associated with the second principal direction of the material (i.e., E_2 , ν_{23} , G_{23} and ϵ_{T2} , respectively)
- p10-p13 = Properties p2 to p5 associated with the third principal direction of the material (i.e., E_3 , ν_{31} , G_{31} and ϵ_{T3} , respectively)

For all cases, the Poisson's ratios must be in the range $0.0 \leq \nu_{ij} \leq 1.0$ where the calculated values are obtained from the expression $\nu_{ji} = \nu_{ij}(E_j/E_i)$.

140.2.1.3 Variation 3--Orthotropic Materials with Design Allowables

This variation permits stress allowables to be defined for Special materials referenced via the Design Data (sec. 112.0). The features mentioned above for Variation 2 also apply to this variation.

p1 p2 p3, ..., p31

- p1-p13 = Same as defined for Variation 2
- p14-p16 = Ultimate tension stresses, FTU_1 , FTU_2 and FTU_3 in directions 1-2-3, respectively (lbs/in. 2)
- p17-p19 = Ultimate compression stresses, FCU_1 , FCU_2 and FCU_3 in directions 1-2-3, respectively (lbs/in. 2)

- p20-p22 = Ultimate shear stresses, FSU1, FSU2 and FSU3 in planes 1-2, 2-3 and 3-1, respectively (lbs/in.²)
- p23-p25 = Yield tension stresses, FTY1, FTY2 and FTY3 in directions 1-2-3, respectively (lbs/in.²)
- p26-p28 = Yield compression stresses, FCY1, FCY2 and FCY3 in directions 1-2-3, respectively (lbs/in.²)
- p29-p31 = Yield shear stresses, FSY1, FSY2 and FSY3 in planes 1-2, 2-3 and 3-1, respectively (lbs/in.²)

Additional Special materials with unique reference codes may be defined by repeating Records 3 and 4.

140.2.2 Composite Material Data (optional)

Composite materials required for analysis of composite finite elements are defined via Records 5-7. Records 6 and 7 are repeated for each Composite material. Property data are specified for a layer (a tape composed of fibers and a matrix) which is the basic unit of a composite element (a laminate). Multiple layers are used to define laminas for composite elements via the element-definition records.

Record 5 Material Type Identification

COMPOSITE MATERIAL DATA

This record indicates that subsequent records define Composite materials.

Record 6 Composite Material Identification

Ccode t <md>

Ccode = Composite material reference code--the character C followed by an integer in the range 1 to 31, (i.e., C1-C31). A particular material code and associated property values may not be redefined.

t = Layer thickness, (in.)

md = Material area density, $\rho * t$, (lb/in.²),
defining the mass per unit area of the
layer.
Default: md = 0.0

Record 7 Composite Material Properties

All the property values for a Composite material must be defined via one of the two input record variations presented below. In each case, zeros (in proper format) must be input for items not used. One record of this type is required for each temperature to be specified for a material. These records must define temperature values in a numerically increasing order. Property values may be specified at 1-10 temperature levels. The interpolation techniques used for the Standard material properties are also used to generate Composite material properties at unspecified temperatures.

140.2.2.1 Variation 1--Material Elastic Properties

p1	p2	p3	p4	p5	p6	p7
----	----	----	----	----	----	----

p1 = Temperature ($^{\circ}$ F as an integer) associated with the properties that follow:

p2, p3 = Young's moduli, E1 and E2, associated with the first and second principal directions of the material, respectively (lbs/in.²). Direction 1 should correspond to the fiber direction of the Composite material.

p4 = Poisson's ratio, ν_{12} , associated with the 1-2 plane

p5 = Shear modulus, G12, associated with the 1-2 plane (lbs/in.²)

p6, p7 = Linear thermal strains, ϵ_{T1} and ϵ_{T2} , in directions 1 and 2, respectively

140.2.2.2 Variation 2--Material Elastic Properties and Design Allowables

This variation permits stress allowables to be defined for Composite materials (CPLATE and CCOVER elements) if lamina margins of safety are to be printed (sec. 254.2.2), or if composite elements are referenced by the Design Data (sec. 112.0).

p1 p2 p3,...,p17

- p1-p7 = Same as defined for Variation 1
- p8,p9 = Ultimate tension stresses, FTU1 and FTU2, in directions 1 and 2, respectively (lb/in.²)
- p10,p11 = Ultimate compression stresses, FCU1 and FCU2, in directions 1 and 2, respectively (lb/in.²)
- p12 = Ultimate shear stress, FSU, in the 1-2 plane (lbs/in.²)
- p13,p14 = Yield tension stresses, FTY1 and FTY2, in directions 1 and 2, respectively (lbs/in.²)
- p15,p16 = Yield compression stresses, FCY1 and FCY2, in directions 1 and 2, respectively (lbs/in.²)
- p17 = Yield shear stress, FSY, in the 1-2 plane (lbs/in.²)

The material directions must be selected such that the following expressions are satisfied:

$$\begin{array}{ll} FTU1 \geq FTU2 & FCU1 \geq FCU2 \\ FTY1 \geq FTY2 & FCY1 \geq FCY2 \end{array}$$

Additional materials with unique reference codes may be defined by repeating Records 6 and 7 or Records 2-7.

Record 8 End Data Set

END MATERIAL DATA

Additional material data sets may be input by repeating Records 1-8.

Table 140-1. Summary of Material Data Records

Reference Page	Data Records
140.4	BEGIN MATERIAL DATA
140.4	SPECIAL MATERIAL DATA
140.4	Mcode <md>
140.5	p1 p2 ... p5 <p6 p7 ... p13 <p14 p15 ... p31>>
140.7	COMPOSITE MATERIAL DATA
140.7	Ccode t <md>
140.8	p1 p2 ... p7 <p8 p9 ... p17>
140.9	END MATERIAL DATA

146.0 NODAL DATA

The nodal data set defines the geometrical data necessary for structural and/or mass finite-element definition by the stiffness and mass input data (sec. 152.0 and 138.0). Nodal data are interrogated by many of the other modules and thus, this data set is normally the first data set within a data deck. Only the geometry data set (ATLAS data -- sec. 126.0; GCS or NASA-LRCGEOM data--appendix G), if required, must be input before this data set. Geometry data permit nodal coordinates to be extracted from previously-defined geometric surfaces.

Nodal data are comprised of node numbers and coordinates. A nodal data set that is used for a structural and/or a mass model must be identified by the same SET number assigned to the model by the stiffness and/or mass data (sec. 152.0 and 138.0). A maximum of 36 nodal data SETS may be defined per job. Each set may contain a maximum of 4095 nodes.

Coordinates of a node are defined relative to a selected input reference frame, whereas the kinematics of a node are described and analyzed relative to a selected analysis frame. The input and analysis reference frames associated with a node may be different. The GLOBAL and up to a maximum of 4095 local reference frames may be used for these purposes. The local frames may be rectangular, cylindrical and/or spherical coordinate systems (see sec. 100.5). All coordinate systems are orthogonal and right-handed.

There is always a GLOBAL frame associated with a nodal data set. The origin and orientation of this frame are defined implicitly by the input nodal coordinates, whereas the origin and orientation of each local reference frame are explicitly defined relative to the GLOBAL frame. All coordinates input with respect to local coordinate systems are automatically transformed to GLOBAL values. Only the GLOBAL coordinates are printed by the NODAL Postprocessor (sec. 246.1).

There are four types of nodes:

- a) Structural -- A node which may be externally loaded and to which a stiffness finite element is attached. Structural nodes generally have three translational and three rotational degrees of freedom.

PREVIOUS PAGE BLANK NOT FILMED

- b) Shear -- Similar to a Structural node but used only to define SROD and SPLATE finite elements (see appendix B). A Shear node has only one translational degree of freedom which is either along an SROD axis or along an SPLATE edge. The analysis frame of all Shear nodes must be identified as the GLOBAL frame. Although the geometrical location of Shear nodes is immaterial, they may be assigned explicit coordinates so that they are sorted during the geometry-dependent node reordering (see Record 13 of nodal-data input).
- c) Auxiliary -- A node used to orient and/or position a finite element relative to Structural nodes. The distortion of an element is not affected by the kinematics of these nodes. An Auxiliary node of one element may be used as a Structural node for another element.
- d) Mass -- A node used to define mass finite elements and concentrated masses. A Mass node may also be used as one of the foregoing types of nodes in defining a model.

An ATLAS node may be defined with three or four coordinates depending on how the node is used to describe the elements. Nodes with four coordinates are referenced as mid-surface nodes. These nodes are required only to define SPAR, COVER and CCOVER elements. Coordinates of a mid-surface node defined with respect to a rectangular frame are specified by a list of x , y , z and Δz components. The Δz component results in two implicit points, $(z+\Delta z)$ and $(z-\Delta z)$, located along a line parallel to the z -axis. Thus, three points are established on a straight line wherein the middle point lies on the "mid-surface" of the finite element (see appendix B and C). Other types of elements may be defined via mid-surface nodes.

Printout of the nodal data may be requested from the NODAL Postprocessor (sec. 246.1). Plots of these data can also be effected via the EXTRACT and GRAPHICS Postprocessors as discussed in sections 218.0 and 228.0.

146.1 INPUT DATA

Record 1 Data Set Identification

BEGIN NODAL DATA

This record initiates execution of the NODAL Preprocessor.

Record 2 Data Set Number (optional)

Each set of nodal data is identified by an integer number. If multiple sets are used in a particular job, each set must be assigned a different number by this record. Multiple nodal data sets are defined by repeating Records 2-13 or Records 1-14.

SET n

n = Data set number (integer) in the range 1 to 36, inclusive.

Default Record: SET 1

If local reference frames are not used to define a model, Records 3, 4 and 5 are not required. In this case, the input and analysis frame of all nodes defined by Records 6 and/or Records 7-12 is the GLOBAL reference frame. Direct definition of nodes is effected via Record 6, whereas Records 7-12 provide capabilities for defining nodes based on the geometries of components previously described via a GEOMETRY data set (sec. 126.0) or via GCS or NASA-LRCGEOM data (see appendix G).

Multiple, local frames can be defined by Record 3 placed anywhere within a nodal set. Nodal data input directly after a Record 3 are assumed to be defined relative to the local coordinate system defined by that Record 3. Nodal data input prior to the first Record 3 are assumed to be defined with respect to the GLOBAL coordinate system. Record 4 allows a reference frame previously-defined within a nodal set to be identified as the input frame until another Record 3 or 4 is encountered. The analysis frame of all nodes in a SFT is assumed to be the GLOBAL frame unless specified otherwise by Record 5.

Record 3 Definition of a Local Reference Frame (optional)

{ REC CYL SPH }	label Origin X-axis Z-axis
-----------------------------	----------------------------

{
REC
CYL
SPH
}

= A key-word denoting a rectangular Cartesian, cylindrical or spherical local reference frame, respectively.

label = An alphanumeric word of 1 to 7 characters that identifies the local coordinate system. The words GLOBAL and INPUT may not be used as labels. A frame identified by "label" may not be redefined.

Each of the items "Origin," "X-axis" and "Z-axis" denotes either the number (integer) of a previously-defined node or a list of three GLOBAL X, Y and Z coordinates (decimal numbers). Each item defines a point relative to the GLOBAL reference frame.

Origin = A point defining the origin of the local coordinate system.

X-axis = A point different from the one denoted by "Origin" which lies on the local, positive x-axis.

Z-axis = A point different from the ones denoted by "Origin" and "X-axis" which defines the local, positive z-axis. This point may be any point with a positive z-coordinate in the x-z plane.

Default Record: The input and analysis reference frame for all nodes is the GLOBAL frame.

Records of this type may appear any number of times anywhere within a nodal data set except between Records 7-12. All nodal data input directly after a Record 3 are defined relative to the local reference frame established by that Record 3.

Record 4 A Previously-Defined Reference Frame to be Used as the Input Frame (optional)

RESUME	{ Label } { GLOBAL }
--------	-------------------------

{ Label }
{ GLOBAL } = The "Label" assigned to a local coordinate system by Record 3 or the word GLOBAL. This name identifies the input frame of all nodal data defined by Record 6 immediately following this record.

Default Record: The input frame identified by the previous Record 3 or the GLOBAL frame.

Records of this type may appear any number of times anywhere within a nodal data set except between Records 7-12.

Record 5 Analysis Frame Identification (optional)

ANALYSIS FRAME	{ Label } { GLOBAL } { INPUT }
----------------	--------------------------------------

{ Label }
{ GLOBAL }
{ INPUT } = Either the "Label" assigned to a local coordinate system by Record 3, the word GLOBAL or the word INPUT. This name identifies the analysis frame of all nodal data defined by Record 6 and/or Records 7-12 immediately following this record. The key-word INPUT denotes that the analysis frame is the same as the input frame for these nodes.

Default Record: ANALYSIS FRAME GLOBAL

Records of this type may appear any number of times anywhere within a nodal data set. The analysis frame of all nodes input prior to the first record of this type is the GLOBAL frame. An analysis frame specified by this record remains effective until redefined by another Record 5.

04

146.1.1 Direct Definition of Nodes

Input lists of three or four nodal coordinates are denoted by the items "list1" and "list2" within the Record 6 format. The lists of input coordinates associated with the different types of reference frames are illustrated in figure 146-1. The "c4" coordinate for mid-surface nodes must be greater than or equal to zero. All angles associated with cylindrical and spherical coordinates are input in degrees (decimal numbers).

Type of Nodal Reference Frame	Sequence of Input Coordinates			
	c1	c2	c3	c4
GLOBAL	X	Y	Z	ΔZ
REC	x	y	z	Δz
CYL	r	θ	z	Δr
SPH	ρ	θ	ϕ	$\Delta \rho$

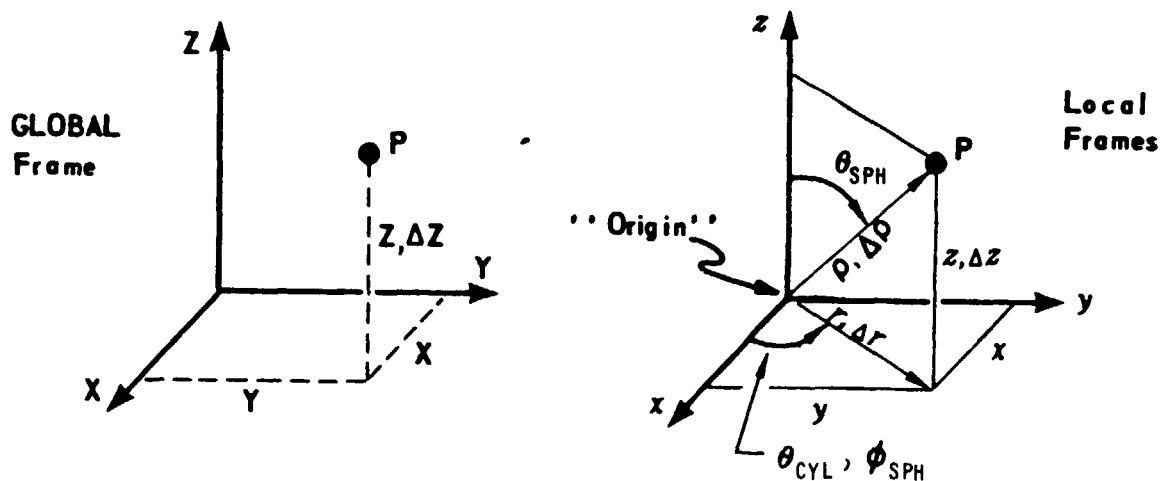


Figure 146-1. Nodal Reference Frame Coordinates

Node numbers and coordinates may be defined by any combination of the three input-record variations presented below. These variations provide the following capabilities:

- a) Explicit definition of a node;
- b) Generation of a sequence of equally-spaced nodes;
- c) Generation of a sequence of variably-spaced nodes;

The input-record formats for nodal data defined with respect to the GLOBAL or local coordinate systems are identical.

Record 6 Direct Definition of Nodes

Variation 1 node1 <list1>

- node1 = User node number (integer) in the range 1-99999, inclusive
- list1 = List of 3 or 4 coordinates (c1, c2 and c3 or c1, c2, c3 and c4) of "node1."
 Default: "node1" is assigned the GLOBAL coordinates of the origin of the last-specified input frame.

Variations 2 and 3 allow nodal data to be generated between two nodes identified by "node1" and "node2." Cylindrical and spherical nodal data are always generated in the positive sense of θ going from "node1" to "node2" (see fig. 146-1).

Variation 2 node1 <list1> TO node2 <list2>

- node1 }
 node2 } = Same as "node1" in Variation 1
- list1 }
 list2 } = Lists of 3 or 4 coordinates associated with "node1" and "node2," respectively. Each list must have the same number of items.
 Defaults: Previously-defined coordinates and analysis frames for "node1" and/or "node2" are retrieved automatically. The current and previously-used input frames for "node1" and "node2" may be different for 3-coordinate nodes, whereas they must be the same if mid-surface nodes are generated.

Equally-spaced nodes between "node1" and "node2" are generated. Node numbers are established by incrementing "node1" by ± 1 depending on whether "node1" or "node2" is larger. Coordinates of the generated nodes are obtained by linear interpolation between "list1" and list2." A maximum of 101 nodes may be defined by one record of this type.

Variation 3

node1 <list1> TO node2 <list2>
 <BY inc> <OF d1 d2 ... dn>

node1
 node2
 list1
 list2

= Same as defined for Variation 2

inc

= The node-number increment (integer) to be used to generate node numbers between "node1" and "node2." The quantity [(node2-node1)/inc] must be a positive integer ≤100.

Default: "inc" is automatically set to ±1 depending on whether "node1" or "node2" is larger.

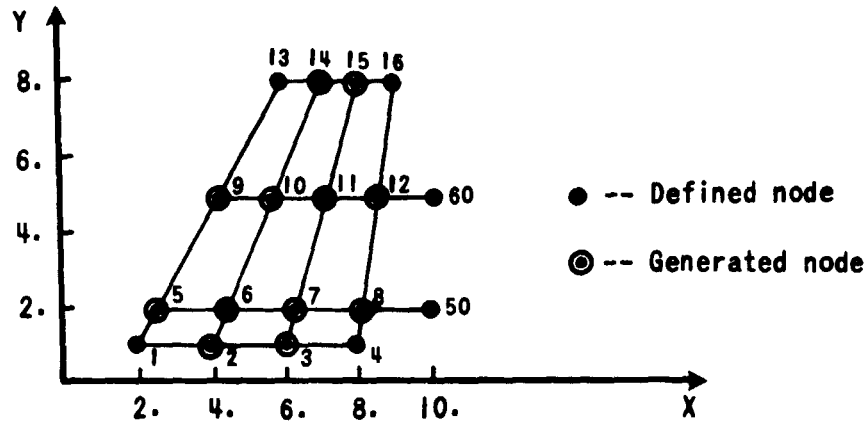
d1,d2,...,dn=

List of proportionate spacing distances between each pair of nodes in the sequence.

Default: Nodes generated from "node1" to "node2" are equally spaced via linear interpolation between "list1" and "list2."

Any combination of Records 3-5 and Record 6 may be repeated.

Examples: Use of the various formats of Record 6 are illustrated below.



1 2. 1. 0. TO 4 8. 1. 0. / EQUALLY-SPACED
 13 6. 8. 0. TO 16 9. 8. 0. / EQUALLY-SPACED
 1 TO 13 BY 4 OF 1. 3. 3. / VARIABLY-SPACED
 *+3 1 0 1 ** / VARIABLY-SPACED
 50 10. 2. 0. / EXPLICIT
 60 10. 5. 0. / EXPLICIT

146.1.2 Definition of Nodes Based on Geometry Components

Node numbers and coordinates may be defined by use of the component-geometry data input to the GEOMETRY Preprocessor (sec. 126.0) or by use of GCS or NASA-LRCGEOM component data (see appendix G). If extraction of coordinates from geometry components is not required, Records 7-12 should not be input.

Records 7-12 may be interspersed within a nodal data set to define all or some of the component surface nodes (3 coordinates) and mid-surface nodes (4 coordinates) used to define the model. The following rules must be satisfied:

- a) The data for a geometry component must be defined relative to a rectangular coordinate system (ref. sec. 126.0). The GLOBAL frame and the first 60 local frames defined by Record 3 may be used for these purposes. Nodal coordinates extracted from a geometry component are measured relative to the corresponding reference frame.
- b) The nodal input reference frame associated with a geometry component must be identified by the same name previously-assigned to the geometry component. This one-to-one correspondence identifies the nodes with a particular component. Orientation of a component relative to the total model is established by the origin and orientation of the input frame used for the component.
- c) The geometry data set that defines the component being interrogated must precede the nodal data set in the input stream. It should be noted that a particular geometry component may be associated with different nodal data sets identified by Record 2.

Record 7 Begin Coordinate Extraction from a Geometry Component

BEGIN EXTRACTION

This record indicates that subsequent nodal-data are to be used to extract nodal coordinates from the geometry component identified by Record 8.

The location of a node associated with a component is specified either by explicit definition of two of its x-y-z coordinates (Record 9) or by a plane which cuts the component surface (Records 10-11). Nodes may be positioned arbitrarily on the section contour contained in the cutting plane or they

may be located at points on the contour where the geometry, longitudinal control-curves intersect it. Surface and mid-surface nodes may be defined by the following variations:

- a) Definition of nodes by direct input of only 2 coordinates per node
- b) Definition of nodes located by the longitudinal control curves of a component
- c) Generation of a sequence of equally-spaced nodes located relative to pre-defined surface geometries
- d) Generation of a sequence of variably-spaced nodes located relative to pre-defined surface geometries.

Record 8 Geometry Component Identifier

COMPONENT Label

Label = An alphanumeric word of 1 to 7 characters that identifies the geometry component from which nodal coordinates are to be extracted. The geometry-component "Label" must be the same name that was assigned to a rectangular, nodal reference frame (GLOBAL or a "label" specified by Record 3).

Record 9 Node Definition/Generation Based on Input of 2 Coordinates

The basic format of this record is the same as Record 6. Any combination of the input variations of this record may be used to extract nodal coordinates from a geometry component. If this option is not required, Record 9 should not be input.

node1 list1 <TO node2 list2 <<BY inc>
<OF d1 d2 ... dn>>>

node1
node2
inc
d1,d2,...,dn } = Same as defined for Record 6

list1 } = Lists of 3 or 4 coordinates associated
list2 } with "node1" and "node2," respectively.
Each list must have the same number of items. The following rules must be followed:

A 3-item list used for surface nodes must contain 2 explicit x, y or z coordinates and the letter "G" denoting which coordinate is to be extracted from the component geometry. The coordinate corresponding to the longitudinal axis of the component must be input explicitly. The component surface must be single-valued at points where the "G-coordinate" is to be extracted.

A 4-item list used for mid-surface nodes must contain the x and y explicit coordinates followed by the letters "G G" denoting that the z and Δz coordinates are to be extracted.

This record or any combination of Record 5 and this record may be repeated.

Records 10 and 11 allow surface and mid-surface nodes to be defined/generated relative to selected cross-section contours of a component. The section cutting-plane is defined by Record 10 and the coordinate-extraction data are specified by subsequent Records 11. A section plane must be intersected by all the longitudinal control curves used to define the component.

The order in which the longitudinal control curves are input to the GEOMETRY Preprocessor establishes the positive direction along a section contour. SURFACE nodes are located according to distances measured along the section contour in this positive direction and relative to the ordered set of points created by the intersection of the control curves with the cutting plane.

MIDSURFACE nodes are defined/generated on a selected cross section by definition of a cutting plane that is perpendicular to the x-y plane. The intersection of the cutting plane with the x-y plane is referenced as the u-axis. MIDSURFACE nodes are located according to distances measured along the u-axis.

Record 10 Definition of Section Cutting Plane

$\left\{ \begin{array}{l} \underline{\text{SURFACE}} \\ \underline{\text{MIDSURFACE}} \end{array} \right\}$	NODES IN SECTION x1 y1 z1 x2 y2 z2 x3 y3 z3
---	---

$\left\{ \begin{array}{l} \underline{\text{SURFACE}} \\ \underline{\text{MIDSURFACE}} \end{array} \right\}$ = Key-word denoting that coordinates are to be extracted for SURFACE (3-coordinates) or MIDSURFACE (4-coordinates) nodes.

$\left. \begin{array}{l} x1,y1,z1 \\ x2,y2,z2 \\ x3,y3,z3 \end{array} \right\}$ = The x, y and z coordinates of 3 points located relative to the input reference frame such that they are external to the geometry component. These points establish the cutting plane.

If MIDSURFACE nodes are being defined, the cutting plane must be normal to the x-y plane. The general direction of the u-axis is from the point defined by (x1,y1,z1) toward the point defined by (x2,y2,z2).

Caution: All longitudinal control curves of the component must intersect the cutting plane.

Record 11 Node Definition/Generation Based on Longitudinal Control Curves of Component

<u>Variation 1</u>	node1 TO node2 <BY inc> $\left\langle \begin{array}{l} \text{OF } d1 \text{ } d2 \text{ } \dots \text{ } dn \\ \text{AT CURVES} \end{array} \right\rangle$
--------------------	--

$\left. \begin{array}{l} \text{node1} \\ \text{node2} \\ \text{inc} \end{array} \right\}$ = Same as defined for Record 6

$\left\langle \begin{array}{l} \text{OF } d1, d2, \dots, dn \\ \text{AT CURVES} \end{array} \right\rangle$ Locations of the generated nodes are based either on a list of proportionate spacing distances "di" between each pair of nodes in the sequence or they are pre-defined by the locations of the user-defined and program-generated (by enrichment) longitudinal control curves (AT CURVES). If proportionate distances are specified, SURFACE nodes are spaced along the section contour, whereas MIDSURFACE nodes are spaced according to distances along the u-axis.

Default: Equally-spaced nodes are generated from "node1" to "node2"

SURFACE nodes on an open cross-section are established by assigning "node1" and "node2" to the intersection points of the first and last longitudinal control curves, respectively. Examples of Record 11 for these cases are illustrated in figure 146-2.

If SURFACE nodes are generated for a closed section, "node1" is associated with the first longitudinal curve and the remaining nodes are located according to the selected option. MIDSURFACE nodes are generated by assigning "node1" and "node2" to the longitudinal-curve intersection points which have the minimum and maximum u-coordinates, respectively. Examples of Record 11 for these cases are illustrated in figure 146-3.

Variation 2 allows nodes to be generated between the points established by the intersection of user-selected longitudinal control curves with the cutting plane. This option is only applicable to components defined by the detailed-geometry option (see sec. 126.0).

Variation 2 node1 <Name1> TO node2 <Name2>
 <BY inc> <OF d1 d2 ... dn>
 <AT CURVES>

node1
 node2
 inc
 d1,d2,...,dn } = Same as defined for Variation 1

<Name1>
 <Name2> = The names of 2 different longitudinal control curves previously defined for the geometry component. "Name1" and "Name2" are used as the "first" and "last" control curves in the geometry interrogation. They must be associated with the i-th and the (i+n) curves in the pre-defined sequence of control curves.
 Defaults: "Name1" and "Name2" are the names of the first and last control curves, respectively, that are used to describe the component geometry.

Nodes are generated in the same manner as described for Variation 1 except that "Name1" and "Name2" correspond to the "first" and "last" longitudinal control-curve intersection points, respectively (see fig. 146.3). SURFACE nodes are located on the section contour according to distances measured along the curve from the point established by "Name1" to the point established by "Name2." MIDSURFACE nodes are spaced according to the u-coordinates that correspond to the control curves "Name1" through "Name2." "Node1" is assigned to the longitudinal-curve intersection point that has the least u-coordinate and "node2" is assigned to the point that has the largest u-coordinate.

This record or any combination of Record 5, Record 9 and Records 10 and 11 may be repeated.

Nodes may be defined for different geometry components by repeating Records 8-11.

Record 12 End Coordinate Extraction

END EXTRACTION

This record indicates that extraction of nodal coordinates for geometry components is complete and resets the current input frame and analysis frame to GLOBAL (ref. Records 4 and 5). Additional nodal data may be defined by repeating Records 3-5 and Record 6. Additional nodes may also be defined via geometry components by repeating Records 3-5 and Records 7-12.

Record 13 Node Reorder Specification (optional)

The processing efficiency of most ATLAS jobs is affected by the bandwidth of the gross stiffness matrix. The bandwidth represents the connectivity of the structural model relative to the internal node-number system established by ATLAS. In solving the structural matrix equations, the bandwidth of each row of the stiffness matrix is accounted for. Thus, one should strive to minimize the bandwidth in an overall sense to achieve processing efficiency. Record 13 offers alternate ways of achieving this by effecting the internal node-number system.

REORDER FROM Nn

Nn = Integer with the following options:

- 2--Nodes are assigned internal numbers according to the input sequence of the nodes.
- 1--Nodes are assigned internal numbers according to increasing user node-number order.
- 0--The node closest to the origin of the GLOBAL frame is assigned internal number 1. The remaining nodes are assigned internal numbers according to increasing distances from the origin.

Node-Number--If a node number is input for this item, it is assigned internal node number 1. The remaining nodes are assigned internal numbers according to increasing distances from this node.

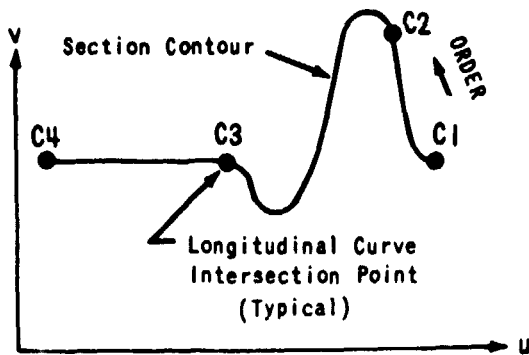
Default Record: REORDER FROM 0

Additional nodal data sets may be defined by repeating Records 2-13.

Record 14 End Nodal Data Set

END NODAL DATA

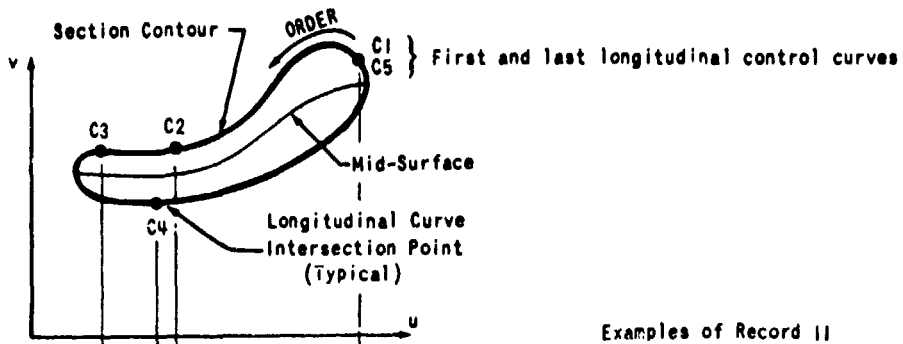
Additional nodal data sets may be input by repeating Records 1-14.



Examples of Record 11

	<p>1 TO 6 / EQUALLY-SPACED NODES</p>
	<p>1 TO 6 OF 1. 1. 1. 1. 2. /</p>
	<p>1 TO 4 AT CURVES /</p>

Figure 146-2. SURFACE Node Generation on an Open Section Contour



SURFACE NODES	
	<p>EQUALLY-SPACED NODES</p> <p>1 TO 6 /</p> <p>1 C1 TO 7 C5 /</p>
	<p>VARIABLY-SPACED NODES</p> <p>1 TO 6 OF 1. 1. 1. 1. 1. 5. /</p> <p>1 C1 TO 7 C5 OF 1. 1. 1. 1. 1. 5. /</p>
	<p>NODES LOCATED ON CONTROL CURVES</p> <p>1 TO 5 AT CURVES /</p> <p>1 C1 TO 4 C4 AT CURVES /</p>
MIDSURFACE NODES	
	<p>EQUALLY-SPACED NODES</p> <p>1 TO 6 /</p> <p>1 C3 TO 6 C5 /</p>
	<p>VARIABLY-SPACED NODES</p> <p>1 TO 6 OF 1. 1. 2. 1. 1. /</p> <p>1 C3 TO 6 C5 OF 1. 1. 2. 1. 1. /</p>
	<p>NODES LOCATED ON CONTROL CURVES</p> <p>1 TO 5 AT CURVES /</p> <p>1 C1 TO 4 C4 AT CURVES /</p>

Figure 146-3. SURFACE and MIDSURFACE Node Generation on a Closed Section Contour

Table 146-1. Summary of Nodal Data Records

Reference Page	Data Records
146.3	BEGIN NODAL DATA
146.3	SET n
146.4	{ REC CYL SPH } label Origin X-axis Z-axis
146.5	RESUME { Label GLOBAL }
146.5	ANALYSIS FRAME { Label GLOBAL INPUT }
146.7	node1 <list1><T0 node2 <list2> <<BY inc><OF d1 d2 ...>>
146.9	BEGIN <u>EXTRACTION</u>
146.10	COMPONENT Label
146.10	node1 list1 <T0 node2 list2 <<BY inc><OF d1 d2 ...>>
146.12	{ SURFACE MIDSURFACE } NODES IN SECTION x1 y1 z1 x2 y2 z2 x3 y3 z3
146.12	node1 T0 node2 <BY inc> <OF d1 d2 ...> AT CURVES
146.13	node1 <Name1> T0 node2 <Name2><BY inc> <OF d1 d2 ...> AT CURVES
146.14	END <u>EXTRACTION</u>
146.15	REORDER FROM Nn
146.15	END NODAL DATA

150.0 RHO3 AERODYNAMICS DATA

Unsteady aerodynamic analyses of a rigid or elastic, planar wing with or without one to four trailing-edge control surfaces in subsonic compressible flow can be performed by the solution technique employed by the RHO3 Processor (sec. 250.0). The solution technique employed by this processor is published in reference 150-1. The input data set supporting RHO3 aerodynamic analyses is comprised of the following data subsets:

- a) Geometry -- Aerodynamic surface planform and downwash points.
- b) Modal -- Vibration modes or "AIC" mode substitutes (see sec. 232.0) associated with the aerodynamic surfaces.
- c) Option -- Locations on lifting surface planform at which sectional generalized forces and aerodynamic pressures (pressure difference between upper and lower surfaces) are to be calculated; velocity profile modifications to account for airfoil-thickness aerodynamic effects.

If only the aerodynamic-pressure/downwash-kernel influence coefficient matrices (C-matrices) as discussed in section 250.2 are to be calculated, only the Geometry subset is required. This data set need not be preceded by any other data set within a data deck.

Each set of aerodynamics data is referenced as a CASE. A maximum of 36 RHO3 data cases may be identified per job. Printout of the input aerodynamic case data may be requested from the RHO3 Postprocessor (sec. 250.1).

PREVIOUS PAGE BLANK NOT FILMED

150.1 INPUT DATA

Record 1 Data Set Identification

BEGIN RHO3 DATA

This record initiates execution of the RHO3 Preprocessor.

Record 2 Data Case Number (optional)

Each set of RHO3 aerodynamics data is referenced as a case which is identified by an integer number. If multiple sets are used in a job, each case must be assigned a unique number.

CASE n

n = Data case number (integer) in the range 1 to 36, inclusive.

Default Record: CASE 1

150.1.1 Geometry Data Subset

All geometrical information is defined by this data subset which includes the following:

- a) Planform definition of a main lifting surface with or without trailing-edge control surfaces.
- b) Aerodynamic downwash (control) points on the surface.

Geometry data are defined relative to a coordinate system which is a rectangular reference frame x-y-z oriented so that the x-axis is in the aft direction (parallel to the free stream direction) and the positive y-axis is directed from the root to the tip of the planform. The aerodynamic surface must be planar and must be symmetrical about the x-z plane. If modes calculated by the VIBRATION Processor are used in the aerodynamic analysis, the planform coordinate system must coincide with the analysis frame associated with the retained nodal freedoms (see sec. 150.1.2).

Geometrical data are input only for half the total lifting surface due to the required symmetry. Each surface is defined via straight line segments. Up to four, separate trailing-edge control surfaces may be located along the half-span of a main surface. Chordwise edges of all control surfaces must be parallel to the free stream (parallel to the x-axis). The configuration of a control surface may be full span, tip span,

mid span or partial span. The outboard edge of a full-span or a tip-span control coincides with the tip-most chord of the main surface, whereas the inboard edge of a full-span or a mid-span control is at the root of the main surface ($y=0$). Multiple control surfaces may share common streamwise edges but they should not overlap. One control surface may be attached to another control surface as illustrated by the sample aerodynamic model shown by figure 150-1.

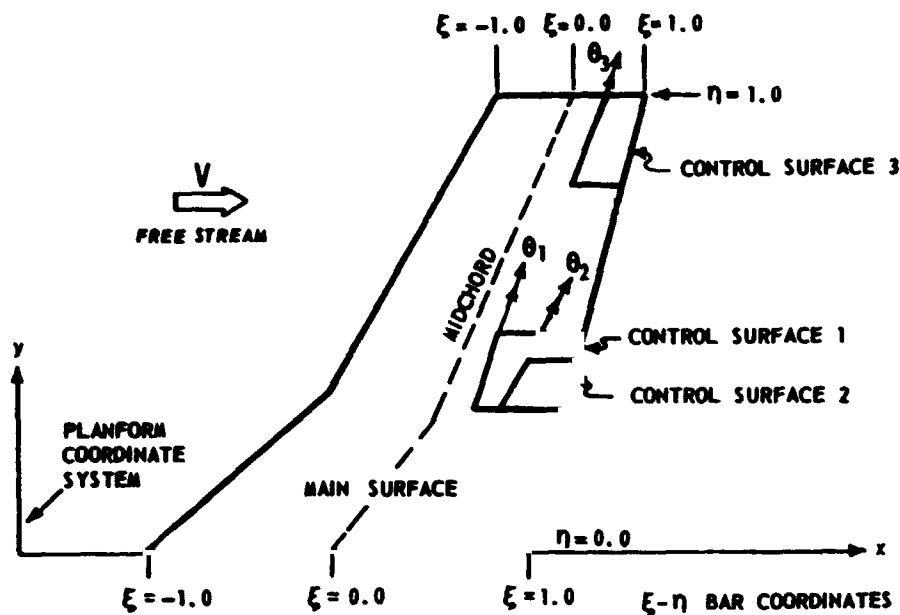


Figure 150-1. Sample RH03 Aerodynamic Model

Record 3 Begin Geometry Subset

BEGIN GEOMETRY DATA

Record 4 Main Surface Identification (optional)

MAIN SURFACE ident

ident = Alphanumeric word of 1 to 7 characters which identifies the main surface associated with the case specified by Record 2.

Default Record: No user identification except the case number is associated with the main surface.

Records 5 and 6 define the leading and trailing edges, respectively, of the main surface. Each edge is defined by a list of x and y-coordinates of 2 to 10 points input in a root-to-tip order such that the y-coordinates are strictly increasing. The y-coordinate of the inboard-most points of the leading edge and the trailing edge must be identical. Additionally, the y-coordinate of the tip-most points of these edges must be identical.

Record 5 Leading Edge Coordinates

LEADING EDGE x1 y1 x2 y2 ...

x_i, y_i = The x and y planform coordinates of the i-th point defining the main surface leading edge.

Record 6 Trailing Edge Coordinates

TRAILING EDGE x1 y1 x2 y2 ...

x_i, y_i = The x and y planform coordinates of the i-th point defining the main surface trailing edge.

Control surfaces along the trailing edge of a main surface are defined by Record 7. The spanwise length of each control surface is defined via the inboard and outboard locations of its hinge line. The leading edge of a control surface must coincide with its hinge line.

The unsteady aerodynamic pressure on a control surface is dependent on the difference of streamwise slopes of the control surface and the surface to which it is attached. Hinge line rotations associated with each analysis mode shape are automatically calculated by the RHO3 Processor based on the mod interpolation function coefficients for the two surfaces as defined by Record 11. The rotation at any point on a hinge line is derived from a cubic function of y by fitting a curve to calculated rotations at four equidistant points along the hinge line. The user may, however, define explicit hinge line rotation expressions to be associated with selected modes. These rotations are specified via Record 7 and are used in lieu of the automatically calculated values.

Record 7 Control Surface Definition (optional)

```

CONTROL SURFACE ident HINGE xi yi xo yo
<MODE m1 c10 c11 c12 c13
  MODE m2 c20 c21 c22 c23
  .
  .
  .
  MODE mn cno cn1 cn2 cn3>

```

ident = Alphanumeric word of 1 to 7 characters which identifies a trailing-edge control surface associated with the main surface.

$\left. \begin{array}{l} xi, yi \\ xo, yo \end{array} \right\}$ = Two pairs of planform x and y coordinates defining the inboard and outboard limits of the control-surface hinge line, respectively.

The remaining items in this record allow hinge line rotation modes to be specified explicitly for the control surface. Each rotation is defined via the coefficients of a cubic equation which is to be associated with the i-th sequential vibration mode used in the aerodynamic analysis. The i-th mode number is defined either by Variation 2 of Record 11 or by the subset of ordered modes which is extracted by the VIBRATION Processor and is associated with the interpolation coefficient matrix specified by Variation 1 of Record 11.

mj = The j-th integer in this record corresponds to the i-th mode number for which a hinge line rotation is defined explicitly. Each hinge line rotation mode is input by the key-word MODE followed directly by "mj" and four decimal numbers "cjk."

$\left. \begin{array}{l} cj0, cj1 \\ cj2, cj3 \end{array} \right\}$ = The j-th list of four decimal numbers in this record which are the coefficients of the cubic equation defining the control-surface hinge-line rotation to be associated with the mode number "mj." In this equation, y denotes the location of a point measured in fraction of control span, inboard to outboard.

$$\theta(j) = c_{j0} + c_{j1}y + c_{j2}y^2 + c_{j3}y^3$$

Default Items: Hinge line rotations not defined explicitly for any of the modes used in an analysis are calculated automatically as discussed above.

This record is repeated to define 1 to 4 control surfaces associated with the main surface defined via Records 4-6.

The integral equation defining the effective angle of attack of a lifting surface due to a vibration mode displacement is evaluated at a number of discrete points on the surface. These points are located by the user via Record 8 and are referred to as downwash (control) points. The user may request a grid of downwash points to be generated automatically (Variation 1) or may define the grid explicitly (Variation 2).

Record 8 Downwash Points

Only one of the following two variations may be used per data case.

Variation 1 DOWNWASH COSINE nchords npoints

Downwash point locations are generated automatically via a cosine distribution in both the chordwise and spanwise directions.

nchords = The number of chords (integer) on which downwash points are to be located ($2 \leq nchords \leq 9$). The spanwise location of the i-th downwash chord is $y(i) = \cos [i\pi / (1+2*nchords)] * (\text{Semi-span})$.

npoints = The number of points (integer) to be generated per downwash chord ($2 \leq npoints \leq 8$). The streamwise distance from the midchord of the i-th point on each planform chord is $x(i) = -\cos [2i\pi / (1+2*nchords)] * (\text{Semi-chord})$.

Variation 2 DOWNWASH <BAR>
 CHORD y1 x11 x12 ... x1n
 CHORD y2 x21 x22 ... x2n
 .
 .
 .
 CHORD ym xm1 xm2 ... xmn

Downwash points are located explicitly by coordinates defined relative to the planform coordinate system or relative to a non-dimensionalized (BAR) planform coordinate system. Spanwise BAR coordinates range from 0.0 to 1.0 at the planform root and tip, respectively. Chordwise BAR coordinates are measured relative to the local semi-chord ranging from -1.0 to 1.0 at the planform chord leading edge and trailing edge, respectively (see fig. 150-1).

- BAR = Key-word denoting that subsequent coordinates input by this record are dimensionless (BAR) planform coordinates. Default: The input coordinates are assumed to be with respect to the planform coordinate system.
- y_i = Spanwise coordinate of the i -th downwash chord ($2 \leq m \leq 9$). A downwash chord should not be located at the planform tip and should not coincide with an edge of a control surface. Additionally, it should not be located at a planform leading-edge or trailing-edge discontinuity.
- x_{ij} = Streamwise coordinate of the j -th downwash point on the i -th chord ($2 \leq n \leq 8$). Downwash points may not coincide. The number of points specified for each chord must be identical. A downwash point should not be placed on the planform leading edge or trailing edge or on a control surface hinge line.

Record 9 End Geometry Subset

END GEOMETRY DATA

This record indicates all geometry data have been defined for the case identified by Record 2.

150.1.2 Modal Data Subset (optional)

This subset defines the vibration modes, "AIC" mode substitutes or rigid-body modes to be associated with each aerodynamic surface. If only the aerodynamic C-matrices are to be calculated, this data subset should not be input.

The unsteady aerodynamic pressure distribution is approximated by a linear combination of discrete pressure modes which are functionally dependent on the modal amplitudes at the downwash points. The aerodynamic theory requires translational components of the vibration modes to be normal to the aerodynamic surface(s). Thus, the analysis frame axis associated with all the retained nodal freedoms must be normal to the surface.

Record 10 Begin Modal Subset

BEGIN MODAL DATA

Record 11 defines the vibration modes or "AIC" mode substitutes that are to be used in the analysis of the case identified by Record 2. Only one of the two variations of this record may be used per data case. Variation 1 must be used when mode shape coefficients generated by the INTERPOLATION Processor are to be employed in the aerodynamic analysis. Mode shapes may be defined directly via Variation 2 for a rigid-body aerodynamic analysis.

Record 11 Modal Data

Variation 1

USE Name WITH { MAIN SURFACE
CONTROL SURFACE Ident }

Name = The name of an interpolation-coefficient matrix previously generated via the INTERPOLATION Processor (see sec. 232.0). It should be noted that this matrix must be associated with a subset of nodes, all of which have the same analysis frame.

{ MAIN SURFACE
CONTROL SURFACE Ident } = Key-word denoting whether "Name" is to be associated with the MAIN SURFACE or a CONTROL SURFACE previously identified by "Ident" in Record 4 or 7.

A record of this type must be input for the main surface and once for each control surface.

Variation 2 allows the user to define 1 to 6 rigid body modal freedoms relative to the planform coordinate system.

Variation 2 RIGID BODY <x y z> Rbf1 mag1 Rbf2 mag2 ...

- x,y,z** = x, y and z coordinates defining the reference point to be used in generating the rigid body modes.
Default: The origin of the planform coordinate system is used as the reference point.
- Rbfi** = Key-word denoting the i-th rigid-body freedom selected from the list TX, TY, TZ, RX, RY and RZ. Within each of these key-words, the letter "T" is associated with a translation freedom and "R" is associated with a rotation freedom relative to the planform x-y-z triad.
- magi** = The relative magnitude of the i-th rigid body mode with respect to the reference point. Items "Rbfi" and "magi" are input in pairs 1 to 6 times in an order which defines the sequence of rigid body modes used in the aerodynamic analysis.

Record 12 End Modal Subset

END MODAL DATA

This record indicates all modal data have been defined for the case identified by Record 2.

150.1.3 Option Data Subset (optional)

This input data subset contains the following types of information:

- a) Data case title
- b) Locations of planform chords along which sectional generalized forces are to be calculated.
- c) A grid of points on the surface planform at which unsteady aerodynamic pressures are to be evaluated.

- d) Velocity profile modification data to account for airfoil-thickness aerodynamic effects.

Record 13 Begin Option Subset

BEGIN OPTION DATA

Record 14 Data Case Title (optional)

A title provides additional user-identification of a data case. This title and the case number are printed within the page headings of RHO3 printout (see sec. 250.0).

LABEL label

label = Alphanumeric text of 1 to 75 characters including embedded blanks.

Default Record: RHO3 data printout is identified only by the data case number.

Record 15 Sectional Generalized Forces (optional)

Calculation of generalized forces along selected planform chords may be requested via this record. The selected chord locations may be a default set (Variation 1) or may be specified explicitly (Variation 2). Only one of the two variations may be used per data case. Sectional generalized forces are calculated only if one or more of these records is input.

Variation 1 SECTIONAL FORCES

This record indicates generalized air forces are to be calculated along chords positioned by the following 11 fractions of semi-span:

.01 .1 .2 .3 .4 .5 .6 .7 .8 .9 and .99

Variation 2 SECTIONAL FORCES <BAR> y1 y2 ...

BAR = Key-word denoting the "yi" items in this record are in units of fraction semi-span.
Default: Each y-coordinate (yi) is measured relative to the planform coordinate system.

yi = Spanwise planform coordinate or fraction of semi-span (if BAR is input) locating the i-th chord for sectional generalized

force calculation. A sectional force chord should not coincide with the edge of a control surface or the tip of the main surface.

Record 16 Pressure Report Points (optional)

Calculation of unsteady aerodynamic pressures at selected planform points may be requested via this record. The selected point locations may be a default set (Variation 1) or may be specified explicitly (Variation 2). Only one of the two variations may be used per data case. Pressure sampling data are calculated only if one or more of these records is input.

Variation 1

PRESSURE REPORT

This record indicates aerodynamic pressures are to be calculated at the 231 points defined by the following dimensionless planform coordinate grid: 11 chords located at .01 .1 .2 .3 .4 .5 .6 .7 .8 .9 and .99 fraction of semi-span and 21 points per chord located at the following local chord (BAR) coordinates:

```
-.99 -.9 -.8 -.7 -.6 -.5 -.4 -.3 -.2 -.1 0.0
 .1 .2 .3 .4 .5 .6 .7 .8 .9 .99
```

Variation 2

```
PRESSURE REPORT <BAR>
CHORD y1 x11 x12 ... x1n
CHORD y2 x21 x22 ... x2n
.
.
.
CHORD ym xm1 xm2 ... xmn
```

Pressure report points are located explicitly by definition of either planform x and y coordinates or dimensionless (BAR) coordinates.

- BAK = Same as defined for Record 8, Variation 2.
- yi = Spanwise coordinate of the i-th pressure-report chord ($1 \leq m \leq 11$). A pressure-report chord should not coincide with an edge of a control surface or the tip of the main surface.
- xij = Streamwise coordinate of the j-th pressure-report point on the i-th chord ($1 \leq n \leq 21$). The number of points

specified for each chord must be identical. A pressure-report point should not be located on a control surface hinge line or on the leading edge or trailing edge of the main surface.

Record 17 Velocity Profile Modification Data (optional)

Modifications to the calculated velocity profile to account for airfoil-thickness aerodynamic effects are defined by this record. A velocity profile is defined as the variation of V_{LOCAL}/V versus the local chord length where V_{LOCAL} is the local air-flow velocity and V is the free-stream velocity. The first or second derivative of the profile at the planform leading and/or trailing edge may optionally be specified in addition to velocity profile values at selected points on all of the planform chords.

<p>VELOCITY PROFILE \langleDLE1 dle\rangle \langleDTE1 dte\rangle \langleDLE2 dle\rangle \langleDTE2 dte\rangle</p> <p>x1 v1 x2 v2 ... xn vn</p>

\langle DLE1,dle \rangle
 \langle DLE2,dle \rangle = Key-word DLE1 or DLE2 denoting whether the first or the second velocity-profile edge is to assume the value specified by "dle." Only one of the derivatives at the leading edge may be specified. Default: Derivatives not specified are calculated automatically.

\langle DTE1,dte \rangle
 \langle DTE2,dte \rangle = Key-word DTE1 or DTE2 denoting whether the first or the second velocity-profile derivative, respectively, at the trailing edge is to assume the value specified by "dte." Only one of the derivatives at the trailing edge may be specified. Default: Derivatives not specified are calculated automatically.

xi = The fraction of the chord, $0.0 \leq xi \leq 1.0$, at which "vi" is defined.

vi = Value of the velocity profile (V_{LOCAL}/V) at "xi." Items "xi" and "vi" are input in pairs where ($3 \leq n \leq 26$) such that the "xi" values are strictly increasing.

Default Record: No modifications are made to the
calculated velocity profile.

Record 18 End Option Subset

END OPTION DATA

Additional cases may be defined by repeating records 2-
18 with a unique identification number assigned to each
case by Record 2.

Record 19 End RHO3 Data Set

END RHO3 DATA

Additional cases may be defined by repeating Records 1-
19.

Table 150-1. Summary of RHO3 Data Records

Reference Page	Data Records
150.2	BEGIN RHO3 DATA
150.2	CASE n
150.3	BEGIN GEOMETRY DATA
150.3	MAIN <u>SURFACE</u> ident
150.4	<u>LEADING EDGE</u> x1 y1 x2 y2 ...
150.4	<u>TRAILING EDGE</u> x1 y1 x2 y2 ...
150.5	<u>CONTROL SURFACE</u> ident <u>HINGE</u> xi yi xo yo <MODE m1 c10 c11 c12 c13 ...>
150.6	<u>DOWNWASH COSINE</u> nchords npoints
150.6	<u>DOWNWASH</u> <BAR> CHORD y1 x11 x12...x1n CHORD ...
150.7	END GEOMETRY DATA
150.8	BEGIN MODAL DATA
150.8	USE Name WITH { <u>MAIN SURFACE</u> <u>CONTROL SURFACE</u> Ident }
150.9	RIGID BODY <x y z> Rbf1 mag1 Rbf2 mag2 ...
150.9	END MODAL DATA
150.10	BEGIN OPTION DATA
150.10	LABEL label
150.10	<u>SECTIONAL FORCES</u>
150.10	<u>SECTIONAL FORCES</u> <BAR> y1 y2 ...
150.11	<u>PRESSURE REPORT</u>
150.11	<u>PRESSURE REPORT</u> <BAR> CHORD y1 x11 x12 ...x1n CHORD ...
150.12	<u>VELOCITY PROFILE</u> <DLE1 d1e> <DTE1 dte> x1 v1 x2 v2 ... <DLE2 d1e> <DTE2 dte>
150.13	END OPTION DATA
150.13	END RHO3 DATA

152.0 STIFFNESS DATA

Stiffness data include all finite element descriptors necessary for generation of finite-element stiffness and stress matrices by the STIFFNESS Processor (sec. 252.0). These data, in addition to the nodal input data (sec. 146.0), establish a structural model that is interrogated by many of the other modules. Thus, the nodal and stiffness data sets are normally the first data sets within a data deck. Only the material data set (sec. 140.0), if required, must be input before this data set.

The stiffness data set is divided into the following data subsets:

- a) Property -- Element section properties
- b) Element -- Element type, material, nodes and properties

The Property subset, if used, must be input before the Element data subset.

A stiffness data set defines the structural finite elements for one or more structural models. Each structural model is identified by a SET number. The geometrical data associated with a SET are defined by nodal data (sec. 146.0) identified by the same SET number. A maximum of 36 node/stiffness data SETS (structural models) may be defined per job. Each set may contain a maximum of 32,767 elements.

Printout of the element data may be requested from the STIFFNESS Postprocessor (sec. 252.0). Plots of these data can also be effected via the EXTRACT and GRAPHICS Postprocessors as discussed in sections 218.0 and 228.0.

PREVIOUS PAGE BLANK NOT FILMED

152.1 INPUT DATA

Record 1 Data Set Identification

BEGIN STIFFNESS DATA

This record initiates execution of the STIFFNESS Preprocessor.

Record 2 Data Set Number (optional)

Each set of stiffness data (a structural model) is identified by an integer number. If multiple sets are used in a particular job, each set must be assigned a different number by this record. Multiple sets are defined by repeating either Records 2-8 or Records 1-9.

SET n

n = Data set number (integer) in the range 1 to 36, inclusive.

Default Record: SET 1

152.1.1 Property Data Subset (optional)

This data subset allows a list of element section-property values to be identified by a property code and optionally, by an alphanumeric title. Both identifiers may be printed when element printout is requested (sec. 252.0). A property code is used in lieu of the "Plist" items in Record 7. This option may be used for any type of stiffness element except the CPLATE and CCCVER composite elements.

Record 3 Begin Property Subset

BEGIN PROPERTY DATA

Record 4 Section-Property Reference Code

Pcode <*/ text */ Plist

Pcode = Section-property reference code comprised of the letter "P" followed by an integer in the range 1-999 (e.g., P6, P382). This code may be used in the element-definition data (Record 7) to identify element section-property values. Unnecessarily large "Pcode" integers should be avoided.

- text = Alphanumeric text of 1 to 30 characters, including blanks, to be associated with "Pcode."
Default: No text identifier.
- Plist = List of element section-property values. See the discussion of the item "Plist" in Record 7 regarding the input sequence, number and type of property values required for each element type.

This record may be repeated to define a maximum of 999 "P" codes for the data set identified by Record 2. A particular property code may not be redefined.

Record 5 End Property Subset

END PROPERTY DATA

152.1.2 Element Data Subset

All stiffness elements required to describe a structural model are defined by this data subset. Detailed information on the structural elements available in the ATLAS library is presented in appendix B.

Record 6 Begin Element Subset

BEGIN ELEMENT DATA

Element data may be defined by any combination of the three variations of Record 7 presented below. These variations provide the following capabilities:

- 1) Explicit definition of an element;
- 2) Generation of a sequence of elements with node-definition lists and user element-numbers incremented by one;
- 3) Generation of a sequence of elements with node-definition lists and user element-numbers incremented as specified.

Records of this type are repeated to define all of the stiffness elements for a structural model.

Record 7 Element Data

Variation 1 Eltype <Mcode> <Tcode> <Userid1> Nlist1 Plist

Eltype = A key-word (or integer-equivalent) that identifies the type of element defined by this record. The options for "Eltype" are:

<u>Key-word</u>	or	<u>Integer-Equivalent</u>
ROD		1
BEAM		2
SPAR		3
COVER		4
PLATE		5
GPLATE		6
BRICK		8
SCALAR		9
SROD		10
SPLATE		11
CPLATE		12
CCOVER		13

Mcode = Material reference code (sec. 140.0) that identifies the element material-property values. The letter "Z" may be used in place of the letter "M" or "C" in a material reference code to indicate zero density. This form is useful for structural elements for which no weight is to be calculated. If Special or Composite materials are required, the material data must precede the stiffness data in the input stream. Default: The "C" code with the smallest integer component for CPLATE or CCOVER elements. For other element types, the "M" code specified by the last record of this type or material code M1.

Tcode = Temperature code of the form T followed by a signed or unsigned integer (e.g., T100, T+80, T-10) which denotes the temperature at which material properties are to be evaluated. This code denotes a reference temperature for evaluation of CPLATE and CCOVER element properties (see description of Plist).

Caution: The specified temperature must be within the temperature range

defined for "Mcode."

Default: The temperature specified by the last record of this type or T70.

Userid1 = The letter N followed by the user element number (integer in the range 1-32767).
Default: The user element-number is the input sequence number of the element within this data subset.

Nlist1 = List of user node numbers (integers) that define the element. Nodal data for the SET identified by Record 2 are input as described in section 146.0. The number of nodes that may be used to define each element type is as follows:

<u>Element</u>	<u>No. of Nodes</u>
ROD	2
BEAM	2, 3, 4 or 5
SPAR	2
COVER	3 or 4
PLATE	3, 4, 6 or 8
GPLATE	3, 4, 5, 6, 8 or 9
BRICK	8, 11, 20, 23, 32, 35, 44 or 47
SCALAR	1, 2 or 3
SROD	3
SPLATE	4 or 8
CPLATE	3, 4, 6 or 8
CCOVER	3 or 4

See appendix B for detailed information on the node input required for each type of element.

Plist = Either a "Pcode" property code defined by Record 4 or a list of element property values. The number, input sequence and type of property values that may be used to define each type of element are summarized in table 152-1 and are presented in detail in appendix B. A variety of input options is provided. For example, 8 options are provided for BEAM elements and 11 options are provided for GPLATE elements.

Each "Plist" property value is input as a decimal number for all element types

except for the composite CPLATE and CCOVER elements. These elements are comprised of plates (laminates) each of which is defined by 1 to 10 laminas. Each lamina consists of a number of layers of composite material with fibers oriented in a prescribed direction.

For CPLATE and CCOVER elements, "Plist" is comprised of a laminate reference angle followed by 1 to 4 data items per lamina. The values for "Plist" are:

Aref--Laminate reference angle (decimal) in degrees measured from the element x-axis where $-360.0 \leq \text{Aref} \leq 360.0$ in increments of 0.1

Axxx.x--Alphanumeric word comprised of the letter A followed by a decimal number that defines the fiber direction of a lamina relative to "Aref." This angle (degrees) must be $-360.0 \leq \text{xxx.x} \leq 360.0$ with an accuracy of 0.1

Txxxx--Alphanumeric word comprised of the letter T followed by an integer number that defines the lamina temperature-deviation from the reference temperature established by "Tcode." The allowable range is $-2000 \leq \text{xxxx} \leq 4095$
Default: The temperature deviation specified for the last lamina or T0

Lnum--Alphanumeric word comprised of the letter L followed by the integer number of layers of composite material in the lamina where $1 \leq \text{Lnum} \leq 4095$
Default: The number of layers specified for the last lamina or L100

Ccode--Composite material reference code (sec. 140.0) that identifies the material property values for the lamina.
Default: The "C" code specified for the last lamina or the composite material identified by "Mcode."

Axxx.x, Txxxx, Lnum and Ccode are repeated to define a maximum of 10 laminas per

laminate. The first laminate defined for a CCOVER is referenced as the upper plate (see appendix B). The lower plate for a CCOVER is defined by repeating the data items "Aref" through "Ccode." If a lower plate is not required, these items should not be input. If, however, only a lower plate is required for a CCOVER, "Aref" must be input twice followed by the lamina definitions.

Example:

```
CPLATE T70 101 102 103 104 50.2
A45. T-30. L10 C3 A-45. T40 L15 A0. A90. /
```

This record defines a composite plate (laminate) with the following properties. The fiber direction is measured relative to the x-axis of the element reference frame.

<u>Lamina No.</u>	<u>Fiber Direction</u>	<u>Temp.</u>	<u>Layers</u>	<u>Ccode</u>
1	95.2	40.0	10	C3
2	5.2	110.0	15	C3
3	50.2	110.0	15	C3
4	140.2	110.0	15	C3

Variations 2 and 3 allow sequences of elements to be generated between two elements. All elements in a sequence are of the same type and have the same temperature, material and section-property values. The user element-numbers and node-numbers are generated by incrementing the first list until the second list is reached. Each of the items denoted by "Userid1" and "Nlist1", when incremented by +1, 0, -1, or the specified increments, must reach the corresponding items denoted by "Userid2" and "Nlist2" simultaneously. A maximum of 100 elements may be defined by one record of this type.

```
Variation 2 Eltype <Mcode> <Tcode> <Userid1> Nlist1 Plist
TO <Userid2> Nlist2
```

Items 1-6 are the same as defined for Variation 1

Userid2 = Number of the last element in the sequence. This item is defined in the same manner as "Userid1." It is required only if "Userid1" is specified.

Nlist2 = Node numbers (integers) defining the last element in the sequence. The number of items in "Nlist1" and "Nlist2" must be equal.

Variation 3 Eltype <Mcode> <Tcode> <Userid1> Nlist1 Plist
TO <Userid2> Nlist2 BY <Inc1> inc2

Items 1-9 are the same as defined for Variation 2

Inc1 = The letter N followed by a signed or unsigned integer (e.g., N5, N+10, N-8) which denotes the user element number increment. This item is required only if "Userid1" and "Userid2" are specified.

inc2 = List of positive and/or negative node-number increments (integers) to be used with "Nlist1." The number of items in "Nlist1", "Nlist2" and "inc2" must be equal.

Example: A record to generate a set of ROD elements with input-sequence number identifiers is:

RCD 1 9 5.6 TO 7 6 BY 2 -1 /

This record is equivalent to the following four records:

ROD 1 9 5.6 /
RCD 3 8 5.6 /
ROD 5 7 5.6 /
RCD 7 6 5.6 /

If user element numbers 10 through 13 are desired, the single record would assume the form:

RCD N10 1 9 5.6 TO N13 7 6 BY N1 2 -1 /

Record 8 End Element Subset

END ELEMENT DATA

Additional stiffness data sets may be defined by repeating records 2-8.

Record 9 End Stiffness Data Set

END STIFFNESS DATA

Additional stiffness data sets may be defined by repeating Records 1-9.

Table 152-1. Input Variations of Stiffness - Element Properties

ELEMENT TYPE		ELEMENT PROPERTIES (ref. appendix B)		INPUT PROPERTY EXPANSION KEYS (1)													
NAME	NO.	LABEL	DESCRIPTION	NUMBER OF INPUT PROPERTY VALUES													
				1	2	3	4	5	6	7	8	9	10	11	12	13	
ROD	1	A(1)	Cross section area at end(1)	1	1												
		A(2)	Cross section area at end(2)	1	2												
BEAM	2	A(1)	Cross section area at end(1)	1	1	1	1		1	1					1	1	
		A-VY(1)	Shear area (local y-dir) at end(1)	0	0	0	0		2	2					2	2	
		A-VZ(1)	Shear area (local z-dir) at end(1)	0	0	0	0		3	3					3	3	
		J(1)	St. Venant's torsion constant at end(1)	0	0	0	0		4	4					4	4	
		IY(1)	Mom. of inertia(oc.y-axis) at end(1)	0	0	2	2		5	5					5	5	
		IZ(1)	Mom. of inertia(oc.z-axis) at end(1)	0	2	3	3		6	6					6	6	
		A(2)		1	1	1	1		1	1					1	1	
		A-VY(2)		0	0	0	0		2	2					2	2	
		A-VZ(2)		0	0	0	0		3	3					3	3	
		J(2)		0	0	0	0		4	4					4	4	
		IY(2)		0	0	2	2		5	5					5	5	
		IZ(2)		0	2	3	3		6	6					6	6	
		CONSTR	End-restraint indicator	0	0	0	4		0	7					0	13	
SPAR	3	T-WEB	Web thickness	1	1	1		1	1	1		1	1	1		1	
		FAREA1U	Upper flange area at end(1)	0	2	2		2	2	2		2	2	2		2	
		FAREA1L	Lower flange area at end(1)	0	2	2		3	3	3		3	3	3		3	
		FAREA2U	Upper flange area at end(2)	0	2	2		2	2	2		4	4	4		4	
		FAREA2L	Lower flange area at end(2)	0	2	2		3	3	3		5	5	5		5	
		O(1)U	Upper offset at end(1)	0	0	3		4	4	4		6	6	6		6	
		O(1)L	Lower offset at end(1)	0	0	3		5	5	5		7	7	7		7	
		O(2)U	Upper offset at end(2)	0	0	3		4	4	4		8	8	8		8	
		O(2)L	Lower offset at end(2)	0	0	3		5	5	5		9	9	9		9	
		A-LMP1U	Upper lumping factor at end(1)	*	*	*		*	6	6		*	10	10		10	
		A-LMP1L	Lower lumping factor at end(1)	*	*	*		*	6	7		*	10	11		11	
		A-LMP2U	Upper lumping factor at end(2)	*	*	*		*	6	6		*	10	10		12	
		A-LMP2L	Lower lumping factor at end(2)	*	*	*		*	6	7		*	10	11		13	
COVER	4	T(0)U	Upper plate thickness	1	1	1	1	1	1		1						
		T(1)U	Smeara uniaxial gage in S1-dir. upper	0	0	2	2	2	2		2		2				
		T(2)U	Smeara uniaxial gage in S2-dir. upper	0	0	3	0	3	3		3		3				
		ALPHA	Angle from x-axis to mat'l axis 1	0	0	0	0	4	0		0		4				
		BETA	Angle from x-axis to stiff. dir. S1	0	0	0	0	5	0		4		5				
		T(0)L		1	2	1	3	1	4		5		6				
		T(1)L		0	0	2	4	2	5		6		7				
		T(2)L		0	0	3	0	3	6		7		8				
		ALPHAL		0	0	0	0	4	0		0		9				
		BETAL		0	0	0	0	5	0		8		10				
PLATE	5	T(0)	Plate thickness	1	1	1	1	1									
		TS(1)	Smeara uniaxial gage in S1-dir.	0	0	2	2	2									
		TS(2)	Smeara uniaxial gage in S2-dir.	0	0	3	2	3									
		ALPHA	Angle from x-axis to mat'l axis 1	0	2	0	3	4									
		BETA	Angle from x-axis to stiff. dir. S1	0	0	0	4	5									
GPLATE	6	T-MEMB1	Membrane thickness at Node A1	1	1	1	1	1	1	1	1	1	1	1	1		
		T-MEMB2	Membrane thickness at Node A2	1	1	1	2	2	2	2	2	2	2	2	2		
		T-MEMB3	Membrane thickness at Node A3	1	1	1	3	3	3	3	3	3	3	3	3		
		T-MEMB4	Membrane thickness at Node A4 (quad. only)	1	1	1	4	4	0	0	4	4	4	4	4		
		T-MEMB5	Membrane thickness at Node A5 (quad. only)	1	1	1	0	0	0	0	0	0	0	5	5		
		T-BEND1	Bending thickness at Node A1	1	2	2	0	0	4	4	5	5	6	6	6		
		T-BEND2	Bending thickness at Node A2	1	2	2	0	0	5	5	6	6	7	7	7		
		T-BEND3	Bending thickness at Node A3	1	2	2	0	0	6	6	7	7	8	8	8		
		T-BEND4	Bending thickness at Node A4 (quad. only)	1	2	2	0	0	0	0	8	8	9	9	9		
		T-BEND5	Bending thickness at Node A5 (quad. only)	1	2	2	0	0	0	0	0	0	0	10	10		
		ALPHA	Angle from x-axis to mat'l axis 1	0	0	3	0	5	0		7		9		0	11	

Table 152-1. Continued on next page.

Table 152-1. Input Variations of Stiffness - Element Properties (Cont'd.)

ELEMENT TYPE		ELEMENT PROPERTIES (ref. appendix B)				INPUT PROPERTY EXPANSION KEYS ①												
NAME	NO	LABEL	DESCRIPTION	NUMBER OF INPUT PROPERTY VALUES														
				1	2	3	4	5	6	7	8	9	10	11	12	13		
SPRNG	8	None																
SCALAR	9	K(11)	Translational stiffness in x direction	1	1	1			1									
		K(12)	Translational stiffness in y direction	0	0	2			2									
		K(13)	Translational stiffness in z direction	0	0	3			3									
		K(21)	Rotational stiffness about x-axis	0	2	0			4									
		K(22)	Rotational stiffness about y-axis	0	0	0			5									
		K(23)	Rotational stiffness about z-axis	0	0	0			6									
SPRD	10	A(1)	Cross section area at end(1)	1	1													
		A(2)	Cross section area at end(2)	1	2													
SP_ATE	11	T	Plate thickness	1														
CPLATE	12	AREF Axxx.x Txxxx Lnum Ccode	Laminate ref. angle from x-axis Angle from "AREF" to fiber direction Δtemp. from "Icode" element ref. temp. No. of layers of composite mat'l Composite material reference code	"AREF" must be input. A maximum of 10 laminas may be defined. The laminas are input by repeating the sequence of properties "Axxx.x" through "Ccode" as described for Record 7.														
CCOVER	13	AREF-U Axxx.x Txxxx Lnum Ccode	Data for upper plate analogous to data for a CPLATE	Each plate may be defined by a maximum of 10 laminas. The upper plate is input first. The notes for CPLATE are applicable for CCOVER.														
		AREF-L Axxx.x Txxxx Lnum Ccode	Data for lower plate analogous to data for upper plate															

① An expansion key is the column of integers associated with a particular number of input property values for a particular type of element. A nonzero integer component of a key denotes which one of the ordered property values defined by an input data record (either Record 4 or 7) is to be assigned to the corresponding element property shown in the table. A zero integer denotes the property value is to be set to zero. An asterisk (*) denotes a property value is to be automatically assigned the corresponding value (specified or default) of the last defined element of the same type.

Table 152-2. Summary of Stiffness Data Records

Reference Page	Data Records
152.2	BEGIN STIFFNESS DATA
152.2	SET n
152.2	BEGIN PROPERTY DATA
152.2	Pcode <* ≠ text ≠> Plist
152.3	END PROPERTY DATA
152.3	BEGIN ELEMENT DATA
152.4	Eitype <Mcode><Tcode><Userid1> Nlist1 Plist <T0<Userid2> Nlist2 <BY<Incl> inc2>>
152.8	END ELEMENT DATA
152.8	END STIFFNESS DATA

154.0 STRESS (SUPERPOSITION) DATA

The data required for superposition of displacements and stresses for a structural model are defined via this data set. Previous input of the nodal, stiffness, BC and loads data sets (secs. 146.0, 152.0, 106.0 and 134.0, respectively) that are associated with the superimposed data is required.

Each group of superimposed displacements/stresses is identified by a unique superposition load case label and a superposition STAGE (SUPSTAGE) number. A maximum of 10 different stages (superposition and BC STAGES) may be defined per stiffness data set. Superposition load cases and SUPSTAGES may be used in execution of the DESIGN and STRESS Processors (secs. 212.0 and 254.0).

A SUPSTAGE (superposition load case) is defined as a linear combination of previously-defined load cases associated with BC STAGES and/or other SUPSTAGES. The load case factors assigned to the component load cases may be known or unknown. Unknown factors are calculated by the STRESS Processor (sec. 254.1) based on kinematic and/or stress constraints defined by the stress data (sec. 154.1.2).

Printout of the superimposed stresses and/or displacements as calculated by the STRESS Processor may be requested from the STRESS Postprocessor (sec. 254.2).

PRECEDING PAGE BLANK NOT FILMED

154.1 INPUT DATA

Record 1 Data Set Identification

BEGIN STRESS DATA

This record initiates execution of the STRESS Preprocessor.

Record 2 Set Identification (optional)

SET Se

Se = Integer identifying a stiffness SET for which superposition of stresses/displacements is to be effected.

Default Record: The SET number specified by the last record of this type or SET 1.

Record 3 Superposition Stage Identification

Each superposition STAGE (SUPSTAGE) is identified by an integer number. Each SUPSTAGE and BC STAGE number associated with a SET must be unique.

SUPSTAGE st

st = Unique stage number (integer) in the range 1 through 10, inclusive, to be associated with set "Se."

154.1.1 Superposition Load Case Data

A superposition load case may be defined either by Records 4 and 5 or by Record 6. Each of these two input formats contains the superposition load case label and the known or unknown factors for superimposing previously-defined load cases. Each load case associated with a particular SET and STAGE must be identified by a unique label. Both types of input-record formats may be used to define superposition load cases for the SUPSTAGE identified by Record 3.

Record 4 Stage Components

This record defines the SUPSTAGES and/or BC STAGES associated with each superposition load case and components thereof as specified by Record 5.

**STAGES { St
St1 St2 ... Stn }**

St,St1,St2... = Integers identifying superposition and/or BC STAGES that are associated with set "Se" and the load cases "Labeli" specified by Record 5.

If "St" is input by this record, it is assumed that each load case identified by "Labeli" in Record 5 is associated with the stage "St." If the "n" load cases identified by Record 5 are associated with different stages, a corresponding list of "n" STAGES must be identified by this record.

Record 5 Load Case Components

label f1 Label1 f2 Label2 ... fn Labeln

label = Superposition load case label--a positive integer or an alphanumeric word of 1 to 7 characters which may not be the word ALL. This load case label may be the same as a "Labeli" in this record if the associated STAGE is different from the one defined by Record 3.

fi,Labeli = The displacements and/or stresses associated with "Labeli" are to be multiplied by "fi" according to the selected mode of execution of the STRESS Processor (see sec. 254.1). A known load case factor, "fi," is input as a decimal number, whereas a factor that is to be calculated is denoted by the letter X. An unknown factor may only be specified for "Labeli" if the associated STAGE is different from the one defined by Record 3. If the stage numbers associated with the "Labeli" components are different, the number and sequence of these pairs of items (fi, Labeli) must correspond with the number and sequence of stages specified by Record 4.

This record may be repeated to define additional superposition load cases based on the STAGE components specified by Record 4. The "label" defined by one record of this type may not be the same as the "label" specified by another Record

5 or Record 6. A "label" defined by this record, however, may appear as a "Labeli" in subsequent Records 5 and 6.

Records 4 and 5 may be repeated.

Record 6 Superposition Load Cases

This record offers an alternative input format for definition of superposition load cases for the SUPSTAGE identified by Record 3.

label	St1	f1	Label1	f2	Label2 ... fm	Labelm
	St2	fj	Labelj	fk	Labelk ... fn	Labeln
	St3	...				

label = Same as defined for Record 5

Sti = Integer identifying a SUPSTAGE or a BC STAGE that is associated with set "Se" (Record 2) and the subsequent load cases "Labeli" specified by this record.

fi,Labeli = The displacements and/or stresses associated with "Labeli" are to be multiplied by "fi" according to the selected mode of execution of the STRESS Processor (see sec. 254.1). A known load case factor, "fi," is input as a decimal number, whereas a factor that is to be calculated is denoted by the letter X. An unknown factor may only be specified for "Labeli" if the associated STAGE is different from the one defined by Record 3.

This record or Records 4 and 5 may be repeated to define additional superposition load cases. The "label" defined by a Record 6 may not be the same as the "label" specified by another Record 5 or Record 6. A "label" defined by a Record 6, however, may appear as a "Labeli" in subsequent Records 5 and 6.

154.1.2 Superposition Displacement and Stress Constraints (optional)

Constraints to be used in calculation of the unknown load case factors for superposition load cases (displacements and/or stresses) are specified by Records 7-10. The number of constraints defined by these records for a particular superposition load case must be the same as the number of unknown factors denoted by the letter X in Records 5 and/or 6. If all

load case factors are known a priori, Records 7-10 should not be input.

Record 7 Displacement Constraint Data

DISPLACEMENT CONSTRAINTS

Record 8 Nodal Displacements for Superposition Load Cases

Label	Fni	d1	Fni	d2	...
-------	-----	----	-----	----	-----

Label = Superposition load case label previously defined by Record 5 or Record 6.

Fni = Alphanumeric word comprised of a kinematic label, as specified by the BC data, followed by a node number (e.g. TX12, TZ54).

di = Value of the displacement which the freedom at the node specified by "Fni" is to be constrained. The displacement is effected relative to the analysis frame of the specified node.

The kinematic labels specified by the BC data (sec. 106.1) for all STAGE components of a SUPSTAGE must be the same.

This record may be repeated if the "Label" identified by each record is different. Only one displacement, however, may be specified for a particular nodal freedom for a particular superposition load case.

Record 9 Stress Constraint Data

STRESS CONSTRAINTS

Record 10 Element Stresses for Superposition Load Cases

Label	Elid1	Stressid1	s1	Elid2	Stressid2	s2	...
-------	-------	-----------	----	-------	-----------	----	-----

Label = Superposition load case label previously defined by Record 5 or Record 6.

Elidi = Stiffness element identifier. Either the user element number (integer) or the letter I followed by the 1 to 5 digit internal element number assigned via the selected node reorder specification (see sec. 146.0).

- Stressidi = Element stress identifier. Either the alphanumeric stress label or the integer denoting the n-th stress label from those tabulated in Appendix B for the type of element identified by "Elidi."
- si = Value at which "Stressidi" for element "Elidi" is to be constrained.

This record may be repeated if the "Label" identified by each record is different. Only one stress value, however, may be specified for a particular stress identifier for a particular element.

154.1.3 Superposition Load Case Titles

Each load case is identified by a user-assigned label. Additionally, an alphanumeric title may be assigned to load cases via Record 11. Both identifiers are printed when printout is requested.

Record 11 Load Case Title (optional)

LOAD CASE ID Label *≠ text ≠

- Label = Superposition load case label previously defined by Record 5 or Record 6.
- text = Alphanumeric text of 1 to 100 characters, including blanks, to be associated with "Label."
- Default Record: Load cases are identified only by the load case labels specified by Record 5 or Record 6.

This record may be repeated.

Additional Records 3-11 may be input to define additional superposition stages and load cases for set "Se." Additional Records 2-11 may also be input to define superposition stages and load cases for other stiffness SETS.

Record 12 End Data Set

END STRESS DATA

Additional superposition data may be defined by repeating Records 1-12.

Table 154-1. Summary of Stress Data Records

Reference Page	Data Records
154.2	BEGIN STRESS DATA
154.2	SET Se
154.2	<u>SUPSTAGE</u> st
154.2	STAGES { St St1 St2 ... Stn }
154.3	label f1 Label1 f2 Label2 ... fn Labeln
154.4	label St1 f1 Label1 f2 Label2 ... St2 fj Labelj ...
154.5	<u>DISPLACEMENT CONSTRAINTS</u>
154.5	Label Fn1 d1 Fn2 d2 ...
154.5	<u>STRESS CONSTRAINTS</u>
154.5	Label Elid1 Stressid1 s1 Elid2 Stressid2 s2 ...
154.6	LOAD CASE ID Label * ≠ text ≠
154.6	<u>END STRESS DATA</u>

156.0 SUBSET-DEFINITION DATA

The SUBSET-DEFINITION Preprocessor extracts node and element subsets of previously-defined nodal, stiffness and mass data sets for use by other modules. This preprocessor is also used to define subsets of labels (key-word identifiers) used to identify the input/analysis data components associated with an ATLAS job. All ATLAS data components are identified by the labels described in the directory shown by table 156-1. Numerical values of the data components corresponding to the labels included in a subset are extracted from the ATLAS data base via the EXTRACT Postprocessor (sec. 218.0). The extracted data are subsequently used to generate printed or graphical displays.

Each input/output quantity is, in general, referenced to a particular node or element used to define the mathematical model. For example, geometry, node loads, displacements, lumped masses, etc., are associated with nodes, whereas stresses, distributed loads, etc., are associated with elements. The SUBSET-DEFINITION Preprocessor allows the user to identify a portion of the nodes, stiffness elements and/or mass elements which are used to define a model. Furthermore, particular data component labels identifying which data quantities are to be extracted for a node/element subset may be defined as label subsets. Standard, ATLAS label subsets may be referenced directly by the user via pre-defined names in the EXECUTE EXTRACT statement (see table 156-1). Use of these standard label subsets obviates the need, in most cases, to define label subsets via the SUBSET-DEFINITION Preprocessor.

Data associated with the subsets are extracted/manipulated by other ATLAS modules. This capability allows, for example, postprocessing activities such as printing and plotting displays of the node geometry and element-definition data associated with user-selected regions (see secs. 228.4.1, 246.1.2 and 252.1.2). Visual examination of segments of the input data sets can thus be effected. Additionally, interpretation and evaluation of processor output data associated with appropriately defined subsets can be performed. For example, printed or plotted subsets of displacements, stresses and vibration mode shapes can be examined by using this capability (see secs. 228.4, 254.2.1 and 258.2). All ATLAS graphical displays are effected via subsets defined by input to the SUBSET-DEFINITION Preprocessor.

PROCESSED PAGE BLANK NOT FILMED

156.1 INPUT DATA

Subset-definition records within this data set allow the following types of subset manipulations to be performed:

- a) Node Subset Definitions
- b) Ordered Node Subset Definitions
- c) Element Subset Definitions
- d) Label Subset Definitions
- e) Extraction of Subsets from Other Subsets
- f) Definition of Subsets via Subset Combinations
- g) Subset Modifications (Exclusions)
- h) Node and Element Subsets Defined via Isolation of Geometric Regions

If nodal, stiffness-element or mass-element subsets are to be identified, the corresponding nodal, stiffness and mass data sets, as required, must be input prior to this data set in the input stream.

Record 1 Input Data Set Identification

BEGIN SUBSET DEFINITION

This record initiates execution of the SUBSET-DEFINITION Preprocessor.

Record 2 Identification of Parent Data Set

This record identifies the type of subsets defined by subsequent Records 3.

SUBSETS OF {
STIFFNESS SET Se
MASS SET Se
NODAL SET Se
LABELS
}

Se = Number (integer) of a STIFFNESS, MASS or NODAL data set from which subsets are to be extracted. The types of subsets which may be defined by subsequent Records 3 depend on the selected variation of this record. The options are identified by one of the following key-words:

STIFFNESS-- Stiffness element and/or node subsets associated with the nodal/stiffness set "Se."

MASS-- Mass element and/or node subsets associated with the nodal/mass set "Se."

NODAL-- Node subsets of nodal set "Se."

LABELS-- Label subsets which are not dependent on a SET identifier.

Record 3 Subset-Definition Records

Each subset-definition record defines the name and contents of one subset. The general format of this type of record is as follows:

Sn = {Subset-definition expression}

At least one space is required before and after the equal sign within a subset-definition record. The subset name "Sn" must be one of the following alphanumeric words:

Nxxx-- Name of a node subset--the leading character is the letter N.

ONxxx-- Name of an ordered node subset--the leading characters are the letters ON.

Exxx-- Name of an element subset--the leading character is the letter E.

The characters xxx in the node and element subset names denote the subset number, a 1 to 3 digit integer. Zeros prior to the first non-zero digit are ignored. Thus, N4, N04 and N004 are equivalent names. Example subset names are N1, N02, N950, ON40, ON097, E1, E005, E76.

name-- User-specified name of a "label" subset; an alphanumeric word of 1 to 7 characters. User-defined "label" subsets may not be identified by any of the ATLAS standard label-subset names which are:

BMODE	KPROP	STRESS
DISGRID	LOADAB	STRPRINT
DISNODE	MGRID	TMS
DISPRINT	NODES	VGVF
KGRID	SMS	VMODE

Subset names allow user-referral to specific subsets within subsequent subset-definition records and within other module statements for data extraction purposes.

Descriptions of the various subset-definition expressions are presented in the following sections. Any combination of the variations may be used. The following guidelines are provided:

- a) If more than one subset is identified by the same name, only the last subset defined by that name can be interrcgated subsequently.
- b) A warning is issued if a null (empty) subset is defined. Subsequent use of a null subset by another module is, in general, not allowed.
- c) An ordered node subset, ONxxx, is used to establish a sequence of nodes which, when connected in the specified order, defines a contiguous series of line segments. This type of subset is used, for example, to define a boundary for contour plots (see EXTRACT and GRAPHICS Postprocessors--secs. 218.0 and 228.0).
- d) "Label" subsets are used only by the EXTRACT Postprocessor (sec. 218.0). Subsets of data component LABELS identify which input/analysis data values are to be extracted for subsequent graphical or print display. Use of the ATLAS standard label subsets identified by table 156-1 as input to the EXTRACT Postprocessor obviates the need, in most cases, to define label subsets.

This record may be repeated to define a maximum of 999 node subsets, 999 ordered node subsets, 999 stiffness element subsets and 999 mass element subsets for the set "Se" identified by Record 2. Additionally, a maximum of 999 set-independent label subsets may be defined per job.

Records 2 and 3 may be repeated to define additional LABEL subsets or subsets of different node, mass and stiffness data sets.

Record 4 End Data Set

END SUBSET DEFINITION

This record denotes the end of a subset-definition data set. Additional subset-definition data sets may be input by repeating Records 1-4.

156.2 SUBSET-DEFINITION RECORDS

Descriptions of the various forms of subset-definition records (Recrd 3) are presented below. The following notation is used:

- a) If a recrd refers specifically to a particular type of subset (nodal, ordered nodal, element or label), the subset name is denoted accordingly by Nxxx, ONxxx, Exxx or "name."
- b) If a record may be used to define multiple types of subsets, the subset names are denoted therein by "Sn" or "Sni" where "i" is an integer.
- c) The item "Elements" in element-related subset-definition records denotes a list of one or more element types such as RODS, BEAMS, PLATES, SCALARS, etc. The list may include element types that are not available in set "Se" (Record 2). See Appendices B and C for available stiffness and mass element type names, respectively.
- d) The key-word CLOSED or the key-word OPEN may be input for the item "Opt" used in several element-related subset-definition records. This key-word controls which elements are included in a particular subset. CLOSED means that only the elements that attach exclusively to the nodes within a specified region are included. OPEN means that all elements which attach to any of the nodes contained within the specified node subset or region are included. The default for "Opt" is CLOSED.

156.2.1 Node Subset Definition Records

Six input-record variations may be used to define node subsets.

Variation 1

Nxxx = ALL

Nxxx will contain all of the nodes in the data set.

Variation 2

Nxxx = ATlist

ATlist = ATLAS list of node numbers (integers) which are to be included in Nxxx.

Variations 3-6 may be used to define node subsets by employing the region definitions presented in section 156.3. Subsets defined by these variations will contain only the nodes that lie within or on the surface of the region. A node is considered to be within a region if the perpendicular distance from the node to the closest boundary of the region is less than 10^{-8} units.

Variation 3

Nxxx = CYLINDER Nn1 TO Nn2 <RADIUS r>

Variation 4

Nxxx = SLAB Nn1 Nn2 Nn3 <TOLERANCE t1 <t2>>

Variation 5

Nxxx = SLAB $\begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}$ cord1 <TC ccrd2>

Variation 6

Nxxx = TUBE Nn1 Nn2 ... Nnk DIRECTION
 $\begin{Bmatrix} dv1 & dv2 & dv3 \\ Np1 & Np2 \\ Np1 \end{Bmatrix}$ <TOLERANCE t1>

156.2.2 Ordered Node Subset Definition Records

An ordered subset of nodes is defined by the following type of record:

ONxxx = ATlist

ATlist = ATLAS list of node numbers (integers) which are to be included in ONxxx. The order in which the node numbers are input defines a particular sequence (string) of nodes.

156.2.3 Element Subset Definition Records

Six input-record variations may be used to define element subsets.

Variation 1

Exxx = $\begin{Bmatrix} \text{ALL} \\ \text{Elements} \end{Bmatrix}$

Exxx will contain either ALL elements in the data set or only the element types specified by "Elements."

Variation 2

Exxx = ATlist

ATlist = ATLAS list of element numbers (integers) which are to be included in Exxx.

Variations 3-6 may be used to define element subsets by employing the region definitions presented in section 156.3. Subsets defined by these variations will contain only the element types denoted by "Elements" or the word ALL (the default is ALL) that attach to the nodes within or on the surface of the region. A node is considered to be within a region if the perpendicular distance from the node to the closest boundary of the region is less than 10^{-8} units. Auxiliary nodes and "Shear" nodes are ignored.

Variation 3

Exxx = <Opt> $\left\langle \begin{array}{c} \text{ALL} \\ \text{Elements} \end{array} \right\rangle$ CYLINDER Nn1 TO Nn2
<RADIUS r>

Variation 4

Exxx = <Opt> $\left\langle \begin{array}{c} \text{ALL} \\ \text{Elements} \end{array} \right\rangle$ SLAB Nn1 Nn2 Nn3
<TOLERANCE t1 <t2>>

Variation 5

Exxx = <Opt> $\left\langle \begin{array}{c} \text{ALL} \\ \text{Elements} \end{array} \right\rangle$ SLAB $\left\{ \begin{array}{c} X \\ Y \\ Z \end{array} \right\}$ cord1
<TO cord2>

Variation 6

Exxx = <Opt> $\left\langle \begin{array}{c} \text{ALL} \\ \text{Elements} \end{array} \right\rangle$ TUBE Nn1 Nn2 ...

Nnk DIRECTION $\left\{ \begin{array}{ccc} dv1 & dv2 & dv3 \\ Np1 & Np2 & \\ Np1 & & \end{array} \right\}$ <TOLERANCE t1>

156.2.4 Label Subset Definition Records

A subset of input/analysis data component labels is defined by the following type of record:

name = ATlist

ATlist = ATLAS list of component labels (alphanumeric words) which are to be included in the subset identified by "name." Key-words specified via ATlist must be selected from the "Component Label Directory" shown by table 156-1.

156.2.5 Records for Extraction of Subsets from Other Subsets

The records described in this section allow the user to define a data subset relative to a previously-defined subset.

Variation 1

$Nxxx = \text{NODES IN } Exxx$

Nxxx will contain only the data set nodes to which the elements in the previously-defined element subset Exxx are attached. Auxiliary nodes and "Shear" nodes are ignored.

Variation 2

$Nxxx = \text{ONxxx}$

Nxxx will contain the same nodes that are included in ONxxx. A particular order of the nodes in Nxxx, however, is not preserved.

Variation 3

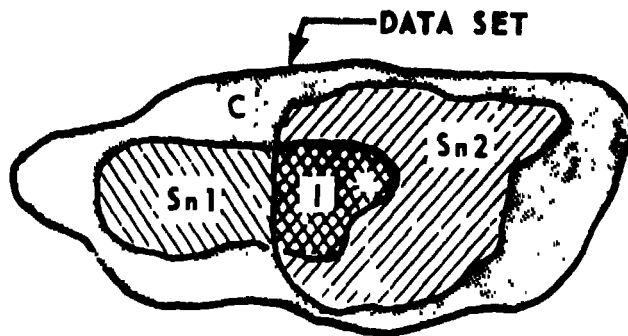
$Exxx = \langle \text{Opt} \rangle \left\langle \begin{array}{c} \text{ALL} \\ \text{Elements} \end{array} \right\rangle \text{IN } Nxxx$

Exxx will contain the element types denoted by "Elements" or the word ALL (the default is ALL) which attach to the nodes in the previously-defined node subset Nxxx.

156.2.6 Records for Definition of Subsets via Subset Combinations

Subsets may be formed by combining previously-defined subsets. Only node subsets can be combined to form a new node subset, only element subsets can be combined to form a new element subset and only label subsets can be combined to form a new label subset. These capabilities are analogous to the set-algebra operations defined as unions, intersections and complements of sets. The letters U, I and C denote these operations, respectively, in the following records. It should be noted that at least one space is required before and after these letters within a subset-definition record.

The cross-hatched region in the sketch below illustrates the subset formed by the intersection (I) of two subsets, Sn1 and Sn2. The union (U) is defined as the sum of the two subregions. The complement (C) to the union set (Sn1 U Sn2) is shown as the shaded region in the sketch.



In the following records, S_n denotes the name of a subset to be generated from previously-defined subsets denoted by S_{n1} , S_{n2} , ..., S_{nk} . Only one type of subset (node, element or label) may be identified in each record of this type.

Variation 1 $S_r = S_{n1} \cup S_{n2}$

S_n will contain all the nodes, elements or component labels that are in subsets S_{n1} and S_{n2}

Variation 2 $S_n = S_{n1} \cap S_{n2}$

S_n will contain only the nodes, elements or component labels common to subsets S_{n1} and S_{n2} . If none are common, the resulting subset is empty (null).

Variation 3 $S_n = C \ S_{n1}$

S_n will contain all the data set nodes or elements which are not in subset S_{n1} .

Variation 4 $S_n = S_{n1}$

S_n will contain the same nodes or elements that are in subset S_{n1} .

Variation 5 $S_n = \{ \text{Set-algebra expression} \}$

This variation allows a maximum of 30 set-algebra operations (U, I and C) to be specified within a single

subset-definition record. The sequence in which the operations will be executed is specified via pairs of parentheses. That is, the operation specified within the inner-most pair of parentheses is executed first. This first operation generates a subset which is then operated on as specified within the next inner-most set of parentheses, etc. All operations are executed in a left-to-right sequence.

Caution: At least one space is required before and after each parenthesis, each U, I or C operator, and the equal sign.

Consider, for example, the following input record:

$$S_n = S_{n1} \cup ((C (S_{n5} \cap S_{n3})) \cup S_{n4})$$

This single record is equivalent to the following four records in the sequence shown.

$$\begin{aligned} S_{na} &= S_{n5} \cap S_{n3} \ / \\ S_{nb} &= C \ S_{na} \ / \\ S_{nc} &= S_{nb} \cup S_{n4} \ / \\ S_n &= S_{n1} \cup S_{nc} \ / \end{aligned}$$

If these four records are input, four subsets (S_{na} , S_{nb} , S_{nc} and S_n) are formed and saved for subsequent use. However, if only subset S_n is to be saved, the single record shown above must be used.

156.2.7 Subset Modification (Exclusion) Records

The records described below allow the user to exclude selected items from previously-defined node and/or element subsets. Each of these records has the following format:

EXCLUDE ATlist1 <FROM ATlist2>

ATlist1 = ATLAS list of one of the following:

- a) node numbers (integers) or node subset names,
- b) element numbers (integers) or element subset names.

ATlist2 = ATlist list of previously-defined node or element subsets from which the nodes or elements contained in ATlist1 are to be excluded.

SUBSET DEFINITION

Default: ATlist2 is the subset defined by the preceding record.

An EXCLUDE command may involve only node numbers and node subsets or only element numbers and element subsets. The subsets that are modified are merely redefined with the indicated items excluded. If a user inadvertently specifies a subset to be excluded from itself, that subset will be left unaltered. To illustrate the use of this command, consider the following node subset modification record.

```
EXCLUDE 5, N2 TO N4 FROM N1 TO N19 BY 9 /
```

This single record is equivalent to the following 12 records:

```
EXCLUDE N2 FROM N1 /  
EXCLUDE N3 FROM N1 /  
EXCLUDE N4 FROM N1 /  
EXCLUDE N2 FROM N10 /  
EXCLUDE N3 FROM N10 /  
EXCLUDE N4 FROM N10 /  
EXCLUDE N2 FROM N19 /  
EXCLUDE N3 FROM N19 /  
EXCLUDE N4 FROM N19 /  
EXCLUDE 5 FROM N1 /  
EXCLUDE 5 FROM N10 /  
EXCLUDE 5 FROM N19 /
```

156.3 SUBSET REGION DEFINITIONS

Several variations of the foregoing node and element subset-definition records may use certain 1-, 2- and 3-dimensional geometric shapes to isolate regions (subsets) of a data set. This library of shapes allows the user to define the following types of regions:

- a) A right circular cylinder of finite length--in its degenerate form, it defines a line segment;
- b) A finite-thickness, infinite slab--in its degenerate form, it defines an infinite plane;
- c) A polygonal-section, infinite-length tube--in its degenerate form, it defines a finite-width infinite-length plane.

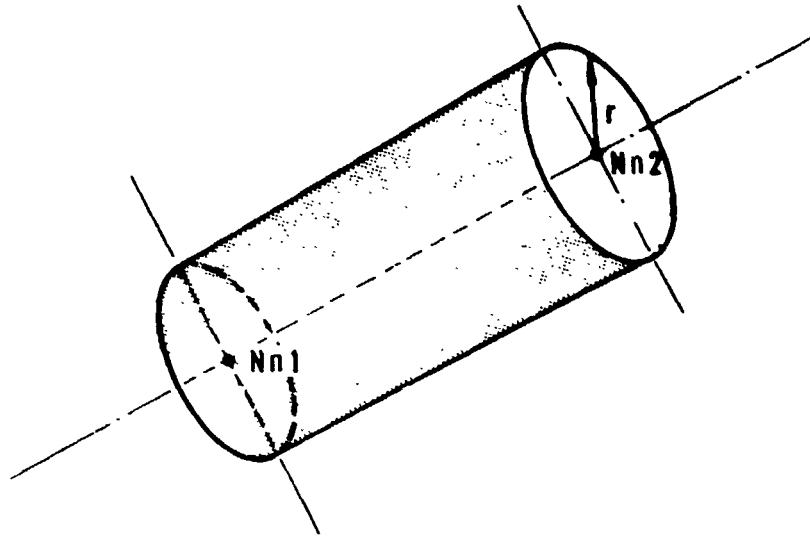
Each of these capabilities is discussed below. Descriptions of the key-words and items required to define these geometric shapes are also presented. Node subset regions will contain only the nodes that lie within or on the surface (boundary) of the region (see sec. 156.2.1). Element subset regions will contain only the elements that attach to the nodes within or on the surface of the region (see sec. 156.2.3). A node is within a region if the perpendicular distance from the node to the closest boundary of the region is less than 10^{-6} units.

156.3.1 Right Circular Cylinder of Finite Length

{CYLINDER Nn1 TO Nn2 <RADIUS r>}

- Nn1 = Number (integer) of a node which defines one point on the longitudinal axis and one base of the cylindrical region
- Nn2 = Number (integer) of a second node on the longitudinal axis which defines the second base of the cylindrical region.
- r = The radius ≥ 0.0 of the cylindrical region.
Default: r=0.0

The region generated is a circular cylinder whose base (end) planes are perpendicular to its longitudinal axis. The region degenerates to a straight line segment between nodes Nn1 and Nn2 when the radius is zero. For this case, only the nodes lying on the line segment are included in the subset.



156.3.2 Finite-Thickness Infinite Slab

Variation 1--General Slab

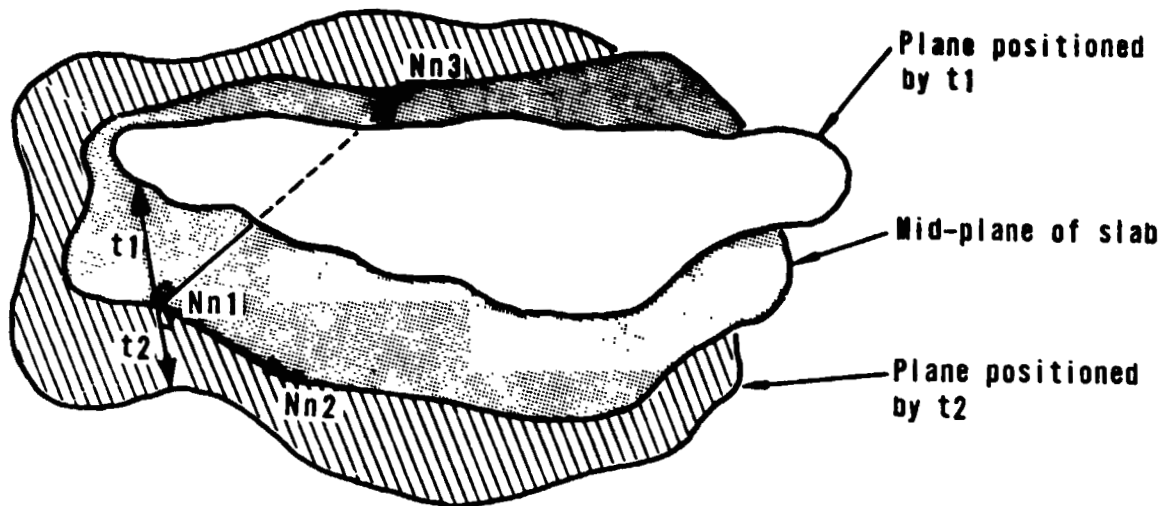
{SLAB Nn1 Nn2 Nn3 <TOLERANCE t1 <t2>>}

Nn1, Nn2, Nn3 = Numbers (integers) of three nodes that define the mid-plane of the slab.

t1 = Perpendicular distance ≥ 0.0 from the mid-plane to the face which is positioned by the vector-product of Nn1-Nn2 and Nn1-Nn3 using the right-hand rule--see the figure below.
Default: t1=t2=0.0

t2 = Perpendicular distance ≥ 0.0 from the mid-plane to the other face--see the figure below.
Default: t2=t1

The basic slab is composed of three infinite planes--a mid-plane, a plane positioned by "t1" and a plane positioned by "t2" (see the figure below). A slab allows the user to define a region with a total thickness of "t1" plus "t2." If "t1" or "t2" is specified as zero, the slab is defined by two parallel planes. Additionally, if "t2" and "t1" are both zero, the slab degenerates to the mid-plane.

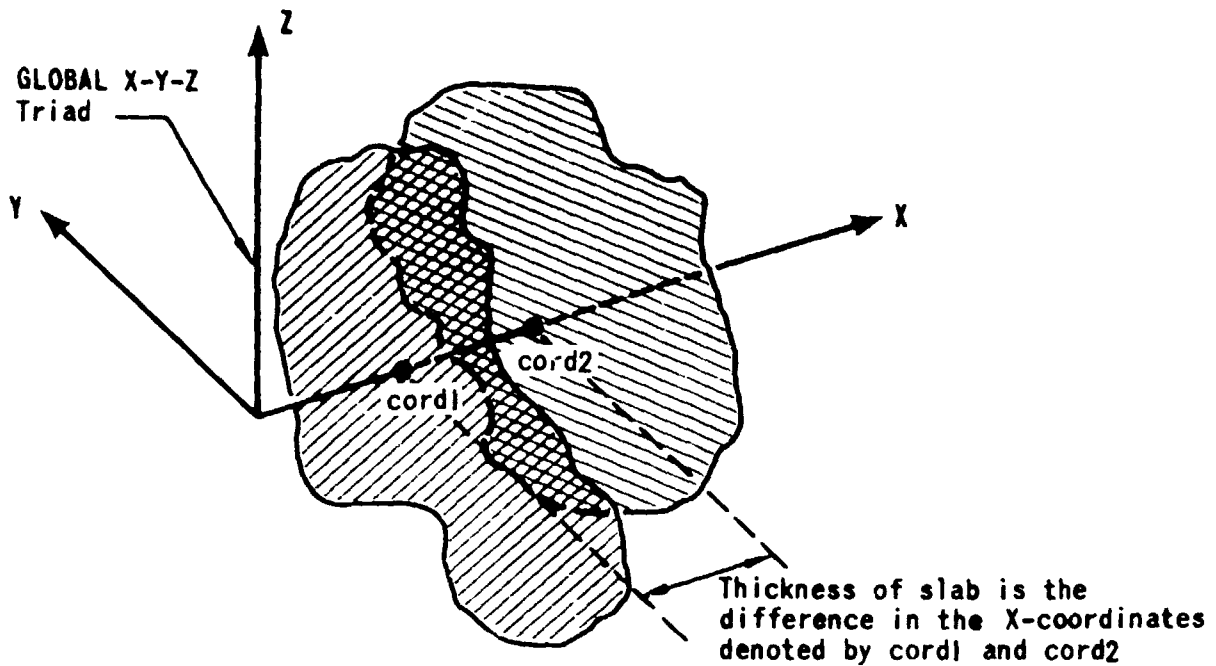


Variation 2--GLOBAL Slab

{SLAB $\begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}$ cord1 <TO ccord2>}

- $\begin{Bmatrix} X \\ Y \\ Z \end{Bmatrix}$ = The letter X, Y or Z denoting the GLOBAL axis which is perpendicular to the slab.
- cord1 = Coordinate of a point on the selected X, Y or Z axis--this point defines one face of the slab.
- cord2 = Coordinate of a second point on the selected X, Y or Z axis which defines the second face of the slab.
Default: cord2=cord1

The infinite slab is perpendicular to the specified GLOBAL axis and has a total thickness of |cord1 - cord2|. If "cord1" equals "cord2" the region degenerates to an infinite plane located by "cord1." The figure shown below illustrates a GLOBAL X slab.



156.3.3 Infinite-Length Polygonal Tube

<pre>{TUBE Nn1 Nn2 ... Nnk DIRECTION <TOLERANCE t1>}</pre>	<pre>{ dv1 dv2 dv3 Np1 Np2 Np1 }</pre>
--	--

Nn1, Nn2, ..., Nnk = A list of numbers (integers) of two or more nodes (Nnk ≠ Nn1). A maximum of 230 nodes may be specified. These nodes define the basic cross section of the tube. The specified nodes need not lie in a plane.

$\begin{Bmatrix} dv1 & dv2 & dv3 \\ Np1 & Np2 \\ Np1 \end{Bmatrix}$ = dv1, dv2 and dv3 are the X, Y, Z coordinates, respectively, (integer or decimal) of a point (not necessarily a node in the data set) in the GLOBAL frame. The direction vector of the tube is directed from the origin of the GLOBAL frame to the point defined by dv1, dv2 and dv3--see the figures below.

The direction vector may be defined via node numbers (integers) "Np1" and "Np2." In this case, the direction vector goes from "Np1" to "Np2." If only "Np1" is specified, the direction vector goes from the GLCBAL origin through "Np1."

The direction vector need not be located within the tube region.

t1 = A tolerance which allows the basic cross section of a tube to be expanded or contracted uniformly. A positive value denotes a uniform expansion of the tube, whereas a negative value denotes a uniform contraction of the tube by an amount "t1" (see the figures below).
Default: t1=0.0

The cross section of a tube is generated by projecting the nodes Nn1, Nn2, ..., Nnk onto a plane that is perpendicular to the direction vector (see the figures below). The resulting points, pn1, pn2, ..., pnk define the basic polygonal cross section. These points, "pi," are connected in the same sequence as the corresponding nodes are specified. This basic polygonal section can be modified, expanded or contracted, by a specified tolerance. The expansion case is illustrated in the figures below. If the tolerance is zero, the basic polygon defines the cross section of the tube region. When only two nodes are specified, the region degenerates to a finite-width infinite-length plane. Caution must be exercised by the user in specifying the node sequence that defines the tube(s). Multiple tubes may be defined by one record.

**SUBSET
DEFINITION**

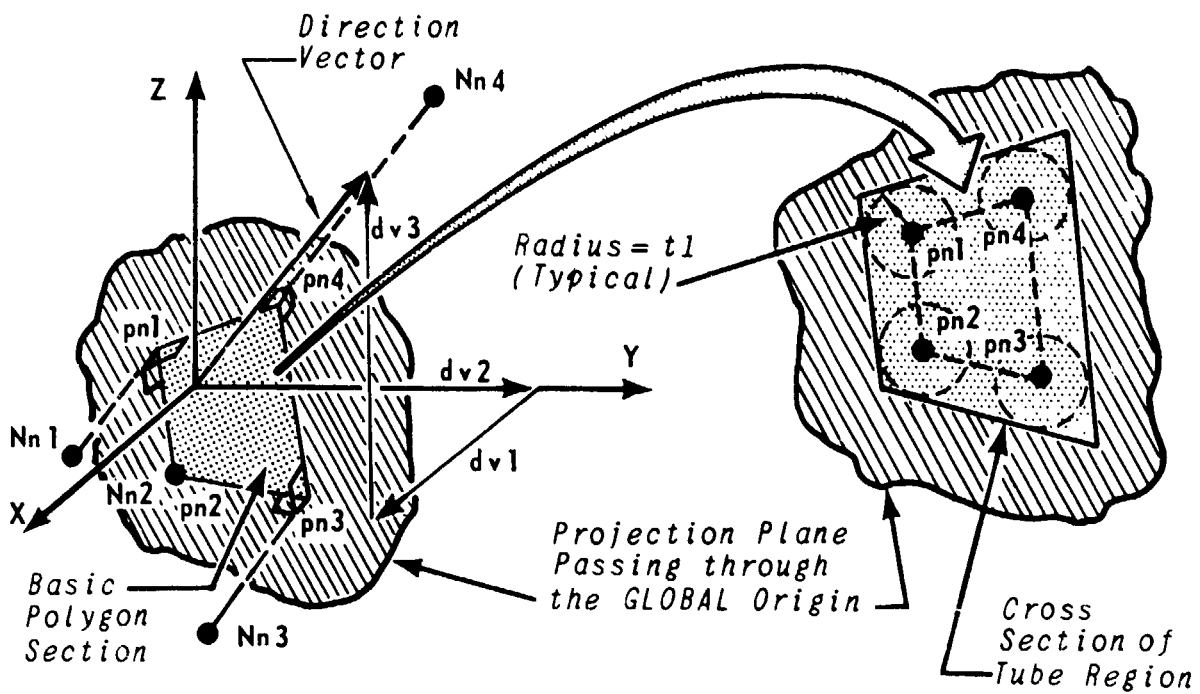


Table 156-1. Directory of ATLAS Data-Component Labels and Standard Label-Subset (LSUB) Names

NO.	LABEL	DATA COMPONENT	LSUB NAMES														
			BDODE	DISGRID	DISNODE	DISPRINT	KGRID	KPROP	LOADAB	MGRID	NODES	SMS	STRESS	STRESS1	TMS	YGVF	WMODE
(1)	BWA1	BEAM auxiliary node defining elastic axis location at end (1)															
(2)	BW22	BEAM auxiliary node defining elastic axis location at end (2)															
(3)	BWA3	BEAM auxiliary node defining the x-y principal plane															
(4)	BWA-VV(1)	BEAM effective shear area in y-direction at end (1)															
(5)	BWZ-VV(2)	BEAM effective shear area in z-direction at end (2)															
(6)	BWA-VV(1)	BEAM effective shear area in y-direction at end (1)															
(7)	BWA-VV(2)	BEAM effective shear area in z-direction at end (2)															
(8)	BWA(1)	BEAM cross-section area at end (1)															
(9)	BWA(2)	BEAM cross-section area at end (2)															
(10)	BPCONSTR	BEAM end-force restraint indicator															
(11)	BWID	BEAM user-id number															
(12)	BWY(1)	BEAM moment of inertia about y-axis at end (1)															
(13)	BWY(2)	BEAM moment of inertia about y-axis at end (2)															
(14)	BWZ(1)	BEAM moment of inertia about z-axis at end (1)															
(15)	BWZ(2)	BEAM moment of inertia about z-axis at end (2)															
(16)	BWJ(1)	BEAM St. Venant torsion constant at end (1)															
(17)	BWJ(2)	BEAM St. Venant torsion constant at end (2)															
(18)	BWMT	BEAM material code number															
(19)	BWY(1)	BEAM bending moment about y-axis at end (1)															
(20)	BWY(2)	BEAM bending moment about y-axis at end (2)															
(21)	BWZ(1)	BEAM bending moment about z-axis at end (1)															
(22)	BWZ(2)	BEAM bending moment about z-axis at end (2)															
(23)	BWNI	BEAM structural node rigidly-connected to end (1)															
(24)	BWZ	BEAM structural node rigidly-connected to end (2)															
(25)	BWZ(2)	BEAM axial force at end (2)															
(26)	BWSMS	BEAM minimum stress, margin of safety															
(27)	BWTEMP	BEAM temperature															
(28)	BWMS	BEAM minimum, thermal, margin of safety															
(29)	BWT(2)	BEAM torque at end (2)															
(30)	BWY(2)	BEAM shear force in y direction at end (2)															
(31)	BWZ(2)	BEAM shear force in z direction at end (2)															
(32)	BWA1	BRICK auxiliary node 1															
(33)	BWA2	BRICK auxiliary node 2															
(34)	BWA3	BRICK auxiliary node 3															
(35)	BWID	BRICK user-id number															
(36)	BWMT	BRICK material code number															
(37)	BWNI	BRICK corner node 1															
(38)	BWNI0	BRICK edge node 10															
(39)	BWNI1	BRICK edge node 11															
(40)	BWNI2	BRICK edge node 12															
(41)	BWNI3	BRICK edge node 13															
(42)	BWNI4	BRICK edge node 14															
(43)	BWNI5	BRICK edge node 15															
(44)	BWNI6	BRICK edge node 16															
(45)	BWNI7	BRICK edge node 17															
(46)	BWNI8	BRICK edge node 18															
(47)	BWNI9	BRICK edge node 19															
(48)	BWNI0	BRICK corner node 2															
(49)	BWNI1	BRICK edge node 20															
(50)	BWNI2	BRICK edge node 21															
(51)	BWNI3	BRICK edge node 22															
(52)	BWNI4	BRICK edge node 23															

Table continued on next page

SUBSET
DEFINITION

Table 156-1. Directory of ATLAS Data-Component Labels and Standard Label-Subset (LSUB) Names (Cont'd.)

NO.	LABEL	DATA COMPONENT	LSUB NAMES														
			BNODE	DISCRD	DISNODE	DISPRINT	KRGRID	KRPOP	LOADAB	MGRID	NODES	SMS	STRESS	STREPRINT	TAS	WGYF	WMODE
(53)	BRN24	BRICK edge node 24															
(54)	BRN25	BRICK edge node 25															
(55)	BRN26	BRICK edge node 26															
(56)	BRN27	BRICK edge node 27															
(57)	BRN28	BRICK edge node 28															
(58)	BRN29	BRICK edge node 29															
(59)	BRN30	BRICK corner node 3															
(60)	BRN31	BRICK edge node 30															
(61)	BRN32	BRICK edge node 31															
(62)	BRN33	BRICK edge node 32															
(63)	BRN34	BRICK edge node 33															
(64)	BRN35	BRICK edge node 34															
(65)	BRN36	BRICK edge node 35															
(66)	BRN37	BRICK edge node 36															
(67)	BRN38	BRICK edge node 37															
(68)	BRN39	BRICK edge node 38															
(69)	BRN40	BRICK edge node 39															
(70)	BRN41	BRICK corner node 4															
(71)	BRN42	BRICK edge node 40															
(72)	BRN43	BRICK edge node 41															
(73)	BRN44	BRICK edge node 42															
(74)	BRN45	BRICK edge node 43															
(75)	BRN46	BRICK edge node 44															
(76)	BRN47	BRICK corner node 5															
(77)	BRN48	BRICK corner node 6															
(78)	BRN49	BRICK corner node 7															
(79)	BRN50	BRICK corner node 8															
(80)	BRN51	BRICK corner node 9															
(81)	BRN52	BRICK axial stress; x-component															
(82)	BRN53	BRICK axial stress; y-component															
(83)	BRN54	BRICK axial stress; z-component															
(84)	BRN55	BRICK minimum stress, margin of safety															
(85)	BRN56	BRICK shear stress; x-y component															
(86)	BRN57	BRICK shear stress; x-z component															
(87)	BRN58	BRICK shear stress; y-z component															
(88)	BRN59	BRICK temperature															
(89)	BRN60	BRICK minimum, thermal, margin of safety															
(90)	BRN61	CCOVER lower-plate reference angle															
(91)	BRN62	CCOVER upper-plate reference angle															
(92)	BRN63	CCOVER lower-plate strain in direction 1															
(93)	BRN64	CCOVER upper-plate strain in direction 1															
(94)	BRN65	CCOVER lower-plate strain in direction 2															
(95)	BRN66	CCOVER upper-plate strain in direction 2															
(96)	BRN67	CCOVER lower-plate shear strain															
(97)	BRN68	CCOVER upper-plate shear strain															
(98)	BRN69	CCOVER user-id number															
(99)	BRN70	CCOVER lower-plate lamina 1 definition															
(100)	BRN71	CCOVER upper-plate lamina 1 definition															
(101)	BRN72	CCOVER lower-plate lamina 2 definition															

Table continued on next page

Table 156-1. Directory of ATLAS Data-Component Labels and Standard Label-Subset (LSUB) Names (Cont'd.)

NO.	LABEL	DATA COMPONENT	LSUB NAMES														
			BLADE	DISGRID	DISNODE	DISPRINT	KGRID	KPROP	LOADAB	WGRID	NODES	SMS	STRESS	STRPRINT	TMS	WVFF	WVADE
(102)	CCVLAN02-U	COVER upper-plate lamina 2 definition															
(103)	CCVLAN03-L	COVER lower-plate lamina 3 definition															
(104)	CCVLAN03-U	COVER upper-plate lamina 3 definition															
(105)	CCVLAN04-L	COVER lower-plate lamina 4 definition															
(106)	CCVLAN04-U	COVER upper-plate lamina 4 definition															
(107)	CCVLAN05-L	COVER lower-plate lamina 5 definition															
(108)	CCVLAN05-U	COVER upper-plate lamina 5 definition															
(109)	CCVLAN06-L	COVER lower-plate lamina 6 definition															
(110)	CCVLAN06-U	COVER upper-plate lamina 6 definition															
(111)	CCVLAN07-L	COVER lower-plate lamina 7 definition															
(112)	CCVLAN07-U	COVER upper-plate lamina 7 definition															
(113)	CCVLAN08-L	COVER lower-plate lamina 8 definition															
(114)	CCVLAN08-U	COVER upper-plate lamina 8 definition															
(115)	CCVLAN09-L	COVER lower-plate lamina 9 definition															
(116)	CCVLAN09-U	COVER upper-plate lamina 9 definition															
(117)	CCVLAN10-L	COVER lower-plate lamina 10 definition															
(118)	CCVLAN10-U	COVER upper-plate lamina 10 definition															
(119)	CCVAL	COVER structural mid-surface node 1	x														
(120)	CCVA2	COVER structural mid-surface node 2	x														
(121)	CCVA3	COVER structural mid-surface node 3	x														
(122)	CCVA4	COVER structural mid-surface node 4															
(123)	CCVSGMAIL	COVER lower-plate axial stress 1															
(124)	CCVSGMAILU	COVER upper-plate axial stress 1															
(125)	CCVSGMA2L	COVER lower-plate axial stress 2															
(126)	CCVSGMA2U	COVER upper-plate axial stress 2															
(127)	CCVSM5	COVER minimum, stress, margin of safety															
(128)	CCVTAU12L	COVER lower-plate shear stress															
(129)	CCVTAU12U	COVER upper-plate shear stress															
(130)	CCVTTEMP	COVER temperature															
(131)	CCVTHICK-L	COVER lower-plate total thickness															
(132)	CCVTHICK-U	COVER upper-plate total thickness															
(133)	CCVTMS	COVER minimum, thermal, margin of safety															
(134)	CCVALPHAL	COVER lower-plate angle between XL and material axis 1															
(135)	CCVALPHAU	COVER upper-plate angle between XU and material axis 1															
(136)	CCVREY1L	COVER lower-plate angle between XL and stiffener direction S1															
(137)	CCVREY1U	COVER upper-plate angle between XU and stiffener direction S1															
(138)	CCVM1T	COVER user-id number	x														
(139)	CCVM1T	COVER material code number															
(140)	CCVA1	COVER structural mid-surface node 1															
(141)	CCVA2	COVER structural mid-surface node 2															
(142)	CCVA3	COVER structural mid-surface node 3															
(143)	CCVA4	COVER structural mid-surface node 4															
(144)	CCVSGMAIL	COVER lower-stiffener axial stress in direction S1															
(145)	CCVSGMAILU	COVER upper-stiffener axial stress in direction S1															
(146)	CCVSGMA2L	COVER lower-stiffener axial stress in direction S2															
(147)	CCVSGMA2U	COVER upper-stiffener axial stress in direction S2															
(148)	CCVSG-S1L	COVER axial stress in lower plate in direction 1															
(149)	CCVSG-S1U	COVER axial stress in upper plate in direction 1															
(150)	CCVSG-S2L	COVER axial stress in lower plate in direction 2															
(151)	CCVSG-S2U	COVER axial stress in upper plate in direction 2															
(152)	CCVSM5	COVER minimum, stress, margin of safety															
(153)	CCVTAU12L	COVER shear stress in lower plate															

Table continued on next page

Table 156-1. Directory of ATLAS Data-Component Labels and Standard Label-Subset (LSUB) Names (Cont'd.)

NO.	LABEL	DATA COMPONENT	LSUB NAMES												
			BDIDE	DISNODE	DISPRINT	KGRID	KPROP	LOADB	KGRID	NODES	SMS	STRESS	STRESSPRINT	WAVE	WAODE
1154	CCV71121	COVER shear stress in upper plate													
1155	CCV71122	COVER temperature													
1156	CCV71123	COVER minimum, thermal, margin of safety													
1157	CCV71124	COVER lower plate thickness													
1158	CCV71125	COVER upper plate thickness													
1159	CCV71126	COVER lower-plate smeared uniaxial thickness in direction 1													
1160	CCV71127	COVER lower-plate smeared uniaxial thickness in direction 2													
1161	CCV7121	COVER upper-plate smeared uniaxial thickness in direction 1													
1162	CCV7122	COVER upper-plate smeared uniaxial thickness in direction 2													
1163	CPL7111	CPLATE reference angle													
1164	CPL7112	CPLATE auxiliary node 1													
1165	CPL7113	CPLATE auxiliary node 2													
1166	CPL7114	CPLATE auxiliary node 3													
1167	CPL7115	CPLATE auxiliary node 4													
1168	CPL7116	CPLATE strain in direction 1													
1169	CPL7117	CPLATE strain in direction 2													
1170	CPL7118	CPLATE shear strain													
1171	CPL7119	CPLATE user-id number													
1172	CPL7120	CPLATE lamina 1 definition													
1173	CPL7121	CPLATE lamina 2 definition													
1174	CPL7122	CPLATE lamina 3 definition													
1175	CPL7123	CPLATE lamina 4 definition													
1176	CPL7124	CPLATE lamina 5 definition													
1177	CPL7125	CPLATE lamina 6 definition													
1178	CPL7126	CPLATE lamina 7 definition													
1179	CPL7127	CPLATE lamina 8 definition													
1180	CPL7128	CPLATE lamina 9 definition													
1181	CPL7129	CPLATE lamina 10 definition													
1182	CPL81	CPLATE structural node 1													
1183	CPL82	CPLATE structural node 2													
1184	CPL83	CPLATE structural node 3													
1185	CPL84	CPLATE structural node 4													
1186	CPL81GM1	CPLATE axial stress in direction 1													
1187	CPL81GM2	CPLATE axial stress in direction 2													
1188	CPL81MS	CPLATE minimum, stress, margin of safety													
1189	CPL81ZU1	CPLATE shear stress													
1190	CPL81TMP	CPLATE temperature													
1191	CPL9THICK	CPLATE total thickness													
1192	CPL9MS	CPLATE minimum, thermal, margin of safety													
1193	DELX	Delta-X coordinate for nodes													
1194	DELY	Delta-Y coordinate for nodes													
1195	DELZ	Delta-Z coordinate for nodes													
1196	GFL9LPM4	CPLATE angle between x and material axis 1													
1197	GFL91	CPLATE auxiliary node 1													
1198	GFL92	CPLATE auxiliary node 2													
1199	GFL93	CPLATE auxiliary node 3													
1200	GFL94	CPLATE auxiliary node 4													
1201	GFL95	CPLATE auxiliary node 5													
1202	GFL9	CPLATE user-id number													

Table continued on next page

Table 156-1. Directory of ATLAS Data-Component Labels and Standard Label-Subset (LSUB) Names (Cont'd.)

NO.	LABEL	DATA COMPONENT	LSUB NAMES														
			GRADE	DISGRD	DISNODE	DISPRINT	KGRID	KPROP	LOADAB	NGRID	NODES	SMS	STRESS	STRESSINT	TMS	WGF	WMOE
(203)	GPLM1	GPLATE material code number															
(204)	GPLM1	GPLATE bending moment 1															
(205)	GPLM2	GPLATE bending moment 1-2															
(206)	GPLM2	GPLATE bending moment 2															
(207)	GPLN1	GPLATE structural node 1															
(208)	GPLN2	GPLATE structural node 2															
(209)	GPLN3	GPLATE structural node 3															
(210)	GPLN4	GPLATE structural node 4															
(211)	GPLSIGM1	GPLATE membrane axial stress in direction 1															
(212)	GPLSIGM2	GPLATE membrane axial stress in direction 2															
(213)	GPLSIGM3	GPLATE minimum stress, margin of safety															
(214)	GPLTAD12	GPLATE membrane shear stress															
(215)	GPLTEMP	GPLATE temperature															
(216)	GPLTMS	GPLATE minimum, thermal, margin of safety															
(217)	GPLT-BEN1	GPLATE bending thickness at node A1															
(218)	GPLT-BEN2	GPLATE bending thickness at node A2															
(219)	GPLT-BEN3	GPLATE bending thickness at node A3															
(220)	GPLT-BEN4	GPLATE bending thickness at node A4															
(221)	GPLT-BEN5	GPLATE bending thickness at node A5															
(222)	GPLT-MEM1	GPLATE membrane thickness at node A1															
(223)	GPLT-MEM2	GPLATE membrane thickness at node A2															
(224)	GPLT-MEM3	GPLATE membrane thickness at node A3															
(225)	GPLT-MEM4	GPLATE membrane thickness at node A4															
(226)	GPLT-MEM5	GPLATE membrane thickness at node A5															
(227)	LCC1A9	Loadcase correspondence table															
(228)	LCC1A9	Loadability diagram (payload-loadability) data															
(229)	LCC1A9	Local nodal-coordinate systems table															
(230)	WMA1	Mass BEAM node defining elastic axis location at end (1)															
(231)	WMA2	Mass BEAM node defining elastic axis location at end (2)															
(232)	WMA3	Mass BEAM user-id															
(233)	WMA1D	Mass BEAM structural node rigidly connected to end (1)															
(234)	WMA1	Mass BEAM structural node rigidly connected to end (2)															
(235)	WMA2	Mass BEAM structural node rigidly connected to end (1)															
(236)	WMA2	Mass BEAM structural node rigidly connected to end (2)															
(237)	WMA1	Mass COVER user-id															
(238)	WMA1	Mass COVER mid-surface node at corner 1															
(239)	WMA2	Mass COVER mid-surface node at corner 2															
(240)	WMA3	Mass COVER mid-surface node at corner 3															
(241)	WMA4	Mass COVER mid-surface node at corner 4															
(242)	WMA5	Mass COVER mid-surface node at corner 5															
(243)	WMA6	Mass COVER mid-surface node at corner 6															
(244)	WMA7	Mass COVER mid-surface node at corner 7															
(245)	WMA8	Mass COVER mid-surface node at corner 8															
(246)	WMA9	Mass COVER mid-surface node at corner 9															
(247)	WMA10	Mass PLATE user-id															
(248)	WMA1	Mass PLATE node at corner 1															
(249)	WMA2	Mass PLATE node at corner 2															
(250)	WMA3	Mass PLATE node at corner 3															
(251)	WMA4	Mass PLATE node at corner 4															
(252)	WMA5	Mass PLATE node at corner 5															
(253)	WMA6	Mass PLATE node at corner 6															
(254)	WMA7	Mass PLATE node at corner 7															

Table continued on next page

Table 156-1. Directory of ATLAS Data-Component Labels and Standard Label-Subset (LSUB) Name. (Cont'd.)

NO.	LABEL	DATA COMPONENT	LSUB NAMES														
			BDODE	DISGRID	DISNODE	DISPRINT	KGRID	KPROP	LOADAB	MGRID	NODES	SNS	STRESS	STREPRINT	TMS	VCF	VNODE
(254)	PLAB	Mass PLATE node at corner 8															
(255)	PLAC	Mass PLATE node at corner 9															
(256)	PLIC	Mass ROD user-id															
(257)	PLDI	Mass ROD node at end (1)															
(258)	PLDJ	Mass ROD node at end (2)															
(259)	PLDK	Mass SCALAR user-id															
(260)	PLDL	Mass SCALAR node 1															
(261)	PLDM	Mass SPAR user-id															
(262)	PLDN	Mass SPAR mid-surface node at end (1)															
(263)	PLDO	Mass SPAR mid-surface node at end (2)															
(264)	PLDP	Node analysis frame															
(265)	PLDQ	Node user-id number															
(266)	PLDRA	PLATE angle between x and material axis 1															
(267)	PLDRB	PLATE auxiliary node at corner 1															
(268)	PLDRD	PLATE auxiliary node at corner 2															
(269)	PLDRE	PLATE auxiliary node at corner 3															
(270)	PLDRF	PLATE auxiliary node at corner 4															
(271)	PLDRG	PLATE angle between x and stiffener direction S1															
(272)	PLDRH	PLATE user-id number															
(273)	PLDRI	PLATE material code number															
(274)	PLDRJ	PLATE structural node 1															
(275)	PLDRK	PLATE structural node 2															
(276)	PLDRL	PLATE structural node 3															
(277)	PLDRM	PLATE structural node 4															
(278)	PLDRN	PLATE stiffener axial stress along direction S1															
(279)	PLDRP	PLATE stiffener axial stress along direction S2															
(280)	PLDRQ	PLATE axial stress along direction 1															
(281)	PLDRR	PLATE axial stress along direction 2															
(282)	PLDRS	PLATE minimum, stress, margin of safety															
(283)	PLDRU	PLATE shear stress															
(284)	PLDRV	PLATE temperature															
(285)	PLDRW	PLATE minimum, thermal, margin of safety															
(286)	PLDRX	PLATE smeared uniaxial thickness in direction S1															
(287)	PLDRY	PLATE smeared uniaxial thickness in direction S2															
(288)	PLDRZ	PLATE thickness															
(289)	PLDR1	ROD cross-section area at end (1)															
(290)	PLDR2	ROD cross-section area at end (2)															
(291)	PLDR3	ROD user-id number															
(292)	PLDR4	ROD material code number															
(293)	PLDR5	ROD structural node at end (1)															
(294)	PLDR6	ROD structural node at end (2)															
(295)	PLDR7	ROD axial force															
(296)	PLDR8	ROD axial stress at node N1															
(297)	PLDR9	ROD axial stress at node N2															
(298)	PLDR0	ROD minimum, stress, margin of safety															
(299)	PLDR1	ROD temperature															
(300)	PLDR2	ROD minimum, thermal, margin of safety															
(301)	PLDR3	Mode x-rotation for stress loadcase															
(302)	PLDR4	Mode x-rotation for buckling mode shape															
(303)	PLDR5	Mode x-rotation for vibration mode shape															

Table continued on next page

Table 156-1. Directory of ATLAS Data-Component Labels and Standard Label-Subset (LSUB) Names (Cont'd.)

NO.	LABEL	DATA COMPONENT	LSUB NAMES														
			BDODE	DISGRD	DISMODE	DISPRINT	KGRID	KPROP	LOADS	ACRID	NODES	SMS	STRESS	STRESSPRINT	TMS	VCVF	WMODE
(304)	RY	Node y-rotation for stress loadcase	X														
(305)	RYWMODE	Node y-rotation for buckling mode shape	X														
(306)	RYVMODE	Node y-rotation for vibration mode shape															
(307)	RZ	Node z-rotation for stress loadcase	X														
(308)	RZWMODE	Node z-rotation for buckling mode shape	X														
(309)	RZVMODE	Node z-rotation for vibration mode shape															
(310)	SCA1	SCALAR auxiliary node 1	X														
(311)	SCA2	SCALAR auxiliary node 2	X														
(312)	SCA1	SCALAR nodal force component 1															
(313)	SCA2	SCALAR nodal force component 2															
(314)	SCA3	SCALAR nodal force component 3															
(315)	SCA1	SCALAR user-id number															
(316)	SCAK(R1)	SCALAR rotational stiffness about x axis															
(317)	SCAK(R2)	SCALAR rotational stiffness about y axis															
(318)	SCAK(R3)	SCALAR rotational stiffness about z axis															
(319)	SCAK(T1)	SCALAR translational stiffness in x direction															
(320)	SCAK(T2)	SCALAR translational stiffness in y direction															
(321)	SCAK(T3)	SCALAR translational stiffness in z direction															
(322)	SCAK(M1)	SCALAR nodal moment component 1															
(323)	SCAK(M2)	SCALAR nodal moment component 2															
(324)	SCAK(M3)	SCALAR nodal moment component 3															
(325)	SCAN1	SCALAR structural node 1	X														
(326)	SCA4	SCALAR minimum stress margin of safety															
(327)	SCA5	SCALAR minimum thermal margin of safety															
(328)	SPL1	SPLATE auxiliary node 1	X														
(329)	SPL2	SPLATE auxiliary node 2	X														
(330)	SPL3	SPLATE auxiliary node 3	X														
(331)	SPL4	SPLATE auxiliary node 4	X														
(332)	SPLD	SPLATE user-id number	X														
(333)	SPLM1	SPLATE material code number	X														
(334)	SPL1	SPLATE structural node 1	X														
(335)	SPL2	SPLATE structural node 2	X														
(336)	SPL3	SPLATE structural node 3	X														
(337)	SPL4	SPLATE structural node 4	X														
(338)	SPLC21	SPLATE shear-flow component 2-1															
(339)	SPLC22	SPLATE shear-flow component 2-2															
(340)	SPLC23	SPLATE shear-flow component 2-3															
(341)	SPLC41	SPLATE shear-flow component 4-1															
(342)	SPLC42	SPLATE shear-flow component 4-2															
(343)	SPLC43	SPLATE shear-flow component 4-3															
(344)	SPL	SPLATE equivalent shear flow															
(345)	SPLT	SPLATE minimum stress, margin of safety															
(346)	SPLT-MAX	SPLATE thickness															
(347)	SPLTEMP	SPLATE maximum shear stress															
(348)	SPLTMS	SPLATE temperature															
(349)	SPLM-AVG	SPLATE minimum thermal, margin of safety															
(350)	SPAFARE11	SPLATE average out-of-plane nodal load for warped plate															
(351)	SPAFARE21	SPAR upper flange area at end (1)															
(352)	SPAFAREZ1	SPAR lower flange area at end (1)															
(353)	SPRA-LWP11	SPAR upper flange area at end (2)															
(354)	SPRA-LWP1U	SPAR lower flange area at end (1)															
(355)	SPRA-LWP2L	SPAR upper lumping factor at end (1)															
(356)	SPRA-LWP2U	SPAR lower lumping factor at end (2)															

Table continued on next page

Table 156-1. Directory of ATLAS Data-Component Labels and Standard Label-Subset (LSUB) Names (Cont'd.)

NO.	LABEL	DATA COMPONENT	LSUB NAMES												
			DISC	DISNODE	DISPRINT	KPROP	LOADAB	ACRID	NODES	SMS	STRESS	STRESSPRINT	TMS	VCVF	WMODE
1351	SPR-LMPD	SPAR upper lumping factor at end (2)													
1352	SPR-ID	SPAR user-id number													
1353	SPR-MAT	SPAR material code number													
1354	SPR-N	SPAR structural mid-surface node at end (1)													
1355	SPR-N2	SPAR structural mid-surface node at end (2)													
1356	SPR-OFF	SPAR lower offset at end (1)													
1357	SPR-OFF2	SPAR lower offset at end (2)													
1358	SPR-AVG	SPAR upper-flange average load													
1359	SPR-AVG2	SPAR lower-flange average load													
1360	SPR-AVG3	SPAR upper-flange average load in lumped areas													
1361	SPR-AVG4	SPAR lower-flange average load in lumped areas													
1362	SPR-EQV	SPAR equivalent shear flow													
1363	SPR-FLG	SPAR upper-flange average stress													
1364	SPR-FLG2	SPAR lower-flange average stress													
1365	SPR-MIN	SPAR minimum stress, margin of safety													
1366	SPR-MIN2	SPAR minimum stress, margin of safety													
1367	SPR-TMP	SPAR temperature													
1368	SPR-WEB	SPAR web thickness													
1369	SPR-X-AREA	SPAR cross-section area at end (1)													
1370	SPR-X-AREA2	SPAR cross-section area at end (2)													
1371	SPR-USER	SPAR user-id number													
1372	SPR-ROD	SPAR rod material code number													
1373	SPR-ROD2	SPAR structural node at end (1)													
1374	SPR-ROD3	SPAR structural node at end (2)													
1375	SPR-SHEAR	SPAR shear node													
1376	SPR-AXIAL	SPAR axial force at node 1													
1377	SPR-AXIAL2	SPAR axial force at node 2													
1378	SPR-SHEAR2	SPAR shear flow along the rod													
1379	SPR-MIN3	SPAR minimum stress, margin of safety													
1380	SPR-TMP2	SPAR temperature													
1381	SPR-MIN4	SPAR minimum, thermal, margin of safety													
1382	SPR-MIN5	SPAR minimum, thermal, margin of safety													
1383	SPR-MIN6	SPAR minimum, thermal, margin of safety													
1384	SPR-MIN7	SPAR minimum, thermal, margin of safety													
1385	SPR-MIN8	SPAR minimum, thermal, margin of safety													
1386	SPR-MIN9	SPAR minimum, thermal, margin of safety													
1387	SPR-MIN10	SPAR minimum, thermal, margin of safety													
1388	SPR-MIN11	SPAR minimum, thermal, margin of safety													
1389	SPR-MIN12	SPAR minimum, thermal, margin of safety													
1390	SPR-MIN13	SPAR minimum, thermal, margin of safety													
1391	SPR-MIN14	SPAR minimum, thermal, margin of safety													
1392	SPR-MIN15	SPAR minimum, thermal, margin of safety													
1393	SPR-MIN16	SPAR minimum, thermal, margin of safety													
1394	SPR-MIN17	SPAR minimum, thermal, margin of safety													
1395	SPR-MIN18	SPAR minimum, thermal, margin of safety													
1396	SPR-MIN19	SPAR minimum, thermal, margin of safety													
1397	SPR-MIN20	SPAR minimum, thermal, margin of safety													
1398	SPR-MIN21	SPAR minimum, thermal, margin of safety													
1399	SPR-MIN22	SPAR minimum, thermal, margin of safety													
1400	SPR-MIN23	SPAR minimum, thermal, margin of safety													
1401	SPR-MIN24	SPAR minimum, thermal, margin of safety													
1402	SPR-MIN25	SPAR minimum, thermal, margin of safety													
1403	SPR-MIN26	SPAR minimum, thermal, margin of safety													
1404	SPR-MIN27	SPAR minimum, thermal, margin of safety													
1405	SPR-MIN28	SPAR minimum, thermal, margin of safety													

Table 156-2. Summary of Subset-Definition Records

Reference Page	Data Records
156.2	BEGIN SUBSET DEFINITION
156.2	SUBSETS OF $\left\{ \begin{array}{l} \text{STIFFNESS SET } S_e \\ \text{MASS SET } S_e \\ \text{MODAL SET } S_e \\ \text{LABELS} \end{array} \right\}$
156.3	$S_n = \{ \text{Subset-definition expression} \}$
156.4	END SUBSET DEFINITION
NODE SUBSET DEFINITIONS	
156.5	$N_{xxx} = \left\{ \begin{array}{l} \text{ALL} \\ \text{ATlist} \\ \text{Region} \end{array} \right\}$
156.12	$\left\{ \begin{array}{l} \text{CYLINDER } N_{n1} \text{ TO } N_{n2} \text{ } \langle \text{RADIUS } r \rangle \\ \text{SLAB } N_{n1} \text{ } N_{n2} \text{ } N_{n3} \text{ } \langle \text{TOLERANCE } t_1 \text{ } \langle t_2 \rangle \rangle \\ \text{SLAB } \left\{ \begin{array}{l} X \\ Y \\ Z \end{array} \right\} \text{ cord1 } \langle \text{T0 cord2} \rangle \\ \text{TUBE } N_{n1} \text{ } \dots \text{ } N_{nk} \text{ } \underline{\text{DIRECTION}} \left\{ \begin{array}{l} dv_1 \text{ } dv_2 \text{ } dv_3 \\ N_{p1} \text{ } N_{p2} \\ N_{p1} \end{array} \right\} \langle \text{TOLERANCE } t_1 \rangle \end{array} \right\}$
156.13	
156.14	
156.15	
ORDERED NODE SUBSET DEFINITIONS	
156.6	$ON_{xxx} = \text{ATlist}$
ELEMENT SUBSET DEFINITIONS	
156.6	$E_{xxx} = \left\{ \begin{array}{l} \text{ALL} \\ \text{ATlist} \\ \text{Elements} \end{array} \right\}$
156.7	$E_{xxx} = \langle \text{CLOSED} \rangle \langle \text{ALL Elements} \rangle \{ \text{Region} \}$
LABEL SUBSET DEFINITIONS	
156.7	$\text{name} = \text{ATlist}$
SUBSETS EXTRACTED FROM OTHER SUBSETS	
156.8	$N_{xxx} = \text{NODES IN } E_{xxx}$
156.8	$N_{xxx} = ON_{xxx}$
156.8	$E_{xxx} = \langle \text{CLOSED} \rangle \langle \text{ALL Elements} \rangle \text{ IN } N_{xxx}$
SUBSETS DEFINED VIA SUBSET COMBINATIONS	
156.9	$S_n = S_{n1} \left\{ \begin{array}{l} U \\ I \end{array} \right\} S_{n2}$
156.9	$S_n = C S_{n1}$
156.9	$S_n = S_{n1}$
156.9	$S_n = \{ \text{Set-algebra expression} \}$
SUBSET MODIFICATIONS	
156.10	EXCLUDE ATlist1 <FROM ATlist2>

200.0 ATLAS SYSTEM EXECUTION

Execution of the ATLAS System is directed by the user via the ATLAS Control Program as described in this section. Each directive to the System is written in the user-oriented, ATLAS-executive language.

A brief outline of the information presented in the following sections is shown below.

Section

- 200.1 --Functions and structure of the ATLAS Control Program.
- 200.2 --Format and notation used to describe all Control-Program statements in this document.
- 200.3 --Descriptions of the ATLAS-language control statements.
- 200.4 --Cataloged control-statement procedures and their use.
- 200.5 --Interactive Control-Program execution.
- 200.6 --Format and types of interactive control commands.
- 200.7 --Descriptions of the ATLAS-language, interactive control commands.

PRECEDING PAGE BLANK NOT FILMED

200.1 ATLAS CONTROL PROGRAM

The Control Program is written by the user to perform the following:

- a) Define the sequence and mode of execution of selected ATLAS System Modules (see table 10-1).
- b) Control ATLAS data-management functions (sec. 200.3.1).
- c) Perform special analyses that are not directly attainable by execution of the ATLAS System.
- d) Perform data interfaces between the ATLAS System and external programs. Established interfaces are described in appendix G.

All ATLAS-System directives are written in a user-oriented, executive language. FORTRAN and/or SNARK language statements (ref. 1-3) may be interspersed with ATLAS directives to create a Control Program to perform special analyses or to perform data interfaces with external programs. Generally, a Control Program comprised of only ATLAS-System directives is adequate to perform the required structural analysis/design tasks.

A Control Program may be executed in batch mode, in a time-shared mode, or in an interactive, time-shared mode. The System directives used in batch or time-shared executions of ATLAS are referenced as ATLAS control statements. These statements are described in sections 200.2 through 200.4. System directives used for interactive execution of ATLAS are referenced as ATLAS control commands. These commands and the various modes of interactive processing are described in detail in sections 200.5 through 200.7.

All ATLAS control statements and control commands have the following general format:

Functional Descriptor (Plist)

The "Functional Descriptor" is one or more key-words which identify one or several System Modules to be executed as the directive is processed. "Plist" is a list of parameters which is passed to the Modules as they are being executed. These parameters are used for three purposes:

- a) Select execution options of a Module.
- b) Change default values as initially set by the Module.

- c) Specify numeric or alphanumeric information to be used during execution.

The first and last statements of each Control Program must be the "BEGIN CONTROL" and "END CONTROL" ATLAS-directives, respectively, as described in section 200.3. Other ATLAS statements which control functions such as data handling are also described in that section. The remaining 200-series sections of this document describe the control directives and parameters therein that are oriented toward particular technologies (see table 10-1). Catalogs of ATLAS statements to control particular technical analyses are available to assist the user in assembling a Control Program. These cataloged control procedures are described in section 200.4.

The following example Control Program to perform a stress analysis illustrates the use of some of the ATLAS statements. Other examples are shown in appendix F.

```
BEGIN CONTROL PROGRAM EXAMPLE
PROBLEM ID (TYPICAL FORMAT OF A CONTROL PROGRAM)
READ INPUT
PRINT INPUT (NODAL)
PRINT INPUT (STIFFNESS)
PERFORM STRESS
PRINT OUTPUT (STRESSES)
SAVE FILES
END CONTROL PROGRAM
```

The problem-definition data deck is preprocessed via the "READ INPUT" statement prior to requests for formatted printout of the Nodal and Stiffness input data. Stress-analysis computations are processed via the "PERFORM" cataloged control procedure. Options were selected to generate stress-data printout and to save all ATLAS data for subsequent restart of problem execution. It should be noted that all postprocessing activities (i.e. data printout and data saving) are performed only if requested.

During problem execution, warning and/or error conditions may be encountered. A warning condition occurs when an ambiguity in the data is detected which can be resolved without user interaction. In this case, a warning message is issued and processing continues without interruption. An error condition occurs either when an inconsistency in the problem data is detected which cannot be resolved by the System or when an error is detected by the computer operating system. In this case, an error message is issued and further processing of the current statement is aborted. At this point, only the statements included in an optional, user-defined "ERROR PROCEDURE" are executed (see sec. 200.3) prior to terminating the job.

200.2 CONTROL-STATEMENT FORMATS

Throughout the 200-series sections of this document, ATLAS language statements are enclosed within rectangular boxes to distinguish them from descriptions of the statements. Within these boxes, words that are typed in all upper case letters are key-words and must be input exactly as shown. The following syntax is used to define the ATLAS language statements and parameters therein:

- a) { } --Parameters enclosed by braces have optional input formats. One of the indicated options must be selected.
- b) < > --Parameters enclosed by brackets have default values. If the default is acceptable, the corresponding parameter or parameters need not be input. All default values are defined in the descriptions of the ATLAS statements.
- c) <n> --Optional statement number ($1 \leq n \leq 70000$) in columns 1 through 5 which may be assigned to an ATLAS statement.
- d) Processor } --Name of an ATLAS Processor or
Postprocessor) Postprocessor which must be one of the following key-words (see table 10-1):

ADDINT	FLUTTER	MERGE
AF1	FREEBODY	MULTIPLY
BC	GRAPHICS	NODAL
BUCKLING	INTERACT	REACTION
CHOLESKY	INTERPOLATION	RHO3
DESIGN	LOADS	STIFFNESS
DUBLAT	MACHBOX	STRESS
EXTRACT	MASS	VIBRATION
FLEXAIR	MATERIAL	

- e) Filename --Name of an ATLAS System data file which must be one of the following key-words:

ADDIRNF	DUBLRNF	MASSRNF
AF10RNF	EXTRRNF	MERGRNF
BUCKRNF	FLEXRNF	MULTRNF
CHOLRNF	FLUTRNF	RHO3RNF
CONTRNF	INTERNF	STIFRNF
DATARNF	LOADRNF	STRERNF
DESIRNF	MACHRNF	VIBRRNF

The data on a particular "Filename" are generated by the "Processor" associated with the first four letters of that Filename except for the DATARNF and CONTRNF files. All input data interrogated by the ATLAS preprocessors are stored on file DATARNF, whereas CONTRNF is available to the user for data manipulation within a Control Program. A list of these key-words which may be used with some of the statements, is denoted by the word Filenames. Entries in Filenames must be separated by commas. Blanks within Filenames are ignored.

f) Savefile --Name of an ATLAS-System, sequential data file which must be one of the following key-words:

SAVESSF	SAVESS2	SAVESS4
SAVESS1	SAVESS3	

Data may be saved on one or more of these files for restart of system execution.

g) Mlist --List of matrix names (parameters) associated with a particular Filename. Entries must be separated by commas. Blanks are ignored. All matrices generated by the system are documented in reference 1-2.

h) Plist --List of parameters denoting user-selected options associated with execution of a particular module. Unless noted, the order in which parameters are listed is immaterial. Entries in Plist must be separated by commas. Blanks are ignored except within a CATlist generation parameter (see d) below).

Each parameter is either a key-word or a function indicator with the general format "CODE=(Values)." Entries within Values must be separated by commas; each entry is referenced as a parameter. If only one "Value" is specified or if "Values" is input as a generation

parameter "a TO b <BY c>" (see d) below),
the parentheses about Values need not
be input.

Examples: SET=2, LC=(10, 12, 40), STRESSES
STIFF=KRED, STURM, SUBSETS=N2 TO N6

In describing the Plist parameters, the following notation
is used:

- a) PARAM --A parameter typed in all upper case letters is a key-word. At least the underlined portion of a key-word must be input.
- b) Param --A parameter with only the leading character typed in upper case denotes that it must either be selected from a list of system key-words or that it is identical to an item (or a parameter) previously-defined by the user.
- c) param --A parameter typed in all lower case letters is defined by the user.
- d) CATlist --This word denotes that one parameter Value or a list of Values may be input for a CODE (see "Plist" above). A list may denote generation of parameter values. The options are as follows:

$$\text{CATlist} = \left\{ \begin{array}{l} a \\ (\text{list}) \\ a \text{ TO } b \text{ <BY } c \end{array} \right\}$$

The word "list" denotes a list of two or more values which must be enclosed within a pair of parentheses. Each value is referenced as a parameter. The list "a TO b <BY c>", with embedded blanks as shown, denotes the sequence of parameter values a, a+c, a+2c, ..., b. The items "a" and "b" are either positive integers or they are alphanumeric words with trailing digits (e.g., N10). The leading characters of "a" and "b," when they are alphanumeric words, must be identical. The item "c" is a positive or negative integer and the words TO and BY are input as shown. The right-

most integer component of "b," such as 18 in N18, minus the integer component of "a," such as 2 in N2, when divided by "c" (e.g., 2) must be a positive integer (the number 8 in this case). If the list-generation increment "c" is +1 or -1, the items "BY c" need not be input. One or more of the input forms enclosed within the braces may be used to define the corresponding parameters.

Execution parameters that may be specified via CATlist may be repeated within Plist (i.e., CODE=CATlist1, CODE=CATlist2, ... may be input). The user is cautioned that a maximum of 28 parameters may be specified within the parentheses of an ATLAS control statement (see sec. 200.3).

Input of CATlist is illustrated by the examples presented for ATlist data items (see sec. 100.2.2). If a list generated by CATlist is not allowed to have non-existent items, it is noted accordingly in the parameter description.

Note: Throughout this document, a parameter that is not indicated as a key-word must be input as a "decimal number" unless noted otherwise in the detailed parameter descriptions. Brackets < > and braces { } used in describing the syntax of the statements are not input. Pairs of parentheses, equal signs, commas and square brackets [], however, must be input as indicated.

e) Asterisk Name Option

--This option provides a method of implicitly referencing previously-defined explicit names (1 to 7 character alphanumeric words). If a name which contains asterisks is specified within Mlist or Plist, a list of names is generated automatically by replacement of the asterisks with characters. All resulting names will be treated in the same manner as if they were specified

explicitly. When this option may be used, it is noted accordingly in the detailed descriptions.

For example, a matrix name specified as AB***1 denotes all matrices whose names start with the letters AB and end with 1.

Another use of this option is for subset names such as Nxxx in Plist. For example, N1*2 is equivalent to the list N102, N112, ..., N192.

f) User Matrix Name

--User Matrices (see sec. 10.2) associated with certain processors are managed internally via a User-Matrix Name Catalog (ref. 1-1). Generally, the user assigns a 1 to 7 character alphanumeric word to each User Matrix generated by a particular job. If the same name is assigned to two different User Matrices, the contents of the first-generated matrix are over-written. User Matrices are generated by the BUCKLING, CHOLESKY, MERGE, MULTIPLY and VIBRATION Processors. Additionally, execution options 2 and 3 of the Mass Processor generate User Matrices.

The ATLAS, FORTRAN and SNARK statements used to create a Control Program follow the conventions of FORTRAN with some restrictions imposed by the ATLAS and SNARK language precompilers. The following rules must be followed.

- a) Columns 1-5 are blank or may contain a statement number in the range 1-70000.
- b) A statement is always written within columns 7-72.
- c) Statements longer than 66 columns may be continued onto following cards (cols. 7-72) provided a FORTRAN character other than zero is punched in column 6. Except for this case or if the card contains a user comment, column 6 is blank.
- d) Columns 73-80 may be used as the user wishes.

e) A user comment card may be included in a Control Program. Such a card must have the letter C punched in column 1. Columns 2-80 are used for comment information.

Subroutines and/or secondary overlays may be used within a Control Program. They must be written in either FORTRAN or SNARK. The following words and any of the words denoted by "Savefile" or "Filename," as defined previously, may not be used as variable names within FORTRAN or SNARK statements.

CONPARS	IPTU	PLOTFIL	SC01RIF	SCALRNF
CPTACC	ISAVERR	RESERVE	SC02RIF	SEARRNF
DIAGFIL	KCONPAR	SUBSRNF	SC03RIF	SYSRNF
FLORRNF	KERROR	SC00RNF	SC04RIF	TAPE3
GRAFSQ1	KLABEL	SC01RNF	SC00SSF	TAPE5
GRAFSQ2	KWARN	SC02RNF	SC01SSF	TAPE6
ICYCLE	KXINST	SC03RNF	SC02SSF	TAPE96
IERRCT	LABEL	SC04RNF	SC03SSF	TAPE99
IFILE	NEXTADR	SC00RIF	SC04SSF	

Additionally, variable names may not begin with any of the following character strings:

BEGI	INTE	PROB	SAVE
CHAN	J0,J1,....,J9	PURG	USER
CQ	KQ	QX	VQ
DMP	LOAD	RENA	WQ
ERRO	LQ	READ	
EXEC	PERF	RQ	
INDE	PRIN	R0,R1,....,R9	

The user is cautioned that blank common established via a Control Program is not preserved after execution of an ATLAS-language control statement. Use of blank common, FORTRAN or SNARK statements in a Control Program are not required for standard ATLAS-system executions.

200.3 ATLAS CONTROL STATEMENTS

All of the ATLAS-language control statements required for execution of the ATLAS System are defined in this section and are summarized in table 200-1. Each ATLAS statement is comprised of two leading key-words followed either by a few additional key-words or a parameter list (Mlist or Plist). A parameter list must be enclosed by a pair of parentheses. No more than 28 parameters may be specified within the parentheses.

An ATLAS statement is one of the following types:

- a) A statement used to perform general, utility-like functions;
- b) A statement with a parameter list that is oriented toward a particular technology.

200.3.1 General Control Statements

```
BEGIN CONTROL <MATRIX> PROGRAM <name>
```

This must be the first statement in a Control Program. If the program contains any SNARK statements (ref. 1-3), the word MATRIX must be included in the BEGIN statement. The "name" option permits the user to identify his program with an alphanumeric word that begins with an alphabetic letter and is comprised of 1 to 7 characters. The default name is NONAME.

If a user desires to manipulate directly any of the data generated by the system, it must be performed by SNARK and FORTRAN statements in the Control Program. Such activities constitute a more advanced use of ATLAS. All SNARK file-declaration statements are prohibited in a Control Program.

```
<n> CALL FILEADD (fet, {Filenames})
```

This statement is required to open the appropriate data files denoted by Filenames so that data matrices can be accessed. All data files are initially closed (nonaccessible) to a Control Program and they are always returned to a closed status after each ATLAS statement. Therefore, if access to one or more files is required at different stages of a Control Program, this statement must be input as many times as necessary. A maximum of 20 files may be opened by one statement of this type.

The word "fet" denotes a File-Environment-Table array. A different array of 50 words is required for each file that is to be opened. Thus, the FORTRAN statement "DIMENSION

Table 200-1. Summary of ATLAS Control Statements

EXECUTION
DATA

```

BEGIN CONTROL <MATRIX> PROGRAM < name >
< n > CALL FILEADD (fet, { Filenames } )
< n > CALL PRNTCAT
< n > CALL REQFL (coresize)
CHANGE ID (text)
< n > END <CONTROL PROGRAM>
ERROR PROCEDURE
< n > EXECUTE { Processor
               Postprocessor } <<(Plist)>>
< n > INDEX FILES <<(Filenames)>>
< n > INTERACTIVE CONTROL
< n > LOAD FILES << ( Savefile = REWIND, ) < Filenames > < , OPTION = { 1
               Savefile,                                     2 } >>>
< n > LOAD MATRIX ( < Savefile = REWIND, ) Filename, Mlist < , OPTION = { 1
               Savefile,                                     2 } >>
< n > PERFORM { Procedure } <<(Plist)>>
< n > PRINT { MATRIX
            MATRIXID } ( Filename, Mlist)
< n > PRINT { INPUT
            OUTPUT } ( { Postprocessor } < , Plist > )
PROBLEM ID (text)
< n > PURGE FILES <<(Filenames)>>
< n > PURGE MATRIX (FILE = Filename, Mlist)
< n > PURGE MATRIX <<(Mlist)>>
< n > READ INPUT << ( I = File, > < MODE1
               MODE2 > ) >>
< n > RENAME MATRIX (Old1 = new1, Old2 = new2, . . . )
< n > SAVE FILES << ( Savefile = REWIND, ) < Filenames > >>
               Savefile,
< n > SAVE MATRIX ( < Savefile = REWIND, ) Filename, Mlist)
               Savefile,
USER COMMON (list)

```

A(size)" must be included in the Control Program. The array dimension. "size," must be an integer equal to 50*n where "n" is the maximum number of files opened by a statement of this type. Examples:

```
DIMENSION A(150),B(100)
CALL FILEADD (A,DATARNF,LOADRNF,MASSRNF)
CALL FILEADD (B,CONTRNF,VIBRRNF)
```

If the data that are to be manipulated directly are stored in User Matrices, they can be accessed only by proper interrogation of the User-Matrix Name Catalog. These interrogations are performed via the ATLAS-System Library routines named INCRB, GMLKUP and GMISRT (ref. 1-1).

<n> CALL PRNTCAT

This statement allows the user to print the User-Matrix Name Catalog. The name, type and size of each User Matrix and the associated filename are displayed.

<n> CALL REQFL (coresize)

The central-memory field length is changed to "coresize" prior to executing subsequent Control Program statements. If "coresize" is input as an octal number, it must be appended with the letter B. The primary benefit to the ATLAS user is to be able to reduce the cost associated with central memory usage whenever possible. Core requirements are, in general, problem dependent and vary from module to module (see sec. 11.3). Users are cautioned that the program field length should not be adjusted higher than the field length requested on the Job Card. Additionally, this statement should be used only after another ATLAS statement in a Control Program.

CHANGE ID (text)

The problem-identification label (text) previously defined by the PROBLEM ID control statement may be changed to the "text" specified by this statement anywhere within a Control Program.

If a Control Program is loaded from magnetic tape or a permanent file, the problem identification label may be redefined via the PROBLEM ID input record. This special data record has the form:

PROBLEM ID * text *

If this record is used, it must be input immediately before the first "BEGIN Set-Name DATA" record in the data deck (see sec. 100.1).

<n> END <CONTROL PROGRAM>

This must be the last statement in a Control Program. It initiates the generation of the overlay statement, program statement, declaration statements, and the opening and closing statements of the Control Program.

ERROR PROCEDURE

This statement allows the user to define error-analysis and recovery procedures located at the end of a Control Program. Recovery code which follows this statement may include any ATLAS statements (including FORTRAN and SNARK) to be executed only if an error is encountered. It is recommended, however, that the user limit his code to output statements. If no ATLAS statements follow "ERROR PROCEDURE," the default recovery code is implicitly SAVE FILES (automatically executed).

The initialization portion of an error procedure execution automatically issues the comment:

NON STANDARD ERROR EXIT OCCURRED DURING EXECUTION OF ATLASn

This indicates an error occurred during execution of the ATLAS control statement which began with the n-th card of the source deck. It will then execute the defined (or defaulted) statement(s) which make up the error procedure. If an error occurs within the error-recovery execution, a FORTRAN EXIT is called.

<n> INDEX FILES <(Filenames)>

The names of all the matrices on Filenames are printed. The default for Filenames is all Filenames (sec. 200..).

<n> INTERACTIVE CONTROL

This statement is used to initiate interactive Control-Program processing in a time-shared mode via a remote terminal. Execution control is transferred to the terminal in the ATLAS interactive mode as described in section 200.5. At this point, Control-Program directives referenced as ATLAS control commands are entered via the key-board. Detailed descriptions of all control commands and interactive execution of ATLAS are presented in sections 200.5 through 200.7. The INTERACTIVE CONTROL statement may be used many times within a Control Program.


```
<n> LOAD FILES (<Savefile=REWIND,><FileNames><,OPTION={1}>)>  
<Savefile,>{2}
```

The contents of one file of Savefile are transferred to the data files specified by FileNames. Only the data that are available on Savefile which are directly associated with a particular Filename are transferred to that data file (ref. 1-2). The OPTION parameter denotes the action to be taken when a User Matrix to be loaded has the same name of a previously-defined User Matrix. In this case, matrix loading stops and an error message is issued if OPTION=2 is used. For OPTION=1, the default, contents of the first matrix are over-written. The default for Savefile is SAVESSF and the default for FileNames is all data files on Savefile. Read positioning within Savefile is the user's responsibility. If the key-word REWIND is used, Savefile is automatically rewound prior to reading its data contents.

```
<n> LOAD MATRIX (<Savefile=REWIND,>Filename,Mlist<,OPTION={1}>)>  
<Savefile,>{2}
```

Only the matrices identified by Mlist are transferred to the data file Filename from one file of Savefile. The Asterisk Name Option (sec. 200.2) may be used in Mlist. The OPTION parameter denotes the action to be taken when a User Matrix to be loaded has the same name of a previously-defined User Matrix. In this case, matrix loading stops and an error message is issued if OPTION=2 is used. For OPTION=1, the default, contents of the first matrix are over-written. The default for Savefile is SAVESSF. Read positioning within Savefile is the user's responsibility. If the key-word REWIND is used, Savefile is automatically rewound prior to reading its data contents.

```
<n> PRINT {MATRIX } (Filename, Mlist)  
{MATRIXID}
```

This statement allows the user to print the full matrix (MATRIX) or only the identification (MATRIXID) of each Filename matrix specified by Mlist. The Asterisk Name Option (sec. 200.2) may be used in Mlist. The matrices identified by Mlist are printed in a compacted format.

PROBLEM ID (text)

This is an optional second statement of a Control Program which allows the user to attach a label to his Control Program. A maximum of 300 characters including any blanks within the pair of parentheses may be used. The problem identification "text" is printed upon execution of each ATLAS statement. When printout is requested via the PRINT INPUT or PRINT OUTPUT statements, this "text" is also printed. Generally, only the first 80 characters of "text" are displayed. If "text" is blank or if this record is not included as a control statement, a default "text" is provided.

<n> PURGE FILES <(Filenames)>

The contents of the specified Filenames are purged via this statement. The specified data files are erased and if any User Matrices were included, the User-Matrix Name Catalog (ref. 1-1) is updated accordingly. The default for Filenames is all Filenames (sec. 200.2).

<n> PURGE MATRIX (FILE=Filename, Mlist)

The matrices identified by Mlist are purged from the specified Filename (erased from storage). The Asterisk Name Option (sec. 200.2) may be used in Mlist. User Matrix names may not be included in Mlist (see next statement).

<n> PURGE MATRIX <(Mlist)>

The User Matrices identified by Mlist are purged. The specified matrix names are removed from the User-Matrix Name Catalog (ref. 1-1) and the corresponding matrix contents are erased from storage. The default for Mlist is all generated User Matrices.

<n> READ INPUT <(I=File,> <MODE1> <MODE2>)>

This statement initiates reading and preprocessing of all the input data included in a data deck (see sec. 100.1). The parameter I=File, which has a default value of I=5LINPUT, identifies the name of the input-data file. If 5LINPUT is not used, one of the Savefiles (SAVESSF or SAVESS1 through SAVESS4) should be assigned to File (e.g. I=SAVESS2). The optional keyword MODE1 or MODE2 defines which data-record input format, as discussed in section 100.2.1, is to be assumed initially. The default input mode is MODE1 (each record is terminated with a slash). Multiple data decks included in a single job are read and preprocessed by multiple statements of this type.

<n> RENAME MATRIX (Old1=new1,Old2=new2, ...)

This statement allows the user to change the names previously assigned to User Matrices. The old names (Old1, Old2, ...) are replaced by the new names (new1, new2, ...). The User-Matrix Name Catalog (ref. 1-1) is updated accordingly.

**<n> SAVE FILES (< Savefile=REWIND, > < Filenames > >
< Savefile,**

The contents of the specified Filenames are copied onto the specified Savefile in SNARK MATRIX1 format (ref. 1-3). One record is written for each matrix that is saved. An additional record is written for each partition of a saved User Matrix. No end-of-file is written on Savefile via this statement. The default for Savefile is SAVESSF and the default for Filenames is all generated data files. The statement "SAVE FILES" causes all generated data files to be copied onto SAVESSF. Positioning within Savefile is the user's responsibility. If the key-word REWIND is used, Savefile is automatically rewound prior to writing data on it.

**<n> SAVE MATRIX (< Savefile=REWIND, > Filename, Mlist)
< Savefile,**

Only the Filename matrices identified by Mlist are copied onto the specified Savefile in SNARK MATRIX1 format (ref. 1-3). The Asterisk Name Option (sec. 200.2) may be used in Mlist. One record is written for each matrix that is saved. An additional record is written for each partition of a saved User Matrix. No end-of-file is written on Savefile via this statement. The default for Savefile is SAVESSF. Positioning within Savefile is the user's responsibility. If the key-word REWIND is used, Savefile is automatically rewound prior to writing data on it.

USER COMMON (list)

This statement or the equivalent FORTRAN COMMON statement "COMMON/USERCOM/list" may be required when the Control Program includes FORTRAN statements. Any Control Program variable whose value is set prior to an ATLAS statement and which is required subsequently must be declared in "list." This is required because the Control Program is reloaded after execution of each ATLAS statement. Any DO-loop, for example, which includes an ATLAS statement within its range must have the DO-loop index in "list." A maximum of 64 variables may be specified in "list." A variable included in "list" may not be initialized by a FORTRAN DATA statement in the Control Program. Additionally, this type of statement may only be used once per Control Program.

200.3.2 Technically-Oriented Control Statements

Each parameter list, Plist, associated with the following three ATLAS statements is oriented towards a particular technology. Detailed descriptions of these Plist parameters are presented in the following 200-series sections of this document.

```
<n> EXECUTE { Processor } <(Plist)>
             { Postprocessor }
```

This statement initiates execution of the specified Processor or Postprocessor. Values of the Plist parameters denote execution options associated with the identified module.

```
<n> PERFORM [Procedure] <(Plist)>
```

This statement is automatically replaced by the sequence of Control-Program statements that are associated with the identified Procedure. A procedure (catalog of control statements) may include FORTRAN and SNARK statements and any other ATLAS statements (including other PERFORM statements). The user may select one of the standard procedures made available by the ATLAS System or he may supply his own procedure to perform the required analysis functions. Use of this statement and cataloged control procedures are described in section 200.4.

```
<n> PRINT { INPUT } ({Postprocessor} <,Plist>)
          { OUTPUT }
```

This statement initiates execution of the specified Postprocessor. Printout of the input data (INPUT) or the output data (OUTPUT) associated with a particular Processor are requested by a statement of this type. Print parameters specified via Plist designate which data are to be printed and how they are to be displayed.

In addition to using ATLAS, FORTRAN and SNARK statements to create a Control Program, any of the routines included in the ATLAS-System library may also be used. Some of these routines are used to interface ATLAS with several computer programs that are external to ATLAS. Other routines, for example, provide data-management functions applicable to modification of the structural-model input data. These interfaces and certain data-management functions are described in appendix G. Descriptions of all of the ATLAS-library routines are presented in reference 1-1. Use of the ATLAS-library routines and use of FORTRAN and SNARK statements are only required when analyses which are not directly attainable by the ATLAS System Modules are to be performed.

200.4 CATALOGED CONTROL PROCEDURES

A catalog of ATLAS, FORTRAN and/or SNARK language statements may be used to define a complete Control Program or portions thereof. Standard control procedures are available for performing certain standard analyses of stiffness and mass SET/STAGE structural models and substructure models. Additionally, the user may write his own control procedures for use in assembling a Control Program to perform special functions. Example Control Programs which use some of the standard control procedures are illustrated in appendix F.

200.4.1 Standard Control-Procedures for SET/STAGE Models

The following ATLAS statement may be used in a Control Program for analyses of stiffness and mass SET/STAGE structural models:

```
<n> PERFORM {Procedure} <(Plist)>
```

Procedure

The name of one of the standard SET/STAGE cataloged procedures. The available options are as follows:

DESIGN	M-REDUCE	STRESS
F-REDUCE	REDUCE	
K-REDUCE	R-STRESS	

The functions of each of these procedures are presented in appendix E.

The optional parameters that may be included in Plist are as follows:

Key-Word=value

This parameter allows the user to change the "value" of an ATLAS statement parameter specified via "Key-Word" that is included in the selected "Procedure." Each ATLAS statement within Procedure that has a parameter identified by Key-Word is changed automatically to reflect the specified "value." The parameter formats "a" and "a TO b <BY c>" may be used to specify "value." A "value" with embedded commas (e.g., "Key-Word=(values)," may not be used. Several parameters of this type may be included in Plist.

The parameters introduced by the keywords SET, STAGE, LUMP, LC and MATERIAL are most commonly changed by this option.

[Olds]=[News]

This parameter allows the user to replace a string of characters specified by Olds with a new character string specified by News. Each ATLAS statement within Procedure that includes Olds in its parameter list is modified automatically. If a character string is to be deleted, News should be input as a blank. Otherwise, all blanks included in the specified strings are ignored except those in a parameter list "a TO b <BY c>." The indicated brackets [] must be input. Several parameters of this type may be included in Plist.

Example: Assume element stresses are to be calculated for stiffness data set 5 and boundary condition stage 2 and the matrix of reaction forces is to be named REACT. The appropriate PERFORM statement to initiate execution of the standard procedure STRESS (see appendix E) would assume the following format.

PERFORM STRESS (SET=5, [R31]=[REACT], STAGE=2)

200.4.2 Standard Control-Procedures for Substructure Analyses

The following forms of the PERFORM statement may be used in a Control Program to reference standard procedures for analyses of substructure models.

<n> PERFORM {Procedure} ({Type,} SS=Subs)

Procedure

The name of one of the standard substructure cataloged procedures. The available options are:

<u>SS-MERGE</u>	<u>SS-VSOLVE</u>
<u>SS-REDUCE</u>	<u>SS-PARTITION</u>
<u>SS-SSOLVE</u>	<u>SS-BACKSUBSTITUTION</u>

The functions of each of these procedures are described in detail in appendices D and E.

Type The name of a matrix type on which the function associated with the selected Procedure is to be performed. The options are:

STIFFNESS MASS LOADS

This parameter is applicable only to the SS-MERGE and SS-REDUCE procedures for which one or more of these key-words may be specified.

SS=CATlist ATLAS list of substructure numbers for which the activity denoted by Procedure is to be performed.

Example: If the structural stiffness matrices for substructures 2, 8, 10, 12, 15 and 16 are to be assembled and reduced, the following statements could be used within a Control Program:

```
PERFORM SS-MERG (STIFFNESS, SS=(2, 8 TO 12 BY 2, 15, 16))
PERFORM SS-REDU (STIFFNESS, SS=(2, 8 TO 12 BY 2, 15, 16))
```

200.4.3 User-Defined Control Procedures

A user-written catalog of Control Program statements may include ATLAS, FORTRAN and/or SNARK language commands in addition to PERFORM statements that reference other user-written or system-standard procedures. A procedure is established in the following manner:

- a) The first statement of each procedure must be the word PROCEDURE, beginning in column 1, followed by at least one blank and the procedure name. This identifier is an alphanumeric word with 1 to 10 non-blank characters. The first 3 characters should not be "SS-."
- b) Subsequent statements are those which are to be retrieved when the corresponding PERFORM statement is included in a Control Program.
- c) A single-record file identified as USERPRO must be made available at job execution time. This file must contain at least all of the cataloged procedures referenced by the Control Program that is to be executed (see sec. 11.2).

The following forms of the PERFORM statement may be used in a Control Program to reference user-defined cataloged statements.

```
<n> PERFORM {Procedure} <{Plist}>
```

Procedure The name of a control procedure
 previously defined by a user.

The optional parameters that may be included in Plist are as follows:

<Key-Word=value> These parameters are the same as defined
<[Olds]=[News]> for the standard SET/STAGE cataloged
 procedures (see sec. 200.4.1).

Example: Assume the user has created a file of cataloged-procedures equivalenced to USERPRO at execution time on which is included the following:

```
PROCEDURE STRE-OUT
C   PRINT DISPLACEMENTS, STRESSES AND REACTIONS
    PRINT OUTPUT(DISP,SET=1,STAGE=1)
    PRINT OUTPUT(STRE,SET=1,STAGE=1)
    PRINT OUTPUT(REACTIONS,SET=1,STAGE=1,EQCHK)
PROCEDURE SAVE-DAT
C   SAVE DATA FOR FUTURE ELEMENT-SUBSET STRESS
C   PRINTING
    SAVE FILES (DATARNF,STRERNF)
    INDEX FILES (DATARNF,STRERNF)
PROCEDURE MORE
C   ADDITIONAL PROCEDURES COULD BE ON THIS FILE.
```

The following Control Program could then be executed to perform a stress analysis of stiffness data set 3 with boundary-condition stage 2 for the load cases identified by LOD4 and LOD9. This Control Program uses the user-defined procedures STRE-OUT and SAVE-DAT as well as the standard procedure STRESS.

```
BEGIN CONTROL PROGRAM DEMO
PROBLEM ID (EXAMPLE FOR USE OF CONTROL PROCEDURES)
READ INPUT (MODE2)
PERFORM STRESS (SET=3, STAGE=5, [ALL]= [ (LOD4,LOD9) ])
PERFORM STRE-OUT (SET=3,[ STAGE=1]=[ STAGE=5,LC= (LOD4,LOD9) ])
PERFORM SAVE-DAT
ERROR PROCEDURE
END CONTROL PROGRAM
```

200.5 INTERACTIVE CONTROL-PROGRAM EXECUTION

Interactive Control-Program processing is initiated by the ATLAS control statement "INTERACTIVE CONTROL" as described in section 200.3. In this mode of system execution, the following activities may be performed via a remote terminal:

- a) Data and control files can be defined, interrogated and/or modified by edit capabilities inherent to the ATLAS Interactive Control Module.
- b) Control commands can be specified to direct execution of selected ATLAS System Modules and to control data management functions.

All interactive Control-Program directives are referenced as ATLAS control commands. The syntax of these commands is presented in sections 200.6 and 200.7.

Each control command may be executed immediately after it is input via the terminal. Alternatively, multiple commands may be entered and stored to create a "command procedure" prior to execution thereof. Command procedures may be nested (concatenated) to form a maximum of 10 procedure levels through procedure calls (RUN commands as described in section 200.7).

Three modes of interactive processing are identified:

- a) ATLAS mode --The basic mode of all interactive Control-Program processing. This mode is entered either from the Control Program via an "INTERACTIVE CONTROL" statement or from a higher-level command procedure via execution of a "RETURN" or "REVERT" command (see sec. 200.7).
- b) EDIT mode --Entered from the ATLAS mode via the "priority" command "&EDIT" to define and/or manipulate files.
- c) EXECUTE mode --Entered from the ATLAS mode via the execute symbol "#." Execution of all control commands is monitored by this mode.

The ATLAS mode is re-entered upon completion of activities in either the EDIT or the EXECUTE mode. Further descriptions of these processing modes are presented in the following sections.

200.5.1 ATLAS Interactive Mode

All interactive processing is directly or indirectly monitored from the ATLAS mode. This mode is identified by the prompt character

A>

displayed at the beginning of each line on the terminal screen. At this point, an ATLAS command is entered via the key-board. If the command is to be executed directly, the execute symbol "#" should be typed followed by depressing the CR (carriage return) key. If the command is to be stored for subsequent execution, the symbol "#" should not be input. Multiple commands entered without the "#" symbol defines a TTY command procedure.

Entry of the execute symbol "#" initiates processing of a TTY command procedure. An image of the commands is stored on the file identified by the name TEXTFIL. During execution of the procedure, control activity may be reverted to the ATLAS mode from the EXECUTE mode at user-selected interrupt points within the procedure. These options are described in section 200.5.3.

Special "priority" commands may be entered in the ATLAS mode to change the central-memory field length, to enter the EDIT mode or to display or delete the last "n" lines entered via the key-board. Priority commands are processed immediately followed by the ATLAS mode being resumed for further preparation of control commands.

200.5.2 EDIT Interactive Mode

The EDIT mode is entered from the ATLAS mode via the priority command "&EDIT." The editor operates in either the input sub-mode or the edit sub-mode identified by the prompt characters

I> or E>

The edit capabilities of the Interactive Control Module are identical to the BCS text editor (CMEDIT) capabilities which are documented in reference 200-1. For convenient refererencing, these capabilities are also described in Appendix H.

Command procedures established in the EDIT mode are referenced as FILE procedures. An image of the commands is stored on a file identified by a user-specified name. This file is subsequently processed via the "COMPILE" or "RUN" command as described in section 200.7.

200.5.3 EXECUTE Interactive Mode

A command procedure, which may contain any number of commands, is processed in two stages.

- a) Each command within the procedure, the "source" code, is interrogated and compiled (interpreted) into a pre-defined format referenced as the "object" code.
- b) The object code is executed provided the commands were compiled successfully.

The execution stage may be performed in either the FULL or STEP mode as specified by the "RUN" or "REVERT" command described in section 200.7. In the FULL mode, all commands in the procedure are processed without any interruption allowed from the terminal except via a "RETURN" command in the procedure. In the STEP mode, control returns to the terminal after processing of each command. At this point, any action may be initiated. A new command procedure, for example, may be processed prior to resuming operations on the current procedure.

The source code and object code for each procedure reside in two different sequential files. Only one procedure may reside in a particular file. When the source code is compiled, the input commands are reformatted and written onto the object file. This file is used for execution of the commands. The names of the source file and object file are used in identifying a procedure.

FILE command procedures as created in the EDIT mode are identified by source and object file names assigned by the user. The source file is identified during creation of the procedure. This file is processed either by the "COMPILE" or "RUN" command wherein the corresponding object file name is assigned.

When the execute symbol "#" is input as the last non-blank character of a command in a TTY procedure, the following sequential steps are performed:

- a) The source code for the TTY procedure is written onto the file TEXTFIL.
- b) If compilation of the procedure is successful, the object code is written onto the file OBJECTx where "x" is the display-code equivalent of the procedure level number (i.e., OBJECTA for level 1, OBJECTB for level 2, etc.).
- c) The procedure is executed in the FULL mode.

TTY and FILE procedures may be concatenated in any manner to form a maximum of 10 procedure levels at any point during execution. The driving (fundamental) procedure is the lowest-level procedure which may call higher-level procedures into execution. After returning to a lower-level procedure from a higher-level procedure during execution, the higher-level procedure and any intermediate procedures are inactivated from the procedure linkage.

The sequential files named TEXTFIL, OBJECTx and HISTFIL are reserved for use by the ATLAS Interactive Control Module. TEXTFIL and OBJECTx are used as described above. During interactive processing, an image of each control command is written onto HISTFIL prior to execution of the command. Thus, this file contains the "dayfile" information for all interactive executions.

C-5

200.6 CONTROL-COMMAND FORMATS

The commands included in a control procedure may be defined either in the TTY format or in the BATCH format as described in the following sections. In these descriptions, the notation { }, < > and Plist, as defined in section 200.2, is used. Only the control commands described in section 200.7 may be included in a command procedure. The format in which a procedure is defined is identified by the "F" parameter in the "COMPILE" and "RUN" commands.

200.6.1 TTY Command Format

The TTY format is generally used to define all command procedures. Each command is specified in free-field format in the following general form:

<label:>	Key-Words	<(Plist)>	\$
----------	-----------	-----------	----

- | | |
|-----------|---|
| label: | = Optional command label. An alphanumeric word with 1 to 5 characters followed by a colon. |
| Key-Words | = One or two key-words that identify the command. At least the 4 leading characters of each key-word must be input as shown in section 200.7. |
| Plist | = Same as defined in section 200.2. If Plist is input, the pair of parentheses must be used. |
| \$ | = This symbol denotes the end of a command. The \$-symbol may be input anywhere on a line. |

Multiple lines may be used to define a command. A new line does not indicate the beginning of a new command unless the last character of the previous line is the \$ symbol. Blanks may be interspersed anywhere within a command except in conjunction with the TTY option of Plist. In this case, at least one blank is required before and after the word TO and the word BY.

Examples:

```

READ INPUT (MODE2) $ EXE
CUIE
MASS
$ INDEX FILES $ $

```

Comment commands for user-identification purposes may be included in a procedure that is defined in the TTY format in the following ways:

- a) A command which begins with the two non-blank characters */ and ends with \$.
- b) A command with only blanks and \$.
- c) An empty command (e.g. \$ \$).

Comment commands are ignored when processing a procedure.

200.6.1.1 ATLAS-Mode Priority Commands

Special, priority commands may be used only in the ATLAS interactive mode. A priority command is executed immediately after it is entered via the key-board. Control is returned to the ATLAS mode after the specified task has been completed. Other commands typed in prior to a priority command are not destroyed. Each priority command is specified in free-field format in the following general form:

\$ Key-Word, Argument

- \$ = The ampersand character which denotes that a priority command is being entered.
- Key-Word = A key-word with 1 to 10 characters which identifies the command. Only the first character is used to distinguish the command.
- Argument = An integer number or an alphanumeric word.

Blanks interspersed within the command are ignored. The comma between Key-word and Argument, however, must be input. Each command must be fully contained on one line and it must be the only command on that line. The CR key is depressed to initiate execution of a priority command.

200.6.2 BATCH Command Format

Command procedures may be defined in the BATCH format which follows the basic conventions of FORTRAN. The general form of each command is:

<label> Key-Words <(Plist)>

label = Optional command label. An alphanumeric word with 1 to 5 characters in columns 1-5.

Key-Words } = Same as defined for the TTY format.
Plist }

The following rules must also be followed:

- a) Key-Words and (Plist) must be input within columns 7-72. If more than 66 columns are required, the command may continue onto columns 7-72 of following lines provided a non-blank character is put in column 6. Except for this case or if the line contains a user comment, column 6 is blank.
- b) Columns 73-80 may be used for comments.
- c) Comment commands may be included for user-identification purposes. Each comment must have the letter C in column 1. Columns 2-80 are used for comment information. These commands are ignored when processing a procedure.

200.7 ATLAS CONTROL COMMANDS

The ATLAS-language commands which may be used to create control procedures for interactive execution of a Control Program are summarized in table 200-2. All control statements defined in section 200.3 may be used as control commands with the following exceptions:

```
BEGIN CONTROL <MATRIX> PROGRAM <name>
CALL FILEADD (fet, {Filenames})
CALL PRNTCAT
CALL REQFL (coresize)
END <CONTROL PROGRAM>
ERROR PROCEDURE
INTERACTIVE CONTROL
PERFORM {Procedure} <{Plist}>
USER COMMON (list)
```

Additional interactive commands as described below are shown in the TTY format.

```
COMPILE <{<I=Afile,> <O=bfile,> <F= { BATCH } >}> $
      { TTY }
```

The source code for a FILE procedure as created in the EDIT interactive mode is compiled to generate the corresponding object code for subsequent execution.

I=Afile = Name of the file that contains the source code for the procedure to be compiled.
Default: I=SOURCE

O=bfile = Name of the file onto which the corresponding object code for "Afile" is to be written.
Default: O=OBJEKT

F= { BATCH } = This parameter denotes whether the source code was defined in the BATCH format or the TTY format as described in section 200.6.
 { TTY } Default: F=TTY

```
CORE (FL=n) $
```

The central-memory field length is changed to the specified value for execution of subsequent Control Program statements. The field length for processing control

commands in the interactive mode is automatically adjusted to the minimum required by the Interactive Control Module.

n = New field length specified as an octal integer.

CONTINUE \$

This command is used primarily for control branching either within a procedure or from one procedure to another.

GO TO (Label) \$

Execution control is transferred to the command identified by "Label." The search for the destination command is performed as follows:

- a) If the GOTO command is contained in a FILE procedure, only the commands in that procedure are searched.
- b) If the GOTO command is contained in a TTY procedure, the search will begin in that procedure and will continue down to, and including the highest-level FILE procedure included in the procedure linkage.

LIST CATALOG \$

The User-Matrix Name Catalog is printed. The name, type and size of each User Matrix and the associated Filename are displayed.

LIST ERROR <(Bfile)> \$

A list of all errors and warnings detected during compilation of a FILE command procedure is printed.

Bfile = The name of the object-code file previously-specified by the "O=bfile" parameter in a COMPILE or RUN command.
Default: OBJEKT

QUIT \$

The interactive processing mode is terminated. Execution control is returned to the Control Program.

RETURN \$

Execution of a procedure is interrupted temporarily by transferring control to the terminal in the ATLAS mode. Any action may be initiated at this point via key-board input. A new command procedure, for example, could be processed. Upon completion of the terminal activities, control may be returned to the initial procedure being executed by depressing the CR key on a new line. Execution is continued with the command immediately following the RETURN command within that procedure.

REVERT (<O=Code,> <M=Mode>)> \$

Execution of the procedure is terminated and control is returned to the lower-level procedure identified by "Code." All procedures at a higher level than "Code" in the procedure linkage are inactivated. Execution within "Code" will resume with the command following the last executed RUN command in that procedure.

O=Code This parameter identifies the destination procedure. The options for "Code" are:

Bfile--Revert to the procedure identified by the object-code file named "Bfile."

0--Revert to the next lower procedure in the procedure linkage. If the REVERT command is used in the lowest-level procedure, control is returned to the ATLAS interactive mode.

ALL--All active procedures are aborted and control is returned to the ATLAS interactive mode.

Default: 0=0

M=Mode This parameter defines which execution mode is to be used. The options for "Mode" are:

FULL--No terminal interrupts during execution.

STEP--Control is temporarily transferred to the terminal in the ATLAS mode after execution of each command. The end of a terminal interrupt is signified by depressing the CR key on a new line.

200.7.1 ATLAS-Mode Control Commands

The following control commands may be used only in the ATLAS interactive mode. One of these statements initiates compilation and execution of a TTY procedure, whereas the remaining commands are referenced as ATLAS-mode priority commands (see sec. 200.6.1.1).

Command \$ #

When the last non-blank character on a line containing a control command is the symbol "#," compilation and execution of the TTY procedure is initiated. "Command" may be any one of the control commands defined in section 200.7.

Example Terminal Screen Image:

```
A> READ INPUT $
A> RUN (O=STRESS) $
A> PRINT OUTPUT (STRESS, LC=DIVE) $ #
```

In this example, the 3-line TTY procedure is compiled and executed interactively.

&CORE, n

The central-memory field length is changed to the specified value for execution of subsequent Control Program statements. The field length for processing control commands in the interactive mode is automatically adjusted to the minimum required by the Interactive Control Module.

n = New field length specified as an octal integer.

Examples: &CORE, 130000
&C, 70000

&DELETE <,n>

The last "n" lines input via the key-board are deleted. If "n" is larger than the actual number of lines input, all lines are deleted. The default value for "n" is 1.

Examples: &DELETE,6
&D,4
&D

EDIT <,File>

The file identified by the name "File" is to be edited. If "File" is new, the editor is entered in the input (I>) sub-mode. If "File" is an existing file, the editor is entered in the edit (E>) sub-mode. The default for "File" is TEXTFIL which is the file onto which is written the source code for a TTY procedure.

Example: EDIT,MYPROC

PRINT <,n>

The last "n" lines input via the key-board are printed. If "n" is larger than the actual number of lines input, all lines are printed. The default value for "n" is 1.

Examples: PRINT, 8
PR, 4
P

Table 200-2. Summary of ATLAS Interactive- Control Commands

CHANGE ID (text)
COMPILE <<(I=Afile,><O=bfile,><F={
 BATCH
 TTY }>>>
CORE (FL=n)
CONTINUE
EXECUTE { Processor
 Postprocessor } <<(Plist)>>
GO TO (Label)
INDEX FILES <<(Filenames)>
LIST CATALOG
LIST ERROR <<(Bfile)>
LOAD FILES <<(Savefile=REWIND,
 Savefile,
 <Filenames><.OPTION={
 1
 2 }>>>
LOAD MATRIX (<(Savefile=REWIND,
 Savefile,
 Filename,Mlist <.OPTION={
 1
 2 }>)
PRINT { MATRIX
 MATRIXID } (Filename,Mlist)
PRINT { INPUT
 OUTPUT } ({ Postprocessor } <.Plist >)
PROBLEM ID (text)
PURGE FILES <<(Filenames)>
PURGE MATRIX (FILE=Filename,Mlist)
PURGE MATRIX <<(Mlist)>
QUIT
READ INPUT <<(I=File,><MODE1
 MODE2 >>>
RENAME MATRIX (Old1=new1,Old2=new2, . . .)
RETURN
REVERT <<(O={ Bfile
 ALL } .,><M={ FULL
 STEP
 0 }>>>
RUN (I<=Afile>,<O=bfile,><F={
 BATCH
 TTY } .,><M={ FULL
 STEP
 0 }>>
RUN <<(O=Bfile,><M={ FULL
 STEP
 0 }>>>
SAVE FILES <<(Savefile=REWIND,
 Savefile,
 <Filenames>>>
SAVE MATRIX (<(Savefile=REWIND,
 Savefile,
 Filename,Mlist)

Control Commands for Use Only in the ATLAS Mode of Interactive Processing

&_ORE,n
 &DELETE <,n >
 &EDIT <.File >
 &PRINT <.n >
 #

202.0 ADDINT MODULES

The ADDINT (ADD and/or INTERpolate) Processor may be used to combine two or more sets of Q-matrices (generalized airforce matrices) into a single set and to interpolate a single set of Q-matrices with respect to k-value (reduced frequency). Each Q-matrix within a set is associated with a particular k-value. Q-matrices to be added and/or interpolated must have been previously generated by the AF1, DUBLAT, FLEXAIR, MACHBOX, RHO3 (secs. 204.0, 216.0, 220.0, 236.0 and 250.0, respectively) or the ADDINT Processor itself and must have the following properties:

- a) The Q-matrices must all have been generated for the same Mach number (to within one percent) except that Q-matrices generated by the AF1 Processor may be used at any Mach number.
- b) The k-values corresponding with the first set of Q-matrices input to ADDINT must be, to within one percent, a subset of the k-values corresponding with the other input sets of Q-matrices. All k-values are based upon a common reference length for the purpose of comparison.
- c) All Q-matrices input to ADDINT must be based upon the same structural nodal degrees-of-freedom.

In combining sets of Q-matrices, the ADDINT Processor simply adds all the Q-matrices corresponding with each k-value associated with the first input set. Thus, the aerodynamic properties of several surfaces may be combined into a single set of Q-matrices. The input sets of Q-matrices need not have been generated by the same module and the associated reference lengths need not be the same. If any of the input Q-matrices are dependent upon the free-stream density the output matrices will exhibit the same dependency.

The interpolation option allows a new set of Q-matrices to be generated at specified k-values via a cubic-spline interpolation technique. This capability allows the large number of Q-matrices required for a flutter analysis to be generated at a reduced cost as compared to generating them directly by one of the aerodynamics processors.

The ADDINT Processor must be executed prior to execution of the FLUTTER Processor (sec. 222.0). Printout of the generalized airforce matrices generated by the ADDINT Processor may be requested from the ADDINT Postprocessor as described in section 202.2.

CONTINUED PAGE BLANK NOT FILMED

202.1 EXECUTE ADDINT STATEMENT

The following statement initiates execution of the ADDINT Processor. Previous execution of the appropriate unsteady aerodynamic processor(s) for calculation of one or more sets of Q-matrices is required.

EXECUTE ADDINT (Plist)

List of optional parameters for Plist:

**< ADD
INT >**

Key-word denoting that previously-calculated sets of Q-matrices are to be added (ADD) or that a single set of Q-matrices is to be interpolated (INT). Both key-words may be used in a single Plist in which case, previously-calculated sets of Q-matrices are added and then the resulting set of matrices is interpolated. The set of Q-matrices resulting from the summation are not saved for subsequent use when both key-words are used.

Default: INT

ID=xxxxx

Alphanumeric word with 1 to 5 characters to be used in naming the set of Q-matrices generated by the ADDINT Processor. Each Q-matrix name is comprised of the 7 characters xx:xxnn, where "nn" is automatically assigned as the corresponding number of the sequential k-value associated with the generated set.

Default: ID=GAF

A set of Q-matrices input to the ADDINT Processor is identified either via the ADDINT parameter, the parameters FLEXAIR and ALT or via a triplet of parameters, Aname, CASE and COND as presented below. The ADDINT parameter identifies a set of Q-matrices generated by a previous execution of the ADDINT Processor. The FLEXAIR and ALT parameters identify a set of Q-matrices calculated by the FLEXAIR Processor (sec. 220.0) and each triplet identifies a set of Q-matrices calculated directly by the AF1, DUBLAT, MACHBOX or RHO3 Processor (secs. 204.0, 216.0, 236.0 and 250.0). When matrices are to be summed (ADD), the ADDINT parameter and/or the FLEXAIR/ALT parameters and/or the triplet of parameters must be

repeated to identify at least two sets or up to a maximum of 20 sets of input Q-matrices. The k-values corresponding with the first set identified in this manner (left-most in Plist) are used as the basis for combining the specified sets. At least four Q-matrices must be associated with the specified set that is to be interpolated.

The following three parameters are specified in triplets.

Aname Name of the unsteady aerodynamics processor which was previously executed to calculate a set of Q-matrices. The options for "Aname" are AF1, DUBLAT, MACHBOX and RHO3.
Default: Execution is terminated if a set of Q-matrices is not identified via the other two options.

CASE=Ca Aerodynamics data case number (integer) associated with the specified COND number for which Q-matrices were calculated by an "EXECUTE Aname" statement.
Default: CASE=1

COND=Co Aerodynamic condition number (integer) specified by the same "EXECUTE Aname" statement.
Default: COND=1

The following two parameters are specified in pairs.

FLEXAIR=Xxxx Alphanumeric word assigned by the ID parameter of a previous EXECUTE FLEXAIR statement.
Default: Execution is terminated if a set of Q-matrices is not identified via the other two options.

ALT=Altitude One of the altitudes (to within one percent) associated with the "Xxxx" matrices as specified via the EXECUTE FLEXAIR statement.
Default: The first altitude associated with the "Xxxx" Q-matrices as defined via the EXECUTE FLEXAIR statement.

ADDINT=Xxxxx

Alphanumeric word assigned by the ID parameter of a previous EXECUTE ADDINT statement. This parameter identifies a set of Q-matrices to be re-processed by this module.

Default: Execution is terminated if a set of Q-matrices is not identified via one of the two foregoing options.

MACH=Mach

The Mach number greater than or equal to zero that is associated with the input Q-matrices which are to be processed. This parameter identifies which of the sets of Q-matrices identified by the CASE and COND parameters is to be used. This is required because a single execution of the DUBLAT, MACHBOX or RHO3 Processor may generate multiple sets of Q-matrices associated with more than one Mach number.

The specified value of Mach must be within one percent of the Mach number associated with each input set of Q-matrices unless they were generated by the AF1 Processor. In this case, they are defined uniquely by the CASE and COND parameters.

Default: The first Mach number associated with the first set of Q-matrices specified via Plist.

BREF=b

A reference length greater than zero to be associated with the generated Q-matrices and their k-values.

Default: The reference length associated with the first set of Q-matrices identified via Plist.

The number of Q-matrices output by the ADDINT Processor when executed with the ADD option is defined by the number of k-values associated with the first set of Q-matrices identified via Plist. The number of Q-matrices generated by this processor when executed with the INT option, however, is defined by the user via one of the four parameters KVAL, GET, UGET or IGAIN. Only one of these parameters may be specified in a single statement for the INT option.

KVAL=(list)

A list of k-values greater than zero, based on BREF, at which Q-matrices are to be generated via interpolation of an input set of Q-matrices. The specified k-values must be within the range of the k-values associated with the input set. The user is cautioned that Plist may contain a maximum of 28 parameters.

GET=nok

The number of Q-matrices (integer) to be output under the specified ID name where $2 \leq nok \leq 500$. Q-matrices are generated via interpolation to (nok-2) k-values equally spaced within the range of input k-values associated with the first set of input Q-matrices. Q-matrices associated with the maximum and minimum values of the input k-values are output under the corresponding ID as the first and last sequential matrices, respectively.

IGET=nok

This parameter is the same as the GET parameter except that the output k-values are equally spaced according to $1/k$.

IGAIN=n

An integer that denotes (n-1) new Q-matrices are to be generated at equal k-value intervals between each adjacent pair of input Q-matrices. The input Q-matrices are automatically arranged in the order of decreasing k-values. The total number of generated Q-matrices is equal to $[1-n+n*(\text{number of input Q-matrices})]$ which must be less than 500.

Default: IGAIN=20 which implies that fewer than 26 initial Q-matrices are to be processed.

202.2 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the generalized airforce matrices generated by the ADDINT Processor.

PRINT OUTPUT (ADDINT, Plist)

List of optional parameters for Plist:

ID=Xxxxx	Alphanumeric word associated with the names of a set of Q-matrices (generalized airforce matrices) generated by the ADDINT Processor which are to be printed. This word is defined via the ID parameter of an EXECUTE ADDINT statement. Default: ID=GAF
KVAL=(List)	A list of k-values identifying which of the "Xxxxx" Q-matrices are to be printed. Default: All matrices included in the specified set are printed.

204.0 AF1 AERODYNAMICS MODULES

The AF1 Processor calculates unsteady and quasi-steady aerodynamic loads on rigid or elastic, three-dimensional configurations comprised of intersecting and/or interfering nonplanar lifting surfaces in incompressible subsonic flow. The processor is based upon the modified strip theory described in reference 104-2 in which two-dimensional aerodynamic coefficients are calculated using Theodorsen coefficients (ref. 104-1), modified to account for three-dimensional effects using a static induction matrix.

Generalized airforces required for flutter analyses via the ADDINT and FLUTTER Modules (sec. 202.0 and 222.0) may be generated using the AF1 Processor (sec. 204.2). These data are calculated by a modal solution based on the aerodynamic data and structural modes defined by the user (sec. 104.0). If the AF1 Processor is executed in conjunction with the AIC option of the INTERPOLATION Processor (sec. 232.0), the generalized airforce matrices will represent Aerodynamic Influence Coefficient (AIC) matrices. Input data may be used to modify theoretically-derived aerodynamic coefficients to reflect experimental data. Other data that may be calculated by this processor include generalized lift and moment forces, stability derivatives and quasi-steady airforces.

Printout of AF1 input data and data calculated by the AF1 Processor may be requested from the AF1 Postprocessor as described in sections 204.1 and 204.3, respectively.

PREVIOUS PAGE BLANK NOT FILMED

204.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the aerodynamics input data associated with a particular AF1 data case (sec. 104.0).

PRINT INPUT (AF1, Plist)

The optional parameter for Plist is:

CASE = Ca

The aerodynamic data case number
(integer) for which input data are
to be printed.
Default: CASE=1

204.2 EXECUTE AF1 STATEMENT

The following statement initiates execution of the AF1 Processor. Previous execution of the INTERPOLATION Processor is required when structural mode shapes generated via the VIBRATION Processor are used in the aerodynamic analysis or Aerodynamic Influence Coefficients are to be generated (see sec. 104.0).

EXECUTE AF1 (Plist)

List of optional parameters for Plist:

COND=co	A condition number (integer) in the range 1 to 36 which identifies the remaining parameters of this statement. If multiple executions are required in a single job, the condition number specified by each statement of this type must be unique. Default: COND=1
CASE=Ca	The AF1 data case number (integer) to be processed. Default: CASE=1
KVAL=(list)	A list of k-values (reduced frequencies of oscillation expressed as $k = \omega(b/V)$ where ω is the circular frequency of oscillation, b is a reference length and V is the free-stream velocity). The reference length is defined by the BREF parameter. Default: Error. Execution is terminated.
BREF=b	A reference length greater than zero for reduced frequencies specified by KVAL. Default: BREF=1.0
Y = $\left\{ \begin{array}{l} \text{SYMM} \\ \text{ANTI} \\ \text{NONS} \end{array} \right\}$	Key-word denoting that a SYMMetrical, ANTIsymmetrical, or a NONSymmetrical analysis about the GLOBAL Y=0.0 reference plane is to be performed.

No contributions of the aerodynamic model in the GLOBAL negative-Y half-space are included in a NONSymmetrical analysis.

Default: Y=SYMM

TWOD

Key-word denoting that the airforces are to be calculated on the basis of two-dimensional aerodynamic theory. Static induction effects are omitted. Default: Static induction effects are included.

MOPT

Key-word denoting that the non-circulatory control surface aerodynamic stiffness derivatives are to be set to zero. Default: The theoretical derivatives are generated.

QS

Key-word denoting that only quasi-steady aerodynamic forces based on a zero reduced-frequency are to be generated. If this key-word is specified, input of the KVAL and BREF parameters is immaterial. Default: An unsteady analysis is performed.

204.3 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the aerodynamic data calculated by the AF1 Processor.

PRINT OUTPUT (AF1, Plist)

List of optional parameters for Plist:

- | | |
|---------------|---|
| CASE=Ca | The aerodynamic data case number (integer) for which generated data are to be printed.
Default: CASE=1 |
| COND=Co | The aerodynamic condition number (integer) defined by an EXECUTE AF1 parameter list for which generated data are to be printed.
Default: COND=1 |
| KVAL= (List) | A list of reduced frequency values associated with the selected COND number.
Default: Data generated for all k-values associated with COND are printed. |
| LEVEL= (List) | A list of 1 to 7 unique integers denoting which group(s) of generated data are to be printed. Each of the LEVEL indicators is as follows:

<ul style="list-style-type: none"> 1--Aerodynamic model geometry data 2--Modal deflections interpolated to the main-surface elastic axis and hinge line 3--Generalized sectional lift and moment forces 4--Generalized component lift and moment forces 5--Total generalized airforces 6--Static Induction Matrix 7--Static Lift Distribution Matrix Default: LEVEL=5 |

206.0 BOUNDARY CONDITION POSTPROCESSOR

Printout of the boundary condition (BC) data associated with a structural model may be requested from the BC Postprocessor as presented in the following section. The printed information is comprised of two lists:

- a) Nodal-freedom activity list--nodes listed in a user-selected order; each nodal freedom is tagged with one of the freedom-activity labels specified by the BC data (sec. 106.0). Cumulative and total counts of the freedoms associated with each of the activities are tabulated.
- b) Retained nodal degrees of freedom listed in the order selected via the BC input data.

If the BC data are printed prior to execution of the STIFFNESS Processor (sec. 252.2), the activity-tag on each nodal freedom is based strictly on the BC input information. However, if the BC printout is requested after execution of the STIFFNESS Processor, the nodal-freedom activity list displays all ignorable (zero stiffness) nodal freedoms as detected by the STIFFNESS Processor.

REPRODUCED FROM BLACK NOT FILMED

206.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the Boundary Condition (BC) data. The printed data may be associated with a complete nodal data set or with node subsets.

PRINT INPUT (BC, Plist)

The available options for Plist are described in sections 206.1.1 and 206.1.2.

206.1.1 Data-Set Print Option

This option allows the user to print the BC data associated with a particular nodal data set. The optional parameters for Plist are:

SET=Se The nodal data set number (integer) for which the BC input data are to be printed.
Default: SET=1

STAGE=St The BC stage number (integer) associated with data set "Se" for which data are to be printed.
Default: The BC data associated with all stages defined for data set "Se" are printed.

NODEORDER=Ord This parameter defines the ascending node-number order in which the BC data are to be printed. Optional key-words for "Ord" are:

USERID -- User number order
INPUT -- Input record number order
INTERNAL -- Internal node number order

Default: NODEORDER=USERID

206.1.2 Data-Subset-Print-Option

This option allows the user to print the BC data associated with specified subsets extracted from a nodal data set via the SUBSET-DEFINITION Preprocessor (sec. 156.0). The data associated with each subset are printed in a separate block.

SET=Se } Parameters defined as above for the data
STAGE=St } set print option.

Nxxx<=Ord> Nxxx denotes a node subset name. The BC data associated with the specified subset are printed in the order denoted by the key-word Ord:

Ord = { USFRID
INPUT
INTERNAL }

These print-orders are interpreted in a manner similar to the NODEORDER key-word for the data set print option.

Default: Ord=USERID

Several parameters of this type may be used in a single statement. A print-order remains in effect for the subsequent subset names in Plist as it is interrogated in a left-to-right order until it is reset.

The Asterisk Name Option (see sec. 200.0) may be used to identify subset names.

Default: All BC data associated with the specified nodal set and BC stage are printed.

208.0 BUCKLING MODULES

The BUCKLING Processor solves the structural, general-instability eigenproblem defined by the real system of linear, homogeneous, algebraic equations denoted by $(K+E*G)*Q=0$. The structural matrices denoted by K and G are the elastic and geometric stiffness matrices, respectively. Each of the matrices, K and G, is symmetric. Additionally, matrix K is positive-definite. Matrices E and Q contain the eigenvalues and eigenvectors associated with the buckling (critical) loads and modes, respectively.

Eigensolutions of the system of equations are effected by the following distinct steps:

- a) $K=L*L^T$; the elastic stiffness matrix is decomposed via the Cholesky square root method (ref. 208-1) into a lower triangular matrix post-multiplied by its transpose.
- b) The eigenproblem is reduced to the standard, symmetric eigenproblem denoted by $D*X=E^{-1}*X$, where $D=-L^{-1}*G*(L^T)^{-1}$
- c) Matrix D is tridiagonalized by the Householder-Givens technique (ref. 208-2) via similarity transformations.
- d) Eigenvalues are extracted from the tridiagonal matrix via either the symmetric Q-R orthogonal-matrix transformation technique or the Sturm-sequencing bisection method (ref. 208-2).
- e) Eigenvectors are calculated by the Wielandt inverse iteration technique (ref. 208-2).
- f) Orthogonalization of the calculated eigenvectors is assured by use of the Gram-Schmidt technique (ref. 208-2).
- g) Each eigenvector is normalized to its largest component.

The numerical quality of an eigensolution is measured via several checks performed by the BUCKLING Processor (sec. 208.1). In one case, each calculated eigenvalue and corresponding eigenvector are substituted into the original equation to be solved. The deviation of the i-th resulting residual vector $R(i)$ from a null vector is measured by its unbiased root-mean-square value $RMS(i)$ calculated as the square root of $[R(i)^T * R(i)] / (n-1)$ where "n" is the order of matrix D. These

quantities, which are a measure of the equilibrium of the system of equations, are printed for each eigensolution.

A second solution check, referenced as the orthogonality check, is a measure of how well the calculated eigenvector (mode) matrix succeeds in diagonalization of the elastic stiffness matrix. That is, the off-diagonal elements of the matrix calculated as $Q^T * K * Q$ are theoretically zero. These elements, however, as calculated by the triple matrix product may differ from zero and therefore, a map of these elements is automatically displayed for each eigensolution. This solution check indicates how well the mode matrix Q transforms the coupled set of equations, expressed via the physical coordinate system, into an uncoupled set of equations expressed in terms of the normal, generalized coordinate system.

Subsequent to an eigensolution, the BUCKLING Processor performs the following additional ill-conditioning check on the eigenproblem: (1) the "Norm" of the D matrix is calculated; the absolute values of the elements in the i -th row of this matrix are summed to form $s(i)$; $i=1, n$. The maximum $s(i)$ is defined as the "Norm" of the D matrix, (2) if the absolute value of the i -th calculated eigenvalue $e(i)$ is less than the quantity $(\text{Norm} * 10^{-12})$, it is automatically set to zero--the quantity within parentheses is referenced as the cutoff value. A zero eigenvalue is an adequate indicator that an error exists in the mathematical model of the physical problem.

The foregoing eigensolution checks are performed so that the user is advised of the quality of the solution. An ill-conditioned eigenproblem may require redefinition of the mathematical model. The in-plane translational stiffness at a node of a model, for example, is generally much larger than the rotational stiffness at the same node. While there is nothing incorrect about retaining both types of kinematic freedoms for buckling analyses, the rounding errors incurred when performing the required mathematical operations may lead to substantial inaccuracies in the small eigenvalues. Since the smaller eigenvalues are, in general, of primary interest, the results of a buckling analysis and the mathematical model details must be examined carefully.

Printout of the buckling analysis data may be requested from the BUCKLING Postprocessor as discussed in section 208.2. Plots of these data can also be effected via the EXTRACT and GRAPHICS Postprocessors as discussed in sections 218.0 and 228.0.

208.1 EXECUTE BUCKLING STATEMENT

The following statement initiates execution of the BUCKLING Processor. An elastic stiffness matrix and a geometric stiffness matrix for the structural model must have been generated previously. These matrices may be gross matrices as generated by the MERGE Processor (sec. 242.0) or they may be reduced matrices as generated via execution of the MERGE, CHOLESKY and MULTIPLY Processors. Reference should be made to the cataloged control statements presented in appendix E when writing a Control Program to perform general-instability analyses. Although the maximum permitted order of the eigenproblem is approximately 400, this processor provides a buckling-analysis capability for small to intermediate sized problems.

EXECUTE BUCKLING (Plist)

List of optional parameters for Plist:

BSET=xx	A buckling set number (integer) in the range 1 to 36 which is used to identify the buckling problem specified by this statement. If multiple statements of this type are required in a single job, the buckling problem associated with a particular BSET number may be redefined. "xx" should correspond to the BSET number specified via the EXECUTE STIFFNESS statement (sec. 252.2) for calculation of the element geometric stiffnesses. This allows node/freedom identifiers associated with the modal components to be displayed by the BUCKLING Postprocessor (sec. 208.2). Default: BSET=1
STIF=Name1	Name of a previously-generated, elastic, stiffness User Matrix. Default: Error. Execution is terminated.
KG=Name2	Name of a previously-generated, geometric, stiffness User Matrix. Default: Error. Execution is terminated.

The eigenproblem to be solved is illustrated by the following matrix equation:

$$(\text{Name1} + \text{EIGENxx} * \text{Name2}) * \text{MODESxx} = 0$$

The calculated eigenvalue matrix and eigenvector matrix are identified by EIGENxx and MODESxx, respectively, where "xx" is the buckling set number, right-adjusted and zero-filled. The four matrices associated with the eigenproblem are User Matrices and are managed via the User-Matrix Name Catalog (see sec. 200.0).

⟨ QR
STURM ⟩

Key-word denoting whether the QR method or the Sturm-sequence (STURM) method is to be used to extract the eigenvalues. The QR method is generally more efficient than the STURM method when five percent or more of the eigenvalues are to be calculated. The STURM method is, however, more stable and therefore allows small eigenvalues to be calculated with greater accuracy.
Default: QR

NEIGS=ne Number (integer) of eigenvalues (buckling load parameters) to be saved. It should be noted that if the QR method is used, all the eigenvalues associated with the eigenproblem are calculated automatically. Only "ne" eigenvalues are calculated, however, if the STURM method is used. The eigenvalues corresponding with the lowest absolute-valued "ne" buckling loads are saved.
Default: The order of the eigenproblem.

NMODES=nm Integer number ($0 < nm \leq ne$) of eigenvectors (modes) to be calculated and saved.
Default: NMODES=ne

208.2 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the data calculated by the BUCKLING Processor.

PRINT OUTPUT (BUCKLING, Plist)

List of optional parameters for Plist:

BSET=Num The buckling set number (integer) for which
 calculated data are to be printed.
 Default: BSET=1

NO{Data} Key-word that suppresses printing of
 selected data. One or both of the following
 optional key-words may be used in a single
 statement:

NOEIGEN -- Eigenvalues (critical loads)
NOMODES -- Modes

Default: Eigenvalues and modes are
 printed.

210.0 CHOLESKY PROCESSOR

The CHOLESKY Processor employs the Cholesky square root technique to solve a real system of linear equations which has a symmetric, positive-definite coefficient matrix (refs. 210-1 and 210-2). The solution of a system of equations denoted by $A*X=B$ is performed in the following three distinct steps:

- a) $A=L*L^T$; the coefficient matrix is decomposed into the product of a lower-triangular matrix post-multiplied by its transpose.
- b) $Y=L^{-1}*B$; a forward substitution is performed on matrix B.
- c) $X=(L^T)^{-1}*Y$; a backward substitution is performed on matrix Y to complete the solution of the original equation unknowns.

A partial or complete solution of a system of equations, as requested by the user, can be performed by a single execution of this module. That is, any one or combination of the foregoing sequential three solution steps may be performed via a single statement. Execution of this module to effect structural analyses via the ATLAS System is illustrated by the cataloged control statements presented in appendix E. These cataloged statements are a user convenience that obviates the need, in most cases, to use the explicit EXECUTE CHOLESKY statement.

PRECEDING PAGE BLANK NOT FILMED

210.1 EXECUTE CHOLESKY STATEMENT

The following statement initiates execution of the CHOLESKY Processor

EXECUTE CHOLESKY ({Operation, In1<=Out1>, Out2, In2},
Options)

The selected optional parameters within braces must appear in the noted sequence. These parameters denote a selected solution "Operation" to be performed on previously-generated matrices "In1" and "In2" according to the "Options." The results are stored in matrices "Out1" and "Out2" which are named by the user via this statement. All matrices processed by this module are User Matrices (see sec. 200.0). See the OPTION parameter below regarding uniqueness of the assigned result-matrix names. Not all the parameters shown within the braces are required for each of the options. The following table illustrates the "Operation" key-word and the User Matrix names which must be specified in the sequence noted above. Characters A, X, B, Y and L in this table denote matrix names associated with the system of equations $A*X=B$ to be solved as discussed above.

"Operation" Key-Word	User Matrix Names				Solution Steps Performed
	"In1"	"Out1"	"Out2"	"In2"	
<u>DECOMPOSE</u>	A	L			$A=L*L^T$
<u>FORWARD</u>	L		Y	B	$Y=L^{-1}*B$
<u>BACKWARD</u>	L		X	Y	$X=(L^T)^{-1}*Y$
<u>SOLVE</u>	A	<L>	X	B	$A=L*L^T; Y=L^{-1}*B; X=(L^T)^{-1}*Y$
<u>DEFO</u>	A	<L>	Y	B	$A=L*L^T; Y=L^{-1}*B$
<u>FOBA</u>	L		X	B	$Y=L^{-1}*B; X=(L^T)^{-1}*Y$
<u>IFORWARD</u>	L		Y		$Y=L^{-1}$

Multiple solution steps can be effected via the SOLVE, DEFO or the FOBA operation key-word in a single statement as noted in this table. The IFOR operation permits a forward sweep on an identity matrix. This capability is used, for example, to generate a structural flexibility matrix (see PERFORM F-REDUCE in appendix E).

For the SOLVE, DECOMPOSE and DEFO operations, matrix A must be stored in the User-Matrix triangular format to be recognized by this processor. Normally, this criterion is automatically satisfied by prior matrix operations.

If the SOLVE or DEFO operation is requested, the decomposed coefficient matrix (L), generated during processing, may be saved for subsequent use by appending parameter "In1" with an equal-sign and a name for "Out1." This matrix is stored in a format for processing only by the CHOLESKY Processor.

The two parameters denoted by "Options" are as follows:

OPTION=Code This parameter denotes the action to be taken when the name assigned to a result matrix, "Out1" and/or "Out2," is the same as the name of a previously-generated User Matrix. The options for the integer "Code" are:

- 1--The specified solution is performed and the contents of the previously-generated matrix are over-written.
- 2--The specified solution is not performed. Subsequent control statements, however, are executed. This option allows the user to avoid recalculation of previously-generated matrices as in the case of problem-execution restart or during substructure analyses.

In both cases, a warning message is issued.

Default: OPTION=1

DECAY=dec A pivot (diagonal) element decay number (integer ≥ 0). All decay numbers generated during decomposition that are greater than or equal to "dec" are printed with the associated matrix row number. In general, large decay numbers indicate an ill-conditioned mathematical model which should not be accepted for further analysis. This criterion indicates where a local mechanism or a region of weak stiffness is inherent in the structural model.

The squared diagonal terms of L are calculated by

$$L_{ii}^2 = A_{ii} - \sum_{j=1}^{i-1} L_{ij}^2 ; i > 1$$

where L_{ij} are previously-generated terms of the i -th row of matrix L. The difference of L_{ii}^2 and A_{ii} is measured by the quantity $-\log_{10} (L_{ii}^2/A_{ii})$ rounded to the nearest integer value. This value is defined as the decay number of the i -th row.

Consider, for example, the case when two rows of matrix A are almost linear dependent (matrix A is ill-conditioned but not truly singular). During generation of the latter of these two rows in matrix L, a large pivot decay would occur. If, however, the two rows are truly linearly dependent, the calculated value of the latter L_{ii}^2 quantity would be zero or negative. Since a complete solution is prohibited in this case, an ERROR message is printed which declares matrix A as being singular. If this occurs, the appropriate L_{ii} diagonal element is set to 10^{10} and complete decomposition of matrix A is performed whereupon execution is terminated. This procedure exposes all singularities and large pivot decays associated with an A matrix via a single decomposition request. The difference between a very large pivot decay number of 14 (a very ill-conditioned A matrix) and a singular condition is purely numerical since the precision of a decimal number stored by the 6600 computer is approximately 14 digits of accuracy.

A pivot decay table is printed regardless of any singularities inherent in the A matrix.

Default: DECAY=10

212.0 DESIGN MODULES

The DESIGN Processor provides a general resize (design) capability for the structural design process supported by the ATLAS System. Selected structural properties of finite-element components of a model are modified (resized) according to the requirements specified by the design data (sec. 112.0). The requirements to be satisfied during resizing can be specified as geometric and/or margin-of-safety constraints on selected elements. The design requirements resulting from a structural optimization can include thermal effects and "local" buckling effects for "plate-like" elements.

The structural optimization to be performed by the DESIGN Processor can be selected as either a fully-stressed-design based on an optimality criterion or a regional optimization of composite structure based on a math-programming approach. Options are provided for controlling the resize cycles and for influencing the convergence behavior of a fully-stressed design. These capabilities are intended for use primarily during the preliminary design activities in the large-problem environment. A technical description of these capabilities is presented in reference 212-1.

Five types of execution of the DESIGN Processor (sec. 212.2) are provided:

- a) Strength execution--A fully-stressed-design (FSD) and/or a thermal fully-stressed-design (TFSD). The FSD resizing (refs. 212-2 and 212-3) is based on margins-of-safety calculated according to Hill's strength-design criterion (ref. 212-4) and/or buckling criteria (ref. 212-5) for "plate-like" elements (COVER, PLATE and GPLATE). The TFSD resizing (ref. 212.6) for ROD,SPAR,PLATE,COVER,SROD and SPLATE elements is based on Hill's strength criterion. Thermal-design load cases must be identified by the design data (sec. 112.0) if TFSD resizing is to be performed. When non-zero margins of safety are required for selected elements, the FSD or TFSD resizing can be constrained to user-selected margin-of-safety envelopes.
- b) Flutter execution--Structural properties of selected elements are modified according to specified factors (e.g., stiffness factors for flutter studies). Margins of safety are calculated for these properties. The selected elements are then resized according to the calculated margin-of-safety envelopes. These envelopes

~~PROCESSED FROM BLACK NOT FILMED~~

may include margins of safety calculated via "strength executions." This technique presents an initial capability for structural design based on flutter stiffness-constraints.

- c) History execution--The minimum margins of safety for selected elements, as calculated via "strength" and/or "flutter" executions of the DESIGN Processor, are extracted for subsequent printing by the DESIGN Postprocessor (sec. 212.3).
- d) Composite optimization -- This type of execution is intended for composite structure modeled with CPLATE and CCOVER elements. Regions of the structural model to be strength-resized are defined by the design data (sec. 112.1.2.6). Each region is identified as a stiffness element subset which can only contain one element type. The lamina thicknesses are optimized for minimum weight according to Hill's strength criterion or the maximum strain criterion. The resized properties of the elements in the defined regions will satisfy the selected design criterion.
- e) Smoothing execution -- Properties of selected elements are modified according to the "smoothing" data subset (sec. 112.1.2.8). The input and/or calculated property values can be redefined or the property values of elements within regions can be made constant.

It should be noted that the Flutter and Smoothing execution options provided by the DESIGN Processor do not require previous calculation of element stresses. Therefore, these capabilities allow the element properties of structural models, as defined by the problem-definition input data, to be modified, as necessary.

The design process is influenced by the user through constraints specified by the design data in addition to the number of design cycles to be performed and the convergence requirements as specified via the Control Program. A design cycle, as directed by the Control Program, includes the following:

- a) Generation of the element stiffness and stress matrices by the STIFFNESS Processor (sec. 252.2);
- b) Assembly and solution of the system of linear equations for the unknown displacements by the LOADS, MERGE and CHOLESKY Processors:

- c) Calculation of the element stresses by the STRESS Processor (sec. 254.1) for selected load cases;
- d) Calculation of "new" stiffness-element properties by the DESIGN Processor;
- e) Modification of the "old" stiffness element properties by the DESIGN Processor according to the calculated and input design requirements;
- f) Calculation of the total structural weight by the MASS Processor (sec. 238.0).

Margins of safety and resized element properties calculated by the DESIGN Processor during a particular design cycle are identified by a user-specified CYCLE number. Design data associated with a CYCLE are referenced by this number.

A standard cataloged control procedure, as presented in appendix E, allows the user to conveniently create a Control Program for performing structural design. In conjunction with using this procedure, the user specifies in the Control Program how many design cycles are to be executed. Optional convergence requirements can also be specified such that if they are satisfied, remaining design cycles are not executed. The following types of convergence criteria can be invoked:

- a) Maximum allowable ratio between the "new" and "old" properties for each element type.
- b) Maximum allowable absolute change in the total weight of the model after each cycle $|W(i+1)-W(i)|$.
- c) Maximum allowable, absolute ratio of the change in total weight to the new total weight of the model after each cycle $|[W(i+1)-W(i)]/W(i+1)|$.

Printout of the design input data and data calculated by the DESIGN Processor may be requested as described in sections 212.1 and 212.3, respectively. Output data include the modified properties of the elements and design histories of minimum margins of safety for selected resize CYCLES. Plots of the resized element properties and plots of the minimum margins of safety resulting from a strength or thermal resize may be generated via the EXTRACT and GRAPHICS Postprocessors (sec. 218.0 and 228.0).

212.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the design input data (sec. 112.0). In the printout, these data are grouped according to the following data types:

<u>Data Type</u>	<u>Design Input-Data Subsets</u>
TABLE	---TABLE and MODULUS
PROPERTY	---PROPERTY, FIXED, LOWER-BOUND, UPPER-BOUND and MARGIN
SIZING	---SIZING and RESTRAIN-SIZING
LOADS	---LOADS, SUPERPOSITION and THERMAL

PRINT INPUT (DESIGN, Plist)

List of optional parameters for Plist:

TABLE	Key-word that is required if TABLE type data are to be printed. Default: TABLE data are not printed.
SET=Se	The number (integer) of the stiffness data set for which input design PROPERTY, SIZING and/or LOADS type data are to be printed. This parameter is ignored if only TABLE type data are to be printed. Default: SET=1
NOPROPERTY	Key-word that suppresses printing of the design PROPERTY type data associated with set "Se." Default: Design PROPERTY data are printed.
SIZING	Key-word that is required if SIZING type data are to be printed. Default: SIZING data are not printed.
STAGE= { St (Slist) }	A boundary condition stage number or a list thereof associated with set "Se" for which design LOADS type data are to be printed. Default: Design LOADS type data are not printed.

212.2 EXECUTE DESIGN STATEMENT

The following statement initiates execution of the DESIGN Processor.

EXECUTE DESIGN (Plist)

Each execution option of this processor (Strength, Flutter History, Composite and Smoothing) is described in sections 212.2.1 through 212.2.5. Execution of the STRESS Processor (sec. 254.1) is required prior to using the Strength and Composite options, whereas previous execution of the DESIGN Processor itself is required prior to using the History option. No other processor need be executed prior to using the Flutter or Smoothing options.

212.2.1 Strength Execution Option

The strength execution option allows the margins of safety to be calculated for structural elements by applying prescribed strength and buckling criteria. Based on the calculated margins of safety, the user may elect to resize selected elements of the structural model. The optional parameters for Plist are:

TFSD	Key-word that is required for execution of the "thermal fully-stressed-design" option (TFSD), in addition to the standard "fully-stressed-design" option (FSD). The thermal-design load cases to be included in the optimization must be defined by the design data (sec. 112.0). Selected load cases to be included in the FSD calculations are also defined by the design data. Default: The FSD option is executed.
NOFSD	Key-word that suppresses execution of the FSD option. This parameter is effective only if the key-word TFSD is used. Default: The FSD option is executed.
SET=Se	The number (integer) of the stiffness data set to be processed. Default: SET=1

STAGE=CATlist ATLAS list of boundary condition (BC) stage numbers (integers) associated with set "Se." These stages identify sets of calculated margins of safety. Default: STAGE=1

CYCLE=n Integer greater than zero that identifies the design-cycle execution for the specified SET/STAGES. If CYCLE is specified as the same number used in a previous "strength" or "flutter" execution and if resizing is requested, the element-property modifications are influenced by the previously-calculated margins of safety. Default: Error. Execution is terminated.

MODE=Start An integer denoting where the analysis is to begin. The options for "Start" are:

- 0--Start by calculating margins of safety.
- 1--Start by resizing based on previously-calculated margins of safety.

Default: MODE=0

MARGIN=Stop An integer denoting where the analysis is to terminate. The options for "Stop" are:

- 0--Stop after resizing
- 1--Stop after calculating margins of safety

Default: MARGIN=0

Note: If MODE=1, the MARGIN parameter is ignored. Regardless of whether MARGIN is set to 0 or 1, only resizing is performed.

CONVERGE=factor This parameter initiates an iterative solution of the stress ratio "R" for "plate-like" elements when gage-dependent allowable stresses are used (sec. 112.1.2.9). These "R" values are used for calculating margins of safety for resizing. The "factor" is input in the form 10.02, for example, which means that 10 iterative cycles are to be

performed for each element. The 0.02 denotes that if the solution has converged so that the tolerance of "R" is in the interval $0.98 \leq R \leq 1.02$, no more iterations are performed.
 Default: No iteration on the solution is performed.

PROCEDURE=Save An integer denoting whether the calculated margins of safety are to be saved on file DESIRNF for subsequent data postprocessing or for subsequent execution of the DESIGN Processor.
 The options for "Save" are:

- 0--Do not save the margins of safety.
- 1--Save the margins of safety.

Default: PROCEDURE=0

RESIZE=CATlist ATLAS list of boundary condition stage numbers (integers) associated with set "Se" for which element resizing is to be performed.
 Default: The stages specified by the parameter STAGE.

RELEASE= $\left\{ \begin{array}{l} N1 \\ (N1, N2, \dots) \end{array} \right\}$

An integer or a list of integers denoting which user-specified design constraint data are not to be used in the element resizing performed by this design cycle.
 The options for "Ni" are:

- 1--All design constraint data
- 2--Fixed data
- 3--Upper-Bound data
- 4--Lower-Bound data
- 5--Restrain-Sizing data
- 6--Margin data

Default: All of the design constraint data for the specified set/stage(s) are used.

212.2.2 Flutter Execution Option

The flutter execution option allows the user to factor the stiffness properties of elements included in subsets previously-defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0). Margins of safety based on these factored properties are calculated for the selected elements. The elements are then resized according to margin of safety envelopes. These envelopes are based on the factored properties and may include margins of safety previously calculated via "strength" executions. The optional parameters for Plist for this option are:

FLUTTER	Key-word which must be the first parameter in Plist to initiate this execution option. Default: The strength execution option is assumed.
SET=Se	The number (integer) of the stiffness data set to be processed. Default: SET=1
CYCLE=n	Integer greater than zero that identifies the design-cycle execution for set "Se." Resizing is performed for all boundary condition stages associated with set "Se." Default: Error. Execution is terminated.
Exxx=factor	The properties of all the elements included in element subset Exxx are to be multiplied by the positive "factor." This parameter may be repeated to identify a maximum of 14 element subsets. Default: Error. Execution is terminated.
PFOCEDURE=Save	An integer denoting whether the calculated margins of safety are to be saved on file DESIRNF for subsequent data postprocessing or for subsequent execution of the DESIGN Processor. The options for "Save" are: 0--Do not save the margins of safety 1--Save the margins of safety Default: PROCEDURE=0

MINMS=Type An integer denoting which type of margins of safety are to be used to establish the envelope of minimum margins of safety for element resizing. The options for "Type" are:

- 0--Use only the factored property margins
- 1--Use the factored property margins and previously calculated strength margins

Default: MINMS=0

212.2.3 History_Execution_Option

The history execution option allows the user to extract the minimum margins of safety for selected elements as calculated previously via "strength" or "flutter" resize cycles. Printout of the extracted history data is effected via the DESIGN Postprocessor (sec. 212.3). The optional parameters for Plist are:

HISTORY	Key-word which must be the first parameter in Plist to initiate the history execution option. Default: The strength execution option is assumed.
SET=Se	The number (integer) of the stiffness data set to be processed. Default: SET=1
STAGE=CATlist	ATLAS list of boundary condition (BC) stage numbers (integers) associated with set "Se." These stages identify groups of margins of safety for which history data are to be calculated. Default: Error. Execution is terminated.
CYCLE=CATlist	ATLAS list of previously-defined design cycles associated with the specified SET/STAGES. Default: Error. Execution is terminated.

$$\left\{ \begin{array}{l} \text{SUB} \\ \text{ELN} \\ \text{IELN} \end{array} \right\} = \left\{ \begin{array}{l} \text{N1} \\ (\text{N1}, \text{N2}, \dots) \end{array} \right\}$$

This parameter identifies one element or a list of elements, a maximum of 150, for which a resize history is to be generated. One or more of the indicated options may be used in a single statement. "Ni" is an integer that denotes one or more elements according to the selected key-word:

SUB --An element subset number (integer) associated with the specified stiffness data set as defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0).

ELN --A user element number

IELN --An internal element number

Default: If none of these parameters is specified, execution is terminated.

212.2.4 Composite Optimization Option

This execution option is used for optimization of composite structure modeled with CPLATE and CCOVER elements. The optimization problems to be solved are defined by the design data (sec. 112.0). Lamina thicknesses are optimized for minimum weight either according to Hill's strength-design criterion or according to the maximum strain criterion. Optionally, the resized properties of the elements in the isolated regions of the structure can be modified according to the "smoothing" design data. The optional parameters for Plist are:

COMPOSITE	Key-word which must be the first parameter in Plist to initiate this execution option. Default: The strength execution option is assumed.
SET=Se	The number (integer) of the stiffness data set to be processed. Default: SET=1
STAGE=St	The number (integer) of the boundary condition (BC) stage number associated with set "Se." The stage number identifies the optimization problems to be solved. Default: STAGE=1

- CYCLE=n Integer greater than zero that identifies the design-cycle execution for set "Se."
Default: CYCLE=1
- MODE=Start An integer denoting where the analysis is to begin. The options for "Start" are:
- 0--Start by performing the optimization (calculate resizing requirements)
 - 1--Start by element resizing based on a previously-executed optimization
- Default: MODE=0
- MARGIN=Stop An integer denoting where the analysis is to terminate. The options for "Stop" are:
- 0--Stop after resizing
 - 1--Stop after the optimization
- Default: MARGIN=0
- Note: If MODE=1, the MARGIN parameter is ignored. Regardless of whether MARGIN is set to 0 or 1, only resizing is performed.
- ⟨ STRAIN ⟩
HILL
- Key-words that denote whether the maximum strain criterion (STRAIN) or the Hill's criterion (HILL) is to be used during the optimization. Only one of these key-words may be specified.
Default: STRAIN
- SMOOTH Key-word denoting that the resized element properties are to be modified. A description of this option is presented in the following section (sec. 212.2.5). This parameter is ignored if only the optimization calculations are to be performed.
Default: The element properties resulting from the resizing are not modified.

212.2.5 Smoothing Execution Option

The smoothing execution option allows the properties of elements contained in element subsets to be modified. Constant properties for all the elements of a certain type that are contained within a subset are generated according to one of the following requirements.

- a) The properties of all the elements are to be the same as the corresponding property values for a specific element.
- b) The properties of all the elements are to be changed to the maximum values of the corresponding properties of the elements.
- c) The properties of all the elements are to be changed to the specified property values.

Execution of this option requires "smoothing" design data (sec. 112.1.2.8). The optional parameters for Plist for this execution option are:

SMOOTHING	Key-word which must be the first parameter in Plist to initiate this execution option. Default: The strength execution option is assumed.
SET=Se	The number (integer) of the stiffness data set to be processed. Default: SET=1
CYCLE=n	Integer greater than zero that identifies the design-cycle execution for set "Se." Default: CYCLE=1

212.3 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the resized element properties as calculated by the DESIGN Processor or to request printout of a design history as generated by a "history execution" of the DESIGN Processor.

PRINT OUTPUT (DESIGN,Plist)

The available options for Plist are described in the following sections.

212.3.1 Resized Element-Property Print Option

The resized property data printout may be associated with a complete stiffness data set or with selected element subsets previously defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0). The optional parameters for Plist are:

- | | | | | | | | | | | | | | |
|------------|---|-----------|----------|---------|---------|-----------|-----------|---------|----------|-----------|----------|-----------|-----------|
| SET=Se | The number (integer) of the stiffness data set for which resized element data are to be printed.
Default: SET=1 | | | | | | | | | | | | |
| NO{Eltype} | Key-word that suppresses printing of data for a selected element type. One or more of these key-words may be used in a single statement. The available options are: | | | | | | | | | | | | |
| | <table border="0"> <tr> <td>NORODS</td> <td>NOPLATES</td> <td>NOSRODS</td> </tr> <tr> <td>NOBEAMS</td> <td>NOGPLATES</td> <td>NOSPLATES</td> </tr> <tr> <td>NOSPARS</td> <td>NOBRICKS</td> <td>NOCPLATES</td> </tr> <tr> <td>NOCOVERS</td> <td>NOSCALARS</td> <td>NOCCOVERS</td> </tr> </table> | NORODS | NOPLATES | NOSRODS | NOBEAMS | NOGPLATES | NOSPLATES | NOSPARS | NOBRICKS | NOCPLATES | NOCOVERS | NOSCALARS | NOCCOVERS |
| NORODS | NOPLATES | NOSRODS | | | | | | | | | | | |
| NOBEAMS | NOGPLATES | NOSPLATES | | | | | | | | | | | |
| NOSPARS | NOBRICKS | NOCPLATES | | | | | | | | | | | |
| NOCOVERS | NOSCALARS | NOCCOVERS | | | | | | | | | | | |
| | Default: Data are printed for all element types not excluded from set "Se" or from the element subsets Exxx as defined below. | | | | | | | | | | | | |
| SUBSETS | Key-word that must be used if property data are to be printed for the stiffness element subsets identified by Exxx.
Default: Property data for all elements in set "Se" are printed. | | | | | | | | | | | | |
| Exxx | Element subset name defined via the SUBSET-DEFINITION Preprocessor. The resized property data for the elements | | | | | | | | | | | | |

in Exxx are printed. This parameter may be repeated.

Default: If SUBSETS is used and Exxx is not defined, no printout is generated.

212.3.2 Design History Print Option

A design history printout which includes minimum margins of safety for selected elements traced through selected load cases and design cycles may be requested by this print option. A history is established via a "history execution" of the DESIGN Processor. The optional parameters for Plist are:

HISTORY	Key-word which must be the first parameter in Plist to initiate a design history printout. Default: The resized element property print option is assumed.
SET=Se	The number (integer) of the stiffness data set for which a design history is to be printed. Default: SET=1

216.0 DUBLAT AERODYNAMICS MODULES

The DUBLAT Processor calculates unsteady aerodynamic loads on rigid or elastic, three-dimensional configurations comprised of intersecting and/or interfering nonplanar lifting surfaces and body surfaces in subsonic compressible flow. The basic analysis technique employed by this processor is similar to that used by the H7WC version of the Air Force Doublet-Lattice program (ref. 116-1). The DUBLAT Processor has refinement options to correct pressure distributions and to modify velocity profiles to account for lifting-surface thickness effects. Development of the theoretical basis for the analysis technique is presented in references 116-1 and 116-2.

Generalized airforce matrices required for flutter analyses via the ADDINT and FLUTTER Modules (sec. 202.0 and 222.0) are generated by execution of the DUBLAT Processor (sec. 216.2). These data are calculated by a modal solution based on the aerodynamic model and structural modes defined by the user (sec. 116.0). If the DUBLAT Processor is executed in conjunction with the AIC option of the INTERPOLATION Processor (sec. 232.0), the generalized airforce matrices will represent Aerodynamic Influence Coefficient (AIC) matrices. Other data optionally calculated by this processor include aerodynamic pressure-difference distributions, sectional generalized forces and static and dynamic stability derivatives (aerodynamic coefficients).

Printout of DUBLAT input data and data calculated by the DUBLAT Processor may be requested from the DUBLAT Postprocessor as described in sections 216.1 and 216.3, respectively.

PRECEDING PAGE BLANK NOT FILMED

216.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the aerodynamic input data associated with a particular DUBLAT data case (sec. 116.0).

PRINT INPUT (DUBLAT, Plist)

The optional parameter for Plist is:

CASE=Ca

The aerodynamic data case number
(integer) for which input data are to
be printed.
Default: CASE=1

216.2 EXECUTE DUBLAT STATEMENT

The following statement initiates execution of the DUBLAT Processor. Previous execution of the INTERPOLATION Processor is required when structural mode shapes generated via the VIBRATION Processor are used in the aerodynamic analysis or Aerodynamic Influence Coefficients are to be generated (see sec. 116.0).

EXECUTE DUBLAT (Plist)

List of optional parameters for Plist:

- COND=co A condition number (integer) in the range 1 to 36 which identifies the remaining parameters of this statement. If multiple executions are required in a single job, the condition number specified by each statement of this type must be unique.
Default: COND=1
- CASE=Ca The DUBLAT data case number (integer) to be processed.
Default: CASE=1
- MACH=(list) A list of 1 to 20 free-stream Mach numbers, M. Each Mach number must be in the range $0.0 \leq M \leq 1.0$.
Default: Error. Execution is terminated.
- KVAL=(list) A list of 1 to 20 k-values (reduced frequencies of oscillation expressed as $k = (\omega b/V)$ where ω is the circular frequency of oscillation, b is a reference length and V is the free-stream velocity) greater than or equal to zero. The reference length is defined by the BREF parameter.
Default: Error. Execution is terminated.
- BREF=b A reference length greater than zero for reduced frequencies specified via KVAL.
Default: BREF=1.0

Y= { SYMM
ANTI
NONS }

Key-word denoting that a SYMMetrical, ANTIsymmetrical or a NONSymmetrical analysis about the GLOBAL Y=0.0 reference plane is to be performed. No contributions of the aerodynamic model in the GLOBAL negative-Y half-space are included in a NONSymmetrical analysis.

Default: Y=SYMM

< PITCH
YAW >

Key-word denoting that the PITCH or the YAW dynamic stability derivatives (aerodynamic coefficients) are to be calculated.

Default: PITCH derivatives are calculated.

QUASI=Label

A label comprised of two alphanumeric characters denoted by "ab" which are associated with a set of quasi-inverse, normal-wash matrix names of the form QXXabKL. If quasi-inverse matrices named via this "Label" are available on file DUBLRNF, they are used in the analysis. Otherwise, they are generated, named according to the specified "Label," written onto file DUBLRNF and used in the analysis.

Calculation of these matrices is the most time consuming part of a DUBLAT aerodynamic solution. To minimize computer time requirements, previously-calculated quasi-inverse matrices may be reused rather than recalculating them under certain circumstances. For example, velocity profile and aerodynamic pressure modifications and structural mode additions and/or alterations may be investigated using previously-calculated matrices. Each of these matrices must be associated with an aerodynamic model that has the same

configuration analyzed with the same symmetry option at the same Mach number and k-value.

Caution: If previously-calculated quasi-inverse matrices are to be used, the associated set of k-values must correspond to the list of k-values specified via KVAL.

Default: Quasi-inverse matrices are generated and used in the analysis but are not saved on DUBLRNF for subsequent use.

216.3 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the aerodynamic data calculated by the DUBLAT Processor.

PRINT OUTPUT (DUBLAT, Plist)

List of optional parameters for Plist:

- CASE=Ca The aerodynamic data case number
(integer) for which generated data are
to be printed.
Default: CASE=1
- COND=Co The aerodynamic condition number
(integer) defined by an EXECUTE DUBLAT
parameter list for which generated data
are to be printed.
Default: COND=1
- MACH=(List) A list of 1 to 20 unique Mach numbers
associated with the selected COND number.
Default: Data generated for all Mach
numbers associated with COND
are printed.
- KVAL=(List) A list of 1 to 20 unique reduced
frequency values associated with the
selected COND number.
Default: Data generated for all k-values
associated with COND are
printed.
- LEVEL=(List) A list of 1 to 5 unique integers denoting
which group(s) of generated data are
to be printed. The LEVEL indicators
are as follows:
- 1-- Aerodynamic model geometry data
 - 2-- Structural mode shapes interpolated
to the aerodynamic control points
 - 3-- Aerodynamic pressure differences
 - 4-- Sectional generalized forces and dynamic
stability derivatives
 - 5-- Generalized airforces
- Default: LEVEL=5

218.0 EXTRACT POSTPROCESSOR

Selected input data and data previously-calculated by another ATLAS processor may be extracted from the primary ATLAS data base by the EXTRACT Postprocessor. The extracted data are saved on the ATLAS file EXTRRNF for subsequent graphical display or printout generation. The data required for all plots generated via the GRAPHICS Postprocessor (sec. 228.0) must be initially extracted by execution of this module. Printout generation of extracted data is currently only functional for nodal displacements and element stresses.

Data values to be extracted for subsequent postprocessing are identified by the data-component labels which are included in a label subset. Label subsets are defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0). Standard label subsets comprised of pre-defined selections of data-component labels are provided for direct reference in the EXECUTE EXTRACT statement. Use of these standard subsets for generation of the data displays described in table 218-1 obviates the need, in most cases, to define label subsets. The data-component labels included in the standard label subsets are identified in section 156.0. The associated data values reside in appropriate matrices in the primary ATLAS data base (see table 218-1).

In addition to the selection of only certain types of input and analysis data by using appropriate label subsets, the extracted data may be associated with user-selected regions of the mathematical model. Regions are defined by node and/or element subsets of the model via the SUBSET-DEFINITION Preprocessor.

Some of the data-components are dependent on load cases, mass-distribution conditions, flutter conditions, etc. Extraction of data values for subsets of cases or conditions is controlled by the parameters specified in the EXECUTE EXTRACT statement described in the following section.

By appropriate selection of label subsets, regions of the mathematical model (node/element subsets), and case/condition subsets, the user may extract any number of input and/or calculated data values for subsequent postprocessing activities. An extracted data block may therefore include anywhere from one data value up to all data values associated with a mathematical model. A maximum of 28 data blocks may be extracted by execution of the EXTRACT Postprocessor.

REPRODUCED FROM BLANK NOT FILMED

Table 218-1. Data Displays Associated with Data Extraction for the ATLAS Standard Label Subsets

STANDARD LABEL-SUBSET NAME	REFERENCE SECTION	DATA DISPLAY TO BE GENERATED (ref. Table 228-1)	REQUIRED INPUT DATA OR PROCESSOR EXECUTION PRIOR TO EXECUTING THE EXTRACT POSTPROCESSOR	ATLAS DATA FILES THAT MUST BE AVAILABLE WHEN THE EXTRACT POSTPROCESSOR IS EXECUTED
BMODE	218.1.6	Plot buckling mode-shape displacements at retained nodes	BUCKLING Processor	DATARNF,STIFRNF,BUCKRNF
DISGRID	218.1.3	Plot displacements on structural grid	STRESS Processor	DATARNF,STRETNF
DISNODE	218.1.3	Plot displacements at nodal locations without structural grid	STRESS Processor	DATARNF,STRETNF
DISPRINT	218.1.12	Print displacements	STRESS Processor	DATARNF,STRETNF,LOADRNF
KGRID	218.1.2	Plot structural model geometry	Nodal and stiffness data	DATARNF
KPROP	218.1.2	Plot stiffness element properties on structural grid	Nodal and stiffness data	DATARNF
LOADAB	218.1.8	Plot fuel/payload loadability diagrams	MASS Processor	DATARNF,MASSRNF
MGRID	218.1.7	Plot mass-element model geometry	Nodal and mass element data	DATARNF
NODES	218.1.1	Plot nodes without grid	Nodal data	DATARNF
SMS	218.1.5	Plot "strength-designed" element-property margins of safety on structural grid	DESIGN Processor	DATARNF,DESIRNF
STRESS	218.1.4	Plot stresses on structural grid	STRESS Processor	DATARNF,STRETNF
STRPRINT	218.1.12	Print stresses	STRESS Processor	DATARNF,STRETNF,LOADRNF
TMS	218.1.5	Plot "thermal-designed" element-property margins of safety on structural grid	DESIGN Processor	DATARNF,DESIRNF
VGVF	218.1.10	Plot velocity-damping (V-g) and velocity-frequency (V-f) graphs	FLUTTER Processor	FLUTRNF
VMODE	218.1.9	Plot vibration mode-shape displacements at retained nodes	VIBRATION Processor	DATARNF,VIBRNF

218.1 EXECUTE EXTRACT STATEMENT

Execution of the EXTRACT Postprocessor is initiated by the following statement. Previous input or calculation of the ATLAS data to be extracted for subsequent graphical or printout displays is required (reference table 218-1).

EXECUTE EXTRACT (Plist)

EXNAME=name An alphanumeric word of 1 to 7 characters that identifies the data block established by the remaining parameters of this statement. A maximum of 28 different names can be specified per job. If the same "name" is specified in multiple EXTRACT executions in a single job, only the last extracted data block for "name" is saved. This identifier is subsequently used as a parameter in an EXECUTE GRAPHICS statement (sec. 228.0) or in a PRINT OUTPUT statement for displacements or stresses (sec. 254.2.3).
Default: Warning. No data are extracted.

The following parameter is required unless the contents of an ATLAS data matrix are to be extracted via the MATRIX parameter (sec. 218.1.11).

LSUB=Label Label subset name defined via the SUBSET-DEFINITION Preprocessor or one of the ATLAS standard label-subset names described in table 218-1. Data values corresponding to each data-component label included in LSUB are extracted and stored on the file EXTRNF for subsequent postprocessing. If "Label" is defined by input to the SUBSET-DEFINITION Preprocessor and if the data display to be generated is the same type as that associated with one of the standard label-subset names (see table 218-1), the data component labels included in "Label" must be at least those labels included in the standard subset.
Default: Warning. No data are extracted unless the MATRIX parameter is used.

The remaining options for Plist are described in the following sections according to the type of data that are to be extracted for subsequent graphical or printout display. These options are:

<u>Section</u>	<u>Data Type</u>
218.1.1	-- Nodal Data for Graphical Display
218.1.2	-- Stiffness Data for Graphical Display
218.1.3	-- Displacement Data for Graphical Display
218.1.4	-- Stress Data for Graphical Display
218.1.5	-- Design Data for Graphical Display
218.1.6	-- Buckling Data for Graphical Display
218.1.7	-- Mass-element Data for Graphical Display
218.1.8	-- Mass Data for Loadability Diagrams
218.1.9	-- Vibration Data for Graphical Display
218.1.10	-- Flutter Data for Graphical Display
218.1.11	-- Matrix Data for Graphical Display
218.1.12	-- Displacement and Stress Data for Printout

The Plist parameters associated with each of the ATLAS standard label-subset names are summarized in table 218-2. These parameters are described in the following sections.

218.1.1 Nodal Data for Graphical Display

Node numbers, coordinates and analysis frame data are extracted by this option. The following parameters are used if the parameter LSUB=NODES is specified.

EXNAME=name LSUB=Label	} Defined in section 218.1
NSET=Se	The number (integer) of the nodal data set for which data are to be extracted. Default: NSET=1
NSUB=Nxxx	Node subset name associated with the nodal data set "Se." Nodal data are extracted only for the nodes included in Nxxx. Default: Warning. No data are extracted.

218.1.2 Stiffness Data for Graphical Display

Stiffness-element section and material properties and the structural-grid nodal data are extracted by this option. The following parameters are used if the parameter LSUB=KGRID or LSUB=KPROP is specified.

EXNAME=name }
LSUB=Label }

KSET=Se The number (integer) of the stiffness data set for which data are to be extracted.
Default: KSET=1

ESUB=Exxx Stiffness-element subset name associated with set "Se." Stiffness data are extracted only for the elements included in Exxx.
Default: No element data are extracted.

NSUB=Nxxx Node subset name associated with set "Se." This subset must include at least those nodes associated with the elements identified by Exxx. Auxiliary nodes must be included in Nxxx if, for example, offset elements are to be plotted accordingly.
Default: Warning. No data are extracted.

BSUB=ONxxx An ordered-node subset name associated with set "Se." The sequence of nodes identified by ONxxx which, when connected in the specified order, establishes the boundary of an element-property contour plot. The generated boundary is a contiguous series of line segments that defines a simply-connected curve. The curve is automatically closed by connecting the last node in ONxxx to the first node in ONxxx. Thus, a particular node number may not be used more than once in defining ONxxx. This subset of nodes should identify those nodes included in Nxxx which, when projected onto the contour surface (plane), define the boundary of the plot.
Default: If property contour plots are generated for the extracted data, a rectangular plot boundary is used. This boundary is defined by the minimum and maximum coordinates of the nodes in Nxxx when projected onto the plot plane.

218.1.3 Displacement Data for Graphical Display

Nodal displacements and optionally, structural-grid data are extracted by this option. The following parameters are used if the parameter LSUB=DISGRID or LSUB=DISNODE is specified.

EXNAME=name LSUB=Label	} Defined in section 218.1
KSET=Se	The number (integer) of the stiffness data set for which data are to be extracted. Default: KSET=1
STAGE=St	A boundary condition (BC) stage number (integer) associated with set "Se." Default: STAGE=1
LC=CATlist	ATLAS list of load case identifiers (integers or alphanumeric words) denoting which load-case dependent data are to be extracted. Default: All load cases associated with the specified stiffness set and BC stage.
NSUB=Nxxx	Node subset name associated with set "Se." Displacements are extracted only for the nodes included in Nxxx. If structural-grid data are to be extracted, this subset must include at least those nodes associated with the elements identified by Exxx. Default: Warning. No data are extracted.
ESUB=Exxx	Stiffness-element subset name associated with set "Se." Structural-grid data are extracted only for the elements included in Exxx. Default: No structural-grid data are extracted.
ESUB=ONxxx	An ordered-node subset name associated with set "Se." The sequence of nodes identified by ONxxx which, when connected in the specified order, defines a grid that is to be used instead of the structural grid when plots are generated. In this case, a particular node number

can be used more than once in defining ONxxx.

Default: An ordered node subset is not extracted.

218.1.4 Stress Data for Graphical Display

Element stress data are extracted by this option. The following parameters are used if the parameter LSUB=STRESS is specified. If BFICK elements are included in the model, those stresses based on the BIGBRICK parameter of the EXECUTE STIFFNESS statement (sec. 252.2) may not be extracted by this module.

EXNAME=name LSUB=Label	} Defined in section 218.1
KSET=Se STAGE=St LC=CATlist	} Same as defined in section 218.1.3
ESUB=Exxx	Stiffness-element subset name associated with set "Se." Stresses are extracted only for the elements included in Exxx. Default: Warning. Element stresses are not extracted.
NSUB=Nxxx	Node subset name associated with set "Se." This subset must include at least those nodes associated with the elements identified by Exxx. Default: Warning. No data are extracted.
BSUB=ONxxx	Same as defined in section 218.1.2 with the exception that ONxxx is used to establish the boundary of an element-stress contour plot. Default: If stress contour plots are generated for the extracted data, a rectangular plot boundary is used. This boundary is defined by the minimum and maximum coordinates of the nodes in Nxxx when projected onto the plot plane.

218.1.5 Design Data for Graphical Display

Structural-grid data and minimum margins of safety resulting from element resizing via the DESIGN Processor are extracted by this option. Only the "most critical" margin of

safety associated with an element is extracted. The following parameters are used if the parameter LSUB=SMS or LSUB=TMS is specified.

EXNAME=name } LSUB=Label }	Defined in section 218.1
KSET=Se	The number (integer) of the stiffness data set for which data are to be extracted. Default: KSET=1
STAGE=St	A boundary condition (BC) stage number (integer) associated with set "Se." Default: STAGE=1
CYCLE=CATlist	ATLAS list of design cycles (integers) associated with the specified stiffness set and BC stage for which minimum margin of safety data are to be extracted. Default: Warning. Design data are not extracted.
ESUB=Exxx	Stiffness-element subset name associated with set "Se." Margin of safety data are extracted only for the elements included in Exxx. Default: Warning. No design data are extracted.
NSUB=Nxxx	Node subset name associated with set "Se." This subset must include at least those nodes associated with the elements identified by Exxx. Default: Warning. No design data are extracted.
BSUB=ONxxx	Same as defined in section 218.1.2 with the exception that ONxxx is used to establish the boundary of an element margin-of-safety contour plot. Default: If margin-of-safety contour plots are generated for the extracted data, a rectangular plot boundary is used. This boundary is defined by the minimum and maximum coordinates of the nodes in Nxxx when projected onto the plot plane.

218.1.6 Buckling Data for Graphical Display

The "critical" load eigenvalues and buckling mode shapes calculated by execution of the BUCKLING Processor are extracted by this option. The following parameters are used if the parameter LSUB=EMODE is specified.

EXNAME=name LSUB=Label }	Defined in section 218.1
BSET=Se	The number (integer) of the buckling data set for which data are to be extracted. This set number must have been defined via the EXECUTE STIFFNESS statement. Default: Warning. Buckling data are not extracted.
MODE=CATlist	ATLAS list of mode shape numbers (integers) for which data are to be extracted. Each entry in CATlist identifies the i-th sequential, mode-shape number associated with the eigensolution. Default: Warning. Buckling data are not extracted.
BSUB=ONxxx	An ordered-node subset name associated with set "Se." The sequence of nodes identified by ONxxx which, when connected in the specified order, defines a grid to be displayed when mode-shape plots are generated. A particular node number can be used more than once in defining ONxxx. Default: No grid data are extracted. Caution: If a grid is to be displayed on mode-shape plots, this parameter must be used.

218.1.7 Mass-Element Data for Graphical Display

Mass-element geometries and associated nodal data required for mass-model grid plots are extracted by this option. The following parameters are applicable if the parameter LSUB=MGRID is specified.

EXNAME=name LSUB=Label }	Defined in section 218.1
-----------------------------	--------------------------

MSET=Se The number (integer) of the mass data set for which data are to be extracted. Default: MSET=1

ESUB=Exxx Mass-element subset name associated with set "Se." Mass element data are extracted only for the elements included in Exxx. Default: No element data are extracted.

NSUB=Nxxx Node subset name associated with set "Se." This subset must include at least those nodes associated with the elements identified by Exxx. Auxiliary nodes must be included in Nxxx if, for example, offset elements are to be plotted accordingly. Default: Warning. No data are extracted.

218.1.8 Mass Data for Loadability Diagrams

The passenger, cargo and fuel condition data, as calculated by execution of the MASS Processor, are extracted by this option for generation of loadability-diagram plots. The following parameters are applicable if the parameter LSUB=LOADAB is specified.

EXNAME=name } LSUB=Label }	Defined in section 218.1
MSET=Se	The number (integer) of the mass data set for which data are to be extracted. Default: MSET=1
PCOND=Pass } CCOND=Car } FCOND=Fuel }	Passenger-payload (PCOND), cargo-payload (CCOND) and fuel (FCOND) condition numbers (integers) associated with set "Se." Each of these parameters may be specified twice to identify multiple payload and fuel conditions for a loadability diagram. At least one of these parameters must be specified if the extracted data are to be used for generation of a loadability plot. Default: Payload and Fuel condition data are not extracted.

218.1.9 Vibration Data for Graphical Display

The natural frequencies and vibration mode shapes calculated by execution of the VIBRATION Processor are extracted by this option. The following parameters are used if the parameter LSUB=VMODE is specified.

EXNAME=name LSUB=Label	} }	Defined in section 218.1
VSET=Se		The number (integer) of the vibration data set for which modes and frequencies are to be extracted. Default: Warning. Vibration data are not extracted.
NSUB=Nxxx		Node subset name associated with the nodal/stiffness data set that corresponds to the eigenproblem identified by VSET=Se. A subset mode-matrix must have been previously-generated via the VIBRATION Processor for the subset of retained nodes identified by Nxxx. Default: All retained degrees-of-freedom associated with the eigenproblem are represented in the extracted mode-shape data.
MODE=CATlist		ATLAS list of mode shape numbers (integers) for which data are to be extracted. Each entry in CATlist identifies the i-th sequential, mode-shape number associated with the eigensolution. Default: Warning. Vibration data are not extracted.
ESUB=ONxxx		An ordered-node subset name associated with set "Se." The sequence of nodes identified by ONxxx which, when connected in the specified order, defines a grid to be displayed when mode-shape plots are generated. A particular node number can be used more than once in defining ONxxx. If the NSUB parameter is used, the nodes identified by ONxxx must be included in Nxxx. Otherwise, any node associated with set "Se" can be used. Default: No grid data are extracted.

Caution: If a grid is to be displayed on mode-shape plots, this parameter must be used.

218.1.10 Flutter Data for Graphical Display

Flutter solution data which are calculated by the FLUTTER Processor and which are required for velocity-damping (V-g) and velocity-frequency (V-f) graphs for selected altitudes are extracted by this option. The following parameters are used if the parameter LSUB=VGVF is specified.

EXNAME=name LSUB=Label	} Defined in section 218.1
CASE=CATlist	ATLAS list of flutter data-case numbers (integers) for which flutter solution data are to be extracted. Default: All flutter CASE data.
CCND=CATlist	ATLAS list of flutter condition numbers (integers) specified via the EXECUTE FLUTTER statement for the identified CASES. Default: All flutter CONDITION numbers.
CSET=CATlist	ATLAS list of flutter data change-set numbers (integers) associated with the specified CASES. Default: All change-set data associated with the specified CASE numbers.
RSET=CATlist	ATLAS list of flutter data retention-vector set numbers (integers) associated with the specified CASES. Default: All retention-set data associated with the specified CASE numbers.
ALT=CATlist	ATLAS list of altitudes associated with the specified CASES. Default: All altitudes associated with the specified CASE numbers. V-g and V-f graphs are generated for each altitude.

218.1.11 Matrix Data for Graphical Display

The contents of an ATLAS data matrix can be extracted for subsequent graphical display. In this case, only the EXNAME parameter and the following MATRIX parameter should be included in Plist.

EXNAME=name Defined in section 218.1

MATRIX={ (Name<,Filename>)
 (Name<,Part>)

This parameter identifies the "Name" of the matrix to be extracted from the ATLAS data file identified by "Filename." "Name" and "Filename" are alphanumeric words. If "Name" identifies a User Matrix (see sec. 200.2), "Filename" need not be specified. The partition number of a User Matrix to be extracted is identified by the integer "Part." The default for "Part" is 1. All standard ATLAS matrices are described in detail in reference 1-2.

Default: Warning. Matrix data are not extracted.

218.1.12 Displacement and Stress Data for Printout

Displacements and element stresses are extracted via this option for subsequent printout by the STRESS Postprocessor (sec. 254.2.3). If BRICK elements are included in the model, those stresses based on the BIGBRICK parameter of the EXECUTE STIFFNESS statement (sec. 252.2) may not be extracted by this module. The following parameters are applicable if the parameter LSUB=DISPRINT or LSUB=STRPRINT is specified.

EXNAME=name } Defined in section 218.1
LSUB=Label }

KSET=Se The number (integer) of the stiffness data set for which data are to be extracted.
Default: KSET=1

STAGE=St A boundary condition (BC) or superposition stage number (integer) associated with set "Se."
Default: STAGE=1

LC=CATlist ATLAS list of load case identifiers
 (integers or alphanumeric words) denoting
 which load-case dependent data are to
 be extracted.
 Default: All load cases associated with
 the specified stiffness set
 and BC stage.

ESUB=Exxx Stiffness-element subset name associated
 with set "Se." Stresses are extracted
 only for the elements included in Exxx.
 If only displacement data are to be
 extracted, this parameter should not
 be used.
 Default: Element stresses are not
 extracted.

NSUB=Nxxx Node subset name associated with set
 "Se." Displacements are extracted only
 for the nodes included in Nxxx. If
 stress data are to be extracted, this
 subset must include at least those nodes
 associated with the elements identified
 by Exxx.
 Default: Warning. No data are extracted.

Table 218-2. Execution Parameters Associated with Extraction of Data Values for the ATLAS Standard Label Subsets

EXECUTE EXTRACT PLIST PARAMETERS	ATLAS STANDARD LABEL-SUBSET NAME SPECIFIED BY THE EXECUTE PARAMETER "LSUB = Name"														
	BMODE	DISGRID	DISNODE	DISPRINT	KGRID	KPROP	LOADAB	MGRID	MODES	SMS	STRESS	STRPRINT	TMS	YGVF	YMODE
ALT = CATlist														✓	
BSET = Se	✓														
BSUB = ONxxx	✓	✓			✓				✓	✓		✓			✓
CASE = CATlist														✓	
CCOND = Car						✓									
COND = CATlist														✓	
CSET = CATlist														✓	
CYCLE = CATlist									✓			✓			
ESUB = Exxx		✓		✓	✓		✓		✓	✓	✓	✓			
EXNAME = name	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
FCOND = Fuel						✓									
KSET = Se		✓	✓	✓	✓	✓			✓	✓	✓	✓			
LC = CATlist		✓	✓	✓						✓	✓				
MODE = CATlist	✓														✓
MSET = Se						✓	✓								
NSET = Se								✓							
NSUB = Nxxx		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			✓
PCOND = Pass						✓									
RSET = CATlist													✓		
STAGE = St		✓	✓	✓					✓	✓	✓	✓			
VSET = Se															✓

220.0 FLEXAIR AERODYNAMICS MODULES

The FLEXAIR Processor calculates generalized airforce matrices that include the effects of the structural flexibility associated with those vibration modes that are not included in the set of generalized coordinates used to analyze a structure (ref. 220-1). These airforces are suitable for subsequent flutter analysis via the ADDINT and FLUTTER Modules (secs. 202.0 and 222.0). The calculated airforce matrices are valid for only one altitude and frequency of oscillation and thus differ from the generalized airforce matrices generated by the other aerodynamics processors.

FLEXAIR requires the prior calculation of a singular or nonsingular structural stiffness matrix, K , a positive definite mass matrix, M , a subset of the eigenvectors, G , of $(K-E*M)*G=0$, where E represents the eigenvalue corresponding with the eigenvector G , and the generalized stiffness matrix G^T*K*G . FLEXAIR also requires that Aerodynamic Influence Coefficient (AIC) matrices be generated previously by execution of the INTERPOLATION Processor (sec. 232.0) and one of the other Aerodynamics Processors (ADDINT, AF1, DUBLAT, MACHBOX and RHO3). The kinematic freedoms represented in the AIC matrices must be the same freedoms or a subset of the freedoms used to obtain the vibration modes G .

Printout of the generalized airforces calculated by the FLEXAIR Processor (sec. 220.1) may be requested from the FLEXAIR Postprocessor (sec. 220.2).

PROCESSED FROM DATA NOT FILMED

220.1 EXECUTE FLEXAIR STATEMENT

The following statement initiates execution of the FLEXAIR Processor. Previous execution of the VIBRATION Processor, (sec. 258.0), the INTERPOLATION Processor (sec. 232.0) and one or more of the Aerodynamics Processors (ADDINT, AF1, DUBLAT, MACHBOX and RHO3) are required.

EXECUTE FLEXAIR (Plist)

List of optional parameters for Plist:

ID = xxxx	Alphanumeric word with 1 to 4 characters to be used in naming the set of Q-matrices (generalized air force matrices) generated by the FLEXAIR Processor. Each Q-matrix name is comprised of the 7 characters xxxl _{nn} , where "l" and "nn" are automatically assigned as the corresponding sequential numbers of the altitude specified by the ALT parameter and the k-value associated with the generated set. Default: Error. Execution is terminated
-----------	---

A set of AIC matrices input to the FLEXAIR Processor is identified either via the ADDINT parameter or via a triplet of parameters, Aname, CASE and COND as presented below. The ADDINT parameter identifies a set of AIC matrices generated by the ADDINT Processor, whereas each triplet identifies a set of AIC matrices calculated directly by the AF1, DUBLAT, MACHBOX or RHO3 Processor.

The following three parameters are specified in triplets.

Aname	Name of the unsteady aerodynamics processor which was previously executed to calculate a set of AIC matrices. The options for "Aname" are AF1, DUBLAT, MACHBOX and RHO3. Default: Error. Execution is terminated unless the keyword ADDINT is input.
CASE-Ca	Aerodynamics data case number (integer) associated with the specified COND number for which AIC

- matrices were calculated by an "EXECUTE Aname" statement.
Default: CASE=1
- COND=Co Aerodynamic condition number (integer) specified by the same "EXECUTE Aname" statement.
Default: COND=1
- ADDINT=Xxxxx Alphanumeric word assigned by the ID parameter of a previous EXECUTE ADDINT statement.
Default: Error. Execution is terminated unless a valid "Aname" is input.
- MACH=Mach The Mach number greater than or equal to zero that is associated with the input AIC matrices which are to be processed. This parameter identifies which of the sets of AIC matrices identified by the CASE and COND parameters is to be used. This is required because a single execution of the DUBLAT, MACHBOX or RHO3 Processor may generate multiple sets of AIC matrices associated with more than one Mach number.
- The specified value of Mach must be within one tenth of one percent of the Mach number associated with each input set of AIC matrices unless they were generated by the AF1 Processor, in which case, the AIC matrices are defined uniquely by the CASE and COND parameters.
- Default: The first Mach number associated with the first set of AIC matrices specified via Plist.
- VSET=VS The vibration set number (integer) identifying the structural model being analyzed (sec. 258.1).
Default: VSET=1
- NKVAL=n The number of reduced frequencies (integer >0) associated with the

AIC matrices for which FLEXAIR Q-matrices are to be generated. The AIC matrices are selected in the order that they were generated by the specified Aerodynamics Processor. If "n" is greater than the number of matrices available, the following default action is taken.

Default: Q-matrices corresponding to all the AIC matrices identified by Plist are generated.

ALT=(list)

List of 1 to 8 altitudes in the range -200,000 through 300,000 feet, in any sequence, for which Q-matrices are to be generated.

Default: ALT=0.0

FLUTFREQ=f

The frequency of oscillation in Hz that is to be used to generate the Q-matrices. If the Q-matrices are to be used in a flutter analysis via the FLUTTER Processor, "f" should be an estimate of the flutter frequency.

Default: "f" is calculated as $[k \cdot V \cdot (2 \cdot \pi \cdot b)]$ where "k" and "b" are the previously-defined reduced frequency and reference length and V is derived from the MACH and ALT parameters.

If the stiffness matrix corresponding to V_{SL} is singular, it is necessary to select freedoms associated with the zero-frequency modes of the "Mode" matrix generated by the VIBRATION Processor such that a non-singular partition thereof is identified. This is done via the following parameter.

FREEDOMS=(List)

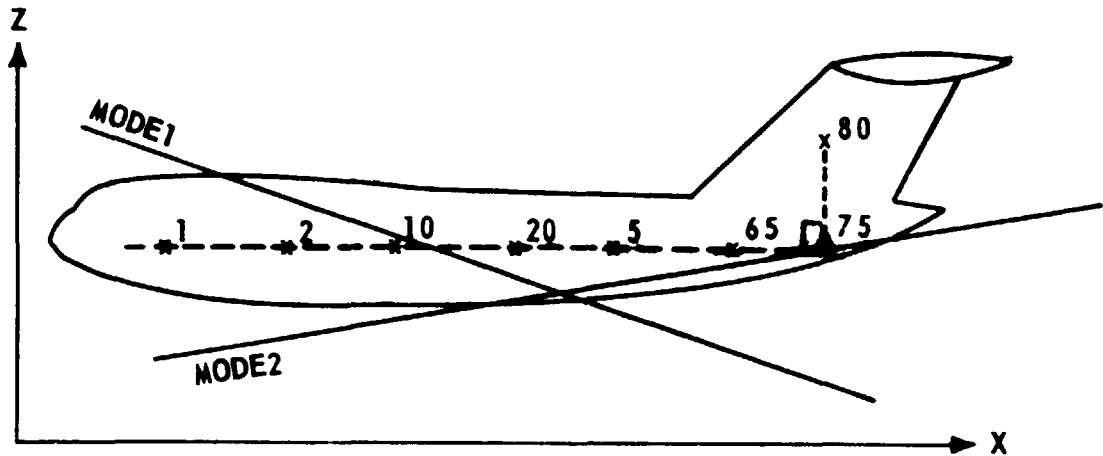
A list of alphanumeric words with 3 to 7 characters of the form "AA.n." Each word identifies a row in the "Mode" matrix: "n" is a user node number: "AA" is one of the six character pairs TX, TY, TZ, RX, RY, RZ identifying the kinematic freedom associated with node "n." The number of words in list must be equal to

the number of zero-frequency modal vectors associated with VSET.
 Default: Error, unless there are no zero-frequency vibration modes.

Example: The figure below shows a typical node distribution along the centerline of an airplane. Suppose that a set of symmetric mode shapes is associated with this configuration and that the set includes two rigid body modes, each of which is a combination of pitch and vertical translation. Non-singular partitions may be identified by one of the lists: (TZ1, TZ2), (TZ1, RY5), (TX75, TX80), etc. The following combinations, however, identify singular partitions: (TX1, TX2), (RY10, RY5), (TZ75, TZ80), etc.

The information in the mode-shape matrix is shown below. For this case, the non-singular partition formed by the parameter FREEDOMS=(TZ1, RY5) is

$$\begin{bmatrix} tz_{11} & tz_{12} \\ ry_{51} & ry_{52} \end{bmatrix}$$



Node-Shape Matrix Information

MODE NUMBER		1	2	3	4	.	.	.
FREQUENCY		0.0	0.0	f3	f4	.	.	.
NODE	FREEDOM							
1	TX	tx ₁₁	tx ₁₂	tx ₁₃	tx ₁₄	.	.	.
1	TZ	tz ₁₁	tz ₁₂	tz ₁₃	tz ₁₄	.	.	.
1	RY	ry ₁₁	ry ₁₂	ry ₁₃	ry ₁₄	.	.	.
.
.
.
5	RY	ry ₅₁	ry ₅₂	ry ₅₃	ry ₅₄	.	.	.
.
.
.

If the available AIC matrices do not correspond in size and arrangement with the previously-calculated vibration mode-shape matrix, the following parameter must be used. For example, the vibration modes may include displacements and rotations at nodes on the wing, body and tail, whereas the AICs are available for displacements on the wing alone. This parameter should not be input if the order and number of freedoms represented by the AIC matrices and the mode-shape matrix are compatible.

SUBSET=Nxxx

The name of the node subset corresponding with the AIC matrices. This subset must have been defined by the SUBSET-DEFINITION Preprocessor (sec. 156.0) and it should be the same subset used in execution of the INTERPOLATION Processor in conjunction with the AIC option (sec. 232.0). A subset mode-matrix must also have been generated for this subset by the VIBRATION Processor (sec. 258.0).

Default: Execution is terminated if the order and number of kinematic freedoms associated with VSET differ from that associated with the AIC matrices.

220.2 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the aerodynamic data calculated by the FLEXAIR Processor.

PRINT OUTPUT (FLEXAIR, Plist)

List of optional parameters for Plist:

- | | |
|--------------|---|
| ID=Xxxx | Alphanumeric word assigned by the ID parameter of a previous EXECUTE FLEXAIR statement. This word identifies the set of Q-matrices (generalized airforce matrices) to be printed.
Default: Error. Print is terminated. |
| AIT= (List) | List of 1 to 8 altitudes identifying which Q-matrices are to be printed.
Default: Q-matrices are printed for all available altitudes. |
| KVAL= (List) | A list of reduced-frequency values identifying which of the "Xxxx" Q-matrices are to be printed.
Default: All matrices included in the "Xxxx" set are printed. |

222.0 FLUTTER MODULES

The FLUTTER Processor calculates eigenvalues and eigenvectors of the linear, complex eigen-problem associated with flutter analyses via the automated V-g solution technique published in reference 122-1. The solution procedure involves solving the eigenvalue problem many times, using the reduced frequency of oscillation (k-value) as a parameter, by the standard Laguerre iterative technique. Continuity of the calculated eigenvalues with respect to k-value is maintained by solving the eigenproblem at a relatively large number of k-values. Sign changes in the imaginary part of the eigenvalue are detected as the k-value is changed so that neutrally stable points are determined automatically via linear interpolation. "Flutter crossings" may be determined at user-selected damping levels near the neutral stability points.

A matched point solution may also be requested whereby a matched flutter point for the critical mode is determined by automatically selecting a different altitude and repeating the complete V-g solution. A matched flutter point occurs when the calculated flutter speed equals the initially-specified Mach number multiplied by the speed of sound at the corresponding altitude.

The equations of motion for flutter may be expressed as

$$\sum_{j=1,n} [\omega^2 * M_{ij} - (1+ig)K_{ij} + 1/2\rho v^2 * Q_{ij}] q_j = 0$$

where n is the number of structural modes used in the analysis, ω is the circular frequency of oscillation, M_{ij} is an element of the real mass matrix, "i" is the value $(-1)^{1/2}$, "g" is the incremental structural damping coefficient, K_{ij} is an element of the complex stiffness matrix including stiffness and structural damping effects, Q_{ij} is an element of the complex airforce matrix, ρ is the free-stream density, v is the free-stream velocity and q_j is the j-th component of the eigenvector corresponding to ω and g.

Most commonly the mass and stiffness matrices will be the generalized mass and stiffness matrices generated via the VIBRATION Processor (sec. 258.0). In this case, the generalized airforce matrices are calculated via the ADDINT Processor (sec. 202.0) based on the same vibration modes used to generate the mass and stiffness matrices. Alternate approaches may be used to generate the necessary ATLAS User Matrices to define the flutter equation. These matrices may be based either on generalized coordinates or on structural kinematic freedoms. For the latter case, reduced stiffness and mass matrices may be generated by

PROCESSED FROM BLANK NOT FILMED

using the REDUCE cataloged procedure (see appendix E). Corresponding airforce matrices may be generated via the AIC option provided by the INTERPOLATION Processor, one of the aerodynamics Processors, and the ADDINT Processor. For all cases, a diagonal, structural damping matrix may be input via the flutter data (sec. 122.0) to define the complex component of the stiffness matrix.

As discussed in reference 122-1, the intervals of the k-value sequence must be small enough so that continuity of the calculated eigenvalues as a function of k-value is maintained. Proper selection of a k-value sequence generally allows a valid eigensolution to be effected via a single Laguerre iteration at each step (each k-value). Lack of convergence in a solution may be detected by discontinuities in the velocity-damping (V-g) and velocity-frequency (V-f) curves which may be plotted via the EXTRACT and GRAPHICS Postprocessors (sec. 218.0 and 228.0). These curves are plotted as continuous curves if no ill-conditioning is inherent to the solution.

If the airforces calculated by the ADDINT Processor are independent of the free stream density, the density is defined in one of the following ways:

- a) One or more altitudes at which solutions are to be performed may be specified by the FLUTTER input data (sec. 122.0). The corresponding standard air densities are generated automatically by the FLUTTER Processor (sec. 222.2).
- b) The density may be specified directly by the DENSITY parameter of the EXECUTE FLUTTER statement.
- c) A "matched point" solution may be requested by the MPS parameter of the EXECUTE FLUTTER statement. A single initial altitude may be specified, however, the altitudes for subsequent analyses are generated automatically. The standard air densities corresponding to the required altitudes are generated automatically by the FLUTTER Processor.

If the airforces calculated by the ADDINT Processor are dependent on the free-stream density, a "matched point" solution may not be performed; all specified parameters relating altitude and density information are ignored.

Printout of the FLUTTER input data and flutter solution data calculated by the FLUTTER Processor may be requested from the FLUTTER Postprocessor as described in sections 222.1 and 222.3, respectively.

222.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the flutter input data associated with selected FLUTTER data cases (sec. 122.0).

PRINT INPUT (FLUTTER, Plist)

List of optional parameters for Plist:

CASE=CATlist ATLAS list of FLUTTER data case numbers (integers) for which input data are to be printed.
Default: All previously-defined FLUTTER data sets.

CSET=CATlist ATLAS list of FLUTTER data change-set numbers (integers) that are associated with each of the specified CASE numbers. Data are printed only for the change-sets identified by this parameter.
Default: All change-sets defined for each FLUTTER CASE identified by this statement.

222.2 EXECUTE FLUTTER STATEMENT

The following statement initiates execution of the FLUTTER Processor. Previous calculation of stiffness and mass matrices is required. Additionally, previous execution of the ADDINT Processor is required for generation of the airforces essential to the flutter equation to be solved.

EXECUTE FLUTTER (Plist)

List of optional parameters for Plist:

COND=co A condition number (integer) in the range 1 to 36 which identifies the remaining parameters of this statement. If multiple executions are required in a single job, the condition number specified by each statement of this type must be unique.
Default: COND=1

CASE=CATlist ATLAS list of FLUTTER data case numbers (integers) associated with the flutter problem(s) to be solved. Non-existent data case numbers identified by CATlist are not allowed.
Default: CASE=1

Note: If flutter input data are not required to define the flutter problem (sec. 122.0), CASE should be set to 1.

CSET=CATlist ATLAS list of FLUTTER data change-set numbers (integers) that are associated with each of the specified CASE numbers.
Default: All change-sets defined for each FLUTTER CASE identified by this statement.

VSET=Num The vibration set number (integer) identifying the structural model being analyzed (sec. 258.1).
Default: VSET=1

⟨STIFID=Name1⟩ Alphanumeric names of the stiffness
⟨MASSID=Name2⟩ and/or mass User Matrices to be used in defining the flutter equation. One or both of these parameters may be used.

These matrices may be based either on generalized coordinates or on structural, kinematic freedoms. If both of these parameters are used, they identify implicitly the structural model being analyzed. In this case, the parameter VSET is ignored.

Default: The generalized mass and/or stiffness matrices generated by the VIBRATION Processor for the specified VSET number are used.

< BRITISH
METRIC >

Key-word identifying which system of units is being used to form the equations of motion. A compatible system of units must be used to define the altitude and the stiffness, mass and airforce matrices.

Default: BRITISH

GAFID=Zzzzz

Alphanumeric name assigned by the ID parameter of a previous EXECUTE ADDINT statement. This name identifies a set of airforce matrices associated with a user-selected sequence of strictly-decreasing reduced frequencies.

Default: GAFID=GAF

NMODES=n

The number of freedoms (integer > 0) to be included as coordinates of the flutter equation to be solved. This parameter denotes that the stiffness, mass, airforce and damping matrices are to be truncated prior to solution. The right-most columns and corresponding rows of the flutter matrix are eliminated first. If "n" exceeds the number of freedoms inherent to the input matrices, this parameter is ignored.

Default: The number of freedoms included in the flutter equation is the same as the number of freedoms associated with the input matrices.

NRF=n

The number of reduced frequencies (integer > 0) associated with the airforces for which flutter solutions are to be performed. Flutter solutions

are effected using each of the first "n" airforce matrices associated with the set identified by the parameter GAFID. If "n" is greater than the number of matrices available, the following default action is assumed. Default: Flutter solutions are effected at each of the reduced frequencies associated with the specified set of airforce matrices.

GCROSS=(q1,q2,q3) This parameter identifies 1 to 3 structural-damping values for which V-g "crossings" are to be calculated. The effects of structural damping that are introduced by use of this parameter complement any damping factors defined by the FLUTTER data (sec. 122.0). Default: GCROSS=0.0

{ EVAL } < =FLUTTER
{ EVEC } < =CATlist
{ AVEC } < =(FLUTTER,CATlist) >

This parameter identifies the reduced frequencies at which eigenvalues, eigenvectors and/or adjoint eigenvectors of the flutter equation are to be calculated and saved. The transposed flutter matrix is used to calculate the adjoint vectors. The type of data to be calculated is denoted by:

EVAL--Eigenvalues
EVEC--Eigenvalues and eigenvectors
AVEC--Eigenvalues and adjoint eigenvectors

Each of these key-words may be specified once per EXECUTE FLUTTER statement.

The data are calculated at the reduced frequencies identified by:

FLUTTER--Key-word that denotes all flutter crossings encountered for the first damping value specified by the GCROSS parameter.

CATlist--ATLAS list of reduced-frequency indices (integers) associated with the matrices identified by the GAFID parameter. The specified indices must conform with the NRF parameter.

Default: If only a key-word, EVAL, EVEC or AVEC, is specified, the data are calculated at FLUTTER. Otherwise, the data are not calculated.

Examples:

EVEC,EVAL=(FLUTTER, 1 to 81 BY 10),AVEC=FLUTTER

For this case, eigenvalues are generated for the flutter crossings and for the first, eleventh, ..., eighty-first reduced frequencies. Additionally, eigenvectors and adjoint vectors are generated at FLUTTER.

EVEC=13

For this case, eigenvalues and eigenvectors are generated only for the 13th reduced frequency.

VMIN=vl
VMAX=vu
FMIN=f1
FMAX=fu

These parameters define the envelope of free-stream velocities (V) and reduced frequencies (F) to be used when performing the flutter solution ($V_{MIN} \leq V_{MAX}$; $F_{MIN} \leq F_{MAX}$). All flutter solutions that exist outside of the envelope are discarded. The flutter search begins at the reduced frequency $k_{-max} = 2 \pi b (F_{MAX}/V_{MIN})$ and stops at $k_{-min} = 2 \pi b (F_{MIN}/V_{MAX})$ where "b" is a reference length. The values of "k-max" and "k-min" must be within the range of frequencies identified by the parameter NRF. Defaults: VMIN=1.E-100
VMAX=1.E+100
FMIN=0.0 Hz
FMAX=1000.0 Hz

STILL

Key-word denoting that the complex still-air modes associated with a free-stream velocity of zero are to be calculated. resulting solution

is used as the initial estimate of the Laguerre iterative solution of the flutter equation.

Default: No still-air mode solution is effected. The initial, trial eigensolution is computed from the uncoupled flutter equation by using only the diagonal elements of the stiffness matrix, mass matrix and the first airforce matrix associated with the set identified by the parameter GAFID.

ITER=n

The number of Laguerre iterations (integer > 0) to be performed when calculating the eigensolution of the flutter equation at each reduced frequency.

There is no single convergence criterion available. Therefore, the number of k-values associated with the set of airforce matrices identified by GAFID and the number of Laguerre iterations to be performed must be selected carefully. In general, a relatively large number of k-values ensures that each eigensolution is obtained with sufficient accuracy after only one iteration.

Default: ITER=1

DENSITY=rho

The free-stream density greater than or equal to zero used to simulate flutter of a model in fluids other than air. The units of "rho" are (slugs/cubic foot) in the BRITISH system and (kilograms/cubic meter) in the METRIC system. This parameter may not be used if a "matched point" solution is requested via the MPS parameter or if the airforces are density-dependent. Only one DENSITY may be associated with the specified data CASE(s). Altitudes specified for the selected aerodynamic cases via the input data (sec. 122.0) are ignored if this parameter is input.

Default: The standard atmospheric air densities associated with the altitudes are calculated automatically.

MPS<=alt>

This parameter denotes that a "matched point" flutter solution is to be effected. The altitude, in feet or meters, at which the search is to begin may be specified by "alt." If "alt" is defaulted or specified as zero, the eigensolution/flutter-speed matching solution begins at sea level. This parameter may not be used if the airforces are dependent on free-stream density.

Default: A V-q solution is effected.

NALT=n

The number of altitudes (integer) at which "matched point" eigensolutions are to be effected ($1 \leq n \leq 40$). This parameter is valid only when used in conjunction with the MPS parameter. If NALT=1, a single, V-q solution is performed.

Default: NALT=5

NO PLOT

Key-word denoting that flutter solution data required for plotting V-q and V-f graphs (sec. 218.0 and 228.0) are not to be generated.

Default: Flutter solution data required for plotting V-q and V-f graphs are generated and saved on FLUTRNF.

222.3 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the flutter solution data calculated by the FLUTTER Processor for selected data cases and aerodynamic conditions.

PRINT OUTPUT (FLUTTER, Plist)

List of optional parameters for Plist:

CASE=CATlist	ATLAS list of FLUTTER data case numbers (integers) for which flutter solution data are to be printed. Default: CASE=1
COND=Co	The flutter condition number (integer) specified by an EXECUTE FLUTTER parameter list for which data are to be printed. Default: COND=1
<div style="border-left: 1px solid black; border-right: 1px solid black; padding: 0 5px; display: inline-block;">EVAL EVEC AVEC</div>	Key-words which denote that the eigenvalues (EVAL), eigenvectors (EVEC) or adjoint eigenvectors (AVEC) are to be printed. Several of these key-words may be used in a single statement. Default: No eigenvalues or eigenvectors are printed.

224.0 FREEBODY POSTPROCESSOR

Printout of the internal nodal forces acting on selected stiffness elements may be requested from the FREEBODY Postprocessor as described in the following section.

A region (free-body) of the complete structural model is isolated by defining an element subset via the SUBSET-DEFINITION Preprocessor (sec. 156.0). The collection of elements contained in this subset need not define a contiguous grid. The internal forces which are associated with the elements of this subset and which are acting at selected nodes associated with this subset are displayed. Selected nodes are identified by a node subset defined via the SUBSET-DEFINITION Preprocessor.

The printed information is displayed in blocks ordered according to increasing node numbers of the selected nodes. Within a block, the internal forces acting on the isolated elements at a particular node are printed. These forces are measured relative to the corresponding node analysis reference frames. The sums of the force components associated with each specified node are also printed. Printout of the external forces acting on a model may be requested from the LOADS and REACTION Postprocessors (sec. 234.0 and 248.0, respectively).

PREVIOUS PAGE BLANK NOT FILMED

224.1 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the internal, stiffness-element forces acting at selected nodes for selected load cases. Execution of this postprocessor is allowed only after calculation of element nodal forces via the STRESS Processor (sec. 254.1). A free-body (element subset of isolated stiffness elements) must have been defined via the SUBSET-DEFINITION Preprocessor prior to use of this statement. "Free-body" data associated with a lowest-level substructure may be displayed by specifying the corresponding stiffness data-set number and boundary-condition stage number.

PRINT OUTPUT (FREEBODY, Plist)

List of optional parameters for Plist:

- SET=Se The number (integer) of the stiffness data set for which data are to be printed.
Default: SET=1
- STAGE=St A boundary condition (BC) or superposition stage number (integer) associated with set "Se."
Default: STAGE=1
- Exxx<=Nxxx> Element and node subset names defined via the SUBSET-DEFINITION Preprocessor. Internal, element nodal forces which are acting on the elements contained in Exxx and which are acting at the nodes contained in Nxxx are displayed. Nxxx must contain at least one Structural node that is associated with Exxx.
- Default: If Nxxx is not specified, the internal forces acting at all Structural nodes associated with Exxx are printed.

Several parameters of this type may be used in a single statement.

Default: Warning. No printout is generated.

LC=CATlist ATLAS list of load case identifiers
(integers or alphanumeric words) to be
processed.
Default: All load cases for which element
nodal forces have been previously
calculated by the STRESS Processor
for the specified SET/STAGE.

LINES=lines Maximum number of lines (integer ≤ 56)
including page headings printed on a page.
Default: LINES=56

228.0 GRAPHICS POSTPROCESSOR

The GRAPHICS Postprocessor produces online graphical displays on the Tektronix Model 4014 graphics console in addition to offline plots on the Stromberg Carlson SC4020, the CALCOMP Model 763, and the GERBER drafting machine. Execution of the GRAPHICS Postprocessor can be performed in the interactive mode and/or in the batch mode. The user is provided with a passive interactive graphics capability which enables plots to be viewed between job steps for verification of the technical analysis process.

The conversational dialogue used during interactive-graphics processing is conducted via menus. These menus allow the user to select which plots are to be displayed on the scope of the console, generate different displays of selected data, and enlarge selected areas of online plots by zooming. Selected hardcopy plots can be generated online via a hardcopy unit that is compatible with the Tektronix 4014. Additionally, selected plots viewed on the screen can be directed to one of the offline plot devices (SC4020, CALCOMP or GERBER) to obtain a higher-quality hardcopy print. These online graphics facilities complement the capabilities provided by the GRAPHICS Postprocessor for creation of plots. Execution of the GRAPHICS Postprocessor is described in section 228.4, whereas the interactive-graphics directives are presented in section 228.5.

The data to be displayed graphically must be extracted from the primary ATLAS data base via previous execution of the EXTRACT Postprocessor (sec. 218.0). Each extracted data block is identified by a user-assigned label (the EXNAME parameter as described in section 218.1). Multiple plots can be generated from the block of data associated with an EXNAME. In most cases, the EXNAME data values are identified by use of the standard label-subset (LSUB) names as described in sections 156.0 and 218.0. A summary of the types of plots that can be generated from the technical data extracted by use of the LSUB names is shown in table 228-1. Descriptions and examples of the various display options are presented in section 228.1.

PRECEDING PAGE BLANK NOT FILMED

Table 228-1. Technical-Data Plot Types

TECHNICAL DATA DISPLAY	PLOT TYPE						APPLICABLE STANDARD LABEL-SUBSET (LSUB) NAME(S) (ref.sec. 218.0)
	UNDEFORMED GEOMETRY	DEFORMED GEOMETRY	SCALAR-GRID	CONTOUR	GRAPH	MATRIX	
STRUCTURAL AND/OR MASS MODEL							
• Nodes	/						NODES, KPROP, KGRID, MGRID, DISNODE, DISGRID or STRESS
• Stiffness finite-element grid	/						KGRID, KPROP or STRESS
• Stiffness element properties			/	/			KPROP
• Mass finite-element grid	/						MGRID
• Explored node/grid subsets	/						NODES, KPROP, KGRID, MGRID, DISNODE, DISGRID or STRESS
DISPLACEMENTS/STRESSES							
• Nodal displacements		/					DISNODE or DISGRID
• Element stresses			/	/	/		STRESS
STRUCTURAL RESIZE RESULTS							
• Stiffness element properties			/	/			KPROP
• Strength-designed margins of safety			/	/	/		SMS
• Thermal-designed margins of safety			/	/	/		TMS
MODE SHAPES							
• Vibration		/					VMODE
• Buckling		/					BMODE
FUEL/PAYLOAD MANAGEMENT							
• Loadability diagrams					/		LOADAB
FLUTTER SOLUTION							
• V-g and V-f graphs					/		VGVT
MATRIX DATA							
• Any ATLAS matrix						/	see sec. 218.1.11

228.1 DISPLAY OPTIONS

Descriptions and examples of the various options provided by the GRAPHICS Postprocessor to create each of the six types of plots identified by table 228-1 are presented in this section.

228.1.1 Undeformed Geometry Plots

The geometry of the nodes and finite elements used to define a structural and/or mass model can be plotted to any scale and relative to any viewing position selected by the user (see sec. 228.2). Pictorial, GLOBAL-axis and exploded views of the model are created via orthographic projection.

Nodes can be displayed according to the following options:

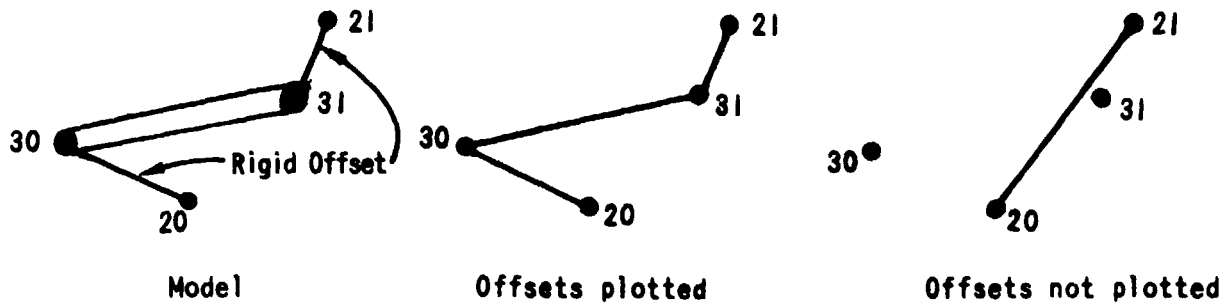
- a) A network of points
- b) A network of points annotated with user node numbers as shown in figure 228-1.

A mid-surface node, which may have ΔX , ΔY and ΔZ GLOBAL coordinates, can be plotted either as a point at the mid-surface or as three distinct points -- one at the mid-surface and one at each of the surfaces. If three points are displayed, they are connected by a straight line to illustrate their relationship as shown in figure 228-2.

Stiffness and mass element grids can be displayed according to the following options:

- a) Straight line segments denoting the nodal connectivity
- b) Same as (a) with user node numbers (see fig. 228-3)
- c) Same as (a) with user element numbers (see fig. 228-4)
- d) Same as (a) with element-type numbers (1=ROD, 2=BEAM, etc.). Reference figure 228-5.
- e) Same as (c) plus user node numbers
- f) Same as (d) plus user node numbers

Elements with offsets can be displayed either in the offset position or by connectivity of the structural nodes defining the element. An offset BEAM element, for example, that is defined by nodes 20, 21, 30 and 31 can be plotted according to the following sketch.



Element numbers are displayed at the geometric centroids of the elements. SPAR, COVER and CCOVER elements, which are defined by mid-surface nodes, are plotted as shown in figure 228-5.

An "exploded view" of the nodes and/or stiffness and mass elements of a model can be created. The relative position (separation) and orientation of each component of the "exploded" model are specified by the user. An example is shown in figure 228-6. The plot options selected to display the individual node and/or element subsets can be different.

228.1.2 Deformed Geometry Plots

The deformed geometry of a structural model that corresponds to a statically loaded model, vibration mode shapes or general-instability buckling mode shapes can be plotted to any scale and relative to any viewing position selected by the user (see sec. 228.2). Pictorial and GLOBAL-axis views of the model are created via orthographic projection.

The options provided for creation of deformed geometry plots are:

- a) Points plus vectors -- The points identify the undisplaced nodes, and the scaled vectors, which start at the nodes, represent the corresponding displacements (see fig. 228-7).
- b) Undeformed grid plus vectors -- Straight line segments denoting the nodal connectivity of the undeformed model plus vectors that represent the nodal displacements. If static displacements are plotted, the stiffness element grid is generally displayed (see sec. 218.1.1). Optionally, an ordered node subset can be used to generate a grid for static displacement plots in lieu of displaying the stiffness element grid. If mode shapes are plotted, the grid is defined by the user via an ordered node subset (see the BSUB=ONxxx parameter for the EXECUTE EXTRACT statement in sec. 218.1). An example is shown in figure 228-8.

- c) Deformed grid -- Dashed, straight line segments denoting the nodal connectivity of the deformed model (see fig. 228-9). The grid that is displayed is the same as described in (b).
- d) Undeformed grid plus deformed grid as shown in figure 228-10.
- e) Same as (d) plus the displacement vectors as shown in figure 228-11.

The node and element-number labeling options described in section 228.1.1 can also be used for the generation of deformed geometry plots corresponding to static loads. Only the node labeling option, however, can be used for mode-shape plots. It should be noted that mode-shape displacements are only associated with the retained nodes of the model. Any of the nodes used to define the model can be included in an ONxxx subset for generation of a deformed and/or undeformed grid for a mode-shape plot. If both retained and non-retained nodes are included in ONxxx, and a deformed grid is plotted, the line segments of the grid are passed through the retained nodes in their displaced position and through the non-retained nodes in their undisplaced position. This produces a distorted deformed grid.

228.1.3 Scalar-Grid Plots

Stiffness-element properties, stress and "designed" margins of safety can be displayed via the scalar-grid plot-type option. The stiffness element grid is projected orthographically onto the display plane during plot generation. The viewing position and scale are selected by the user (see sec. 228.2). Straight line segments, whose lengths correspond to the scaled values of the data component being plotted, are drawn perpendicular to the display plane and at the geometric centers of the respective elements. The display is then reoriented automatically to a predefined position to generate a meaningful view of the data.

Only one property component, one stress component or one minimum margin of safety can be displayed via one plot of this type. Thus, for example, PLATE shear stresses, SPAR upper-chord areas, or COVER margins of safety can be displayed as scalar-grid plots. An example scalar-grid plot is shown in figure 228-12. The node and element-number labeling options described in section 228.1.1 can be used for creating scalar-grid plots.

228.1.4 Contour Plots

Contour plots of stiffness-element properties, stresses and "designed" margins of safety can be created. The isocurves generated for a particular plot define the constant levels of one property component, one stress component or one minimum margin of safety for one element type. Thus, for example, isocurves can be generated for PLATE shear stresses, BEAM area moments of inertia, COVER margins of safety, etc.

The values of the scalar data component to be plotted are associated with the geometric centers of the respective elements. These data points are projected orthographically onto the display plane during plot generation. The viewing position is selected by the user (see sec. 228.2).

Two algorithms are provided for generation of contour plots as described below.

- a) Rectangular Mesh -- The rectangular boundary of the contour plot is defined by the minimum and maximum coordinates of the data points. Values at the data points are interpolated to the intersection points of a uniform grid within the rectangular boundary prior to generation of isocurves. The uniform grid, displaced according to the interpolated data values, can be displayed in addition to the contour plot as shown in figure 228-13.
- b) Triangular Mesh -- The boundary of the contour plot is defined by an ordered node subset (see the BSUB=ONxxx parameter for the EXECUTE EXTRACT statement in sec. 228.1). The sequence of nodes identified by ONxxx which, when connected in the specified order, establishes the boundary. The generated boundary is a contiguous series of line segments that defines a simply-connected curve. Using the boundary nodes and the projected data points, the program automatically generates a mesh of adjacent triangles over the enclosed region. Values at the data points are linearly interpolated during generation of the isocurves. The triangulated grid can be displayed in addition to the contour plot as shown in figure 228-14.

There is no limit on the number of data points (e.g. stress components) that can be plotted by the rectangular mesh option. The number of the boundary nodes plus the number of the data points that can be plotted by the triangular mesh option, however, must be less than 1000. Thus, for example, a contour plot could be created by using 900 shear stresses for 900 PLATE elements and 100 nodes to define the boundary.

228.1.5 Graph Plots

Graph plots of stiffness-element stresses and "designed" margins of safety, flutter-solution data, as well as payload and fuel condition data can be created according to the following descriptions.

228.1.5.1 Element Stresses Versus Loadcases

The variation of one element-stress component (e.g. BMMY(1) for a BEAM element as selected from table 156-1) can be plotted in graph form for selected elements and loadcases as shown by figure 228-15. The particular stress component to be displayed is identified by a parameter in the EXECUTE GRAPHICS statement (sec. 228.4), whereas the loadcases and elements are identified by the LC and ESUB parameters in the EXECUTE EXTRACT statement (sec. 218.1). Since one plot is generated for each element in ESUB, caution should be exercised not to exceed the limit of 40 plots per GNAME (see sec. 228.3).

228.1.5.2 Element Margins of Safety Versus Design Cycles

The variation of the margin of safety for one type of element (e.g. BRTMS for a BRICK element as selected from table 156-1) can be plotted in graph form for selected elements and design cycles. This type of plot is similar to that shown in figure 228-15. The particular margin of safety to be displayed is identified by a parameter in the EXECUTE GRAPHICS statement (sec. 228.4), whereas the design cycles and elements are identified by the CYCLE and ESUB parameters in the EXECUTE EXTRACT statement (sec. 218.1). Since one plot is generated for each element in ESUB, caution should be exercised not to exceed the limit of 40 plots per GNAME (see sec. 228.3).

228.1.5.3 Flutter Solution Graphs

Velocity-damping (V-g) and/or velocity-frequency (V-f) plots of flutter solutions can be generated for selected combinations of CASE number, CONDITION number, altitude, change-set number and retain-set number. Each graph contains "N" curves; each curve corresponds to a particular generalized degree of freedom (e.g. a structural vibration mode). The curves are identified with the letters A, B, ..., N so that the V-g plot curves can be paired with the corresponding curves on the V-f plots (see figures 228-16 and 228-17). The number of reduced frequencies used to solve the flutter problem defines the number of points used to generate each curve.

228.1.5.4 Loadability Diagrams

The passenger, cargo and fuel management data can be plotted as loadability diagrams which show how the center-of-gravity location varies with respect to loading conditions (see fig. 228-18). The fan grid used to plot the loading vectors is calculated from parameters specified in the EXECUTE GRAPHICS statement (sec. 228.4).

228.1.6 Matrix Data Plots

The contents of any ATLAS data matrix that has been extracted from the primary ATLAS data base can be displayed graphically. An example of this type of plot is shown in figure 228-19.

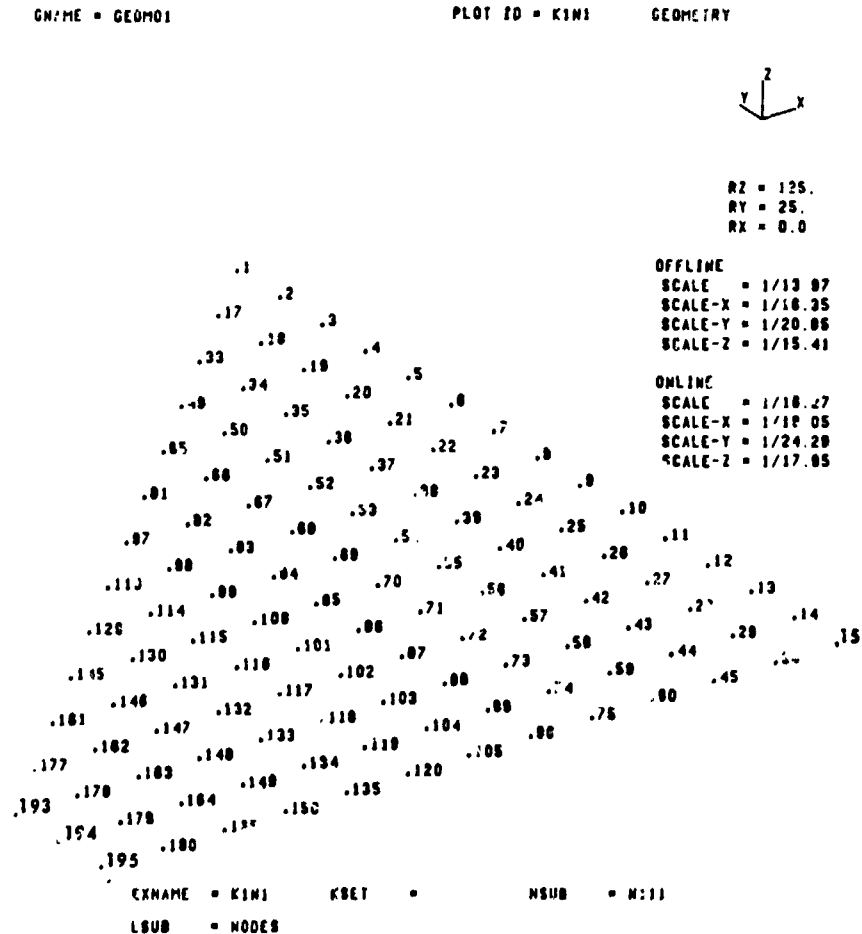


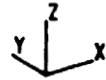
Figure 228-1. Point Plot of Nodes

GRAPHICS

GNAME = GEOM01

PLOT ID = KIN1

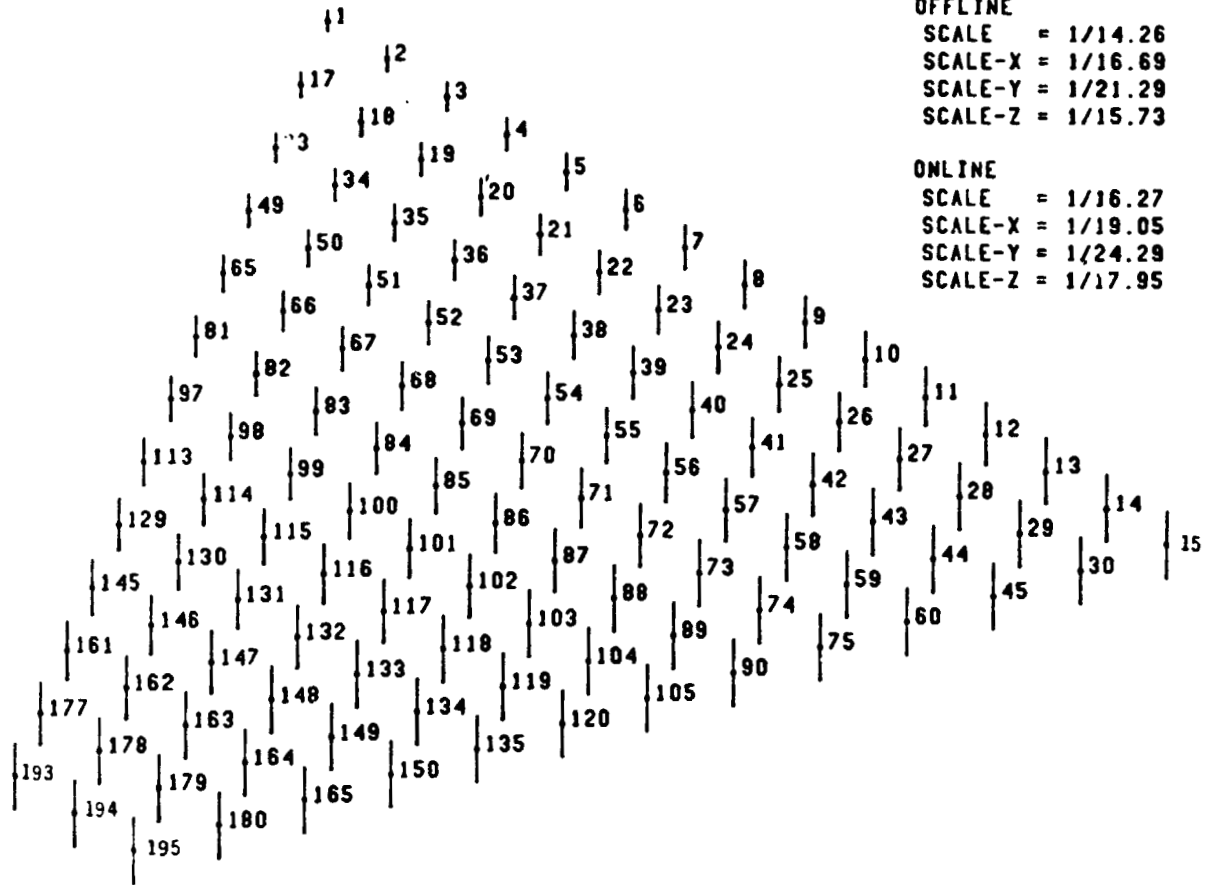
GEOMETRY



RZ = 125.
RY = 25.
RX = 0.0

OFFLINE
SCALE = 1/14.26
SCALE-X = 1/16.69
SCALE-Y = 1/21.29
SCALE-Z = 1/15.73

ONLINE
SCALE = 1/16.27
SCALE-X = 1/19.05
SCALE-Y = 1/24.29
SCALE-Z = 1/17.95



EXNAME = KIN1 KSET = 1 NSUB = N111
LSUB = NODES

Figure 228-2. Point Plot of Mid-Surface Nodes

C-6

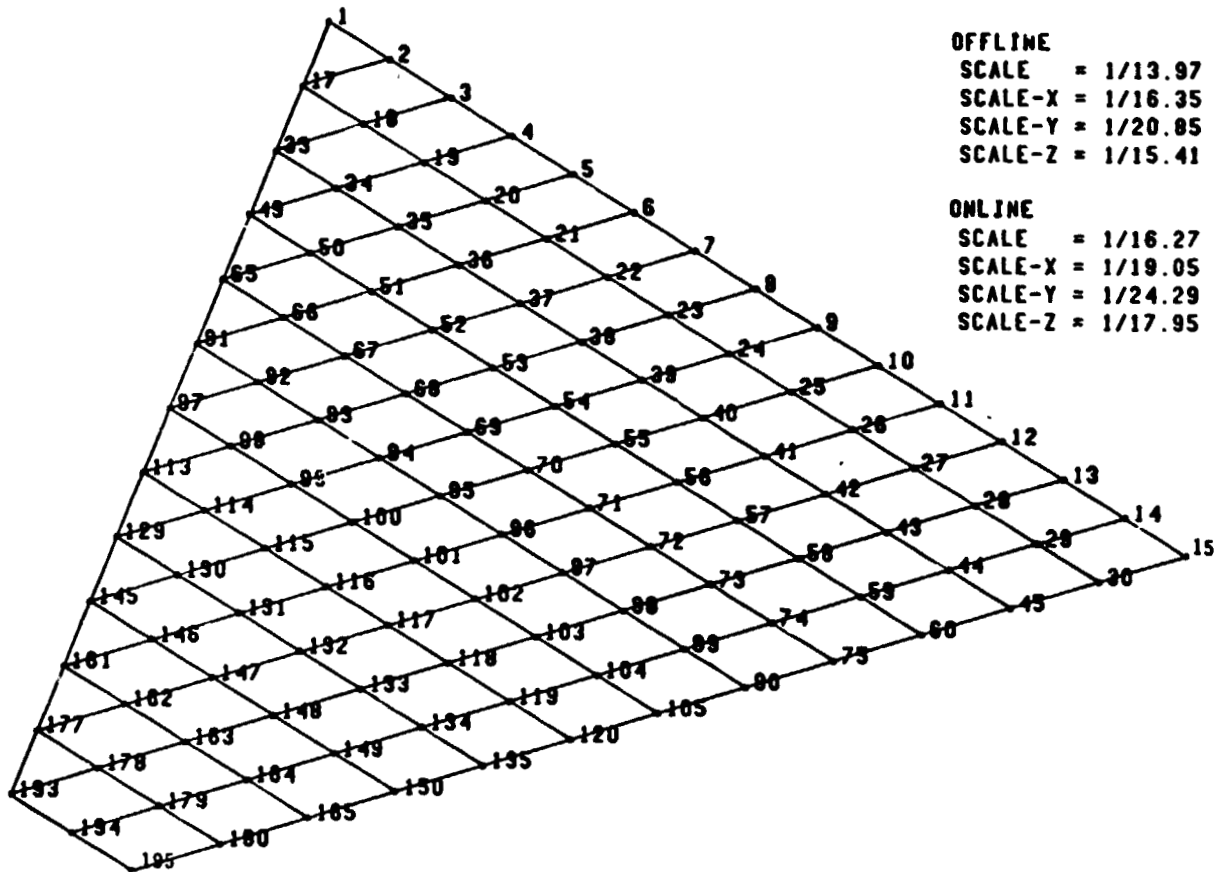
GNAME = GEOM01

PLOT ID = K1E1

GEOMETRY



RZ = 125.
RY = 25.
RX = 0.0



OFFLINE
SCALE = 1/13.97
SCALE-X = 1/16.35
SCALE-Y = 1/20.85
SCALE-Z = 1/15.41

ONLINE
SCALE = 1/16.27
SCALE-X = 1/19.05
SCALE-Y = 1/24.29
SCALE-Z = 1/17.95

EXNAME = K1E1 KSET = 1 NSUB = N111
ESUB = E001 LSUB = KGRID

Figure 228-3. Stiffness Element Grid with Node Numbers

GRAPHICS

GNAME = GEOM01

PLOT ID = K1E1 . GEOMETRY

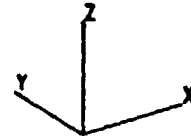
OFFLINE

SCALE = 1/6.917

SCALE-X = 1/8.097

SCALE-Y = 1/10.324

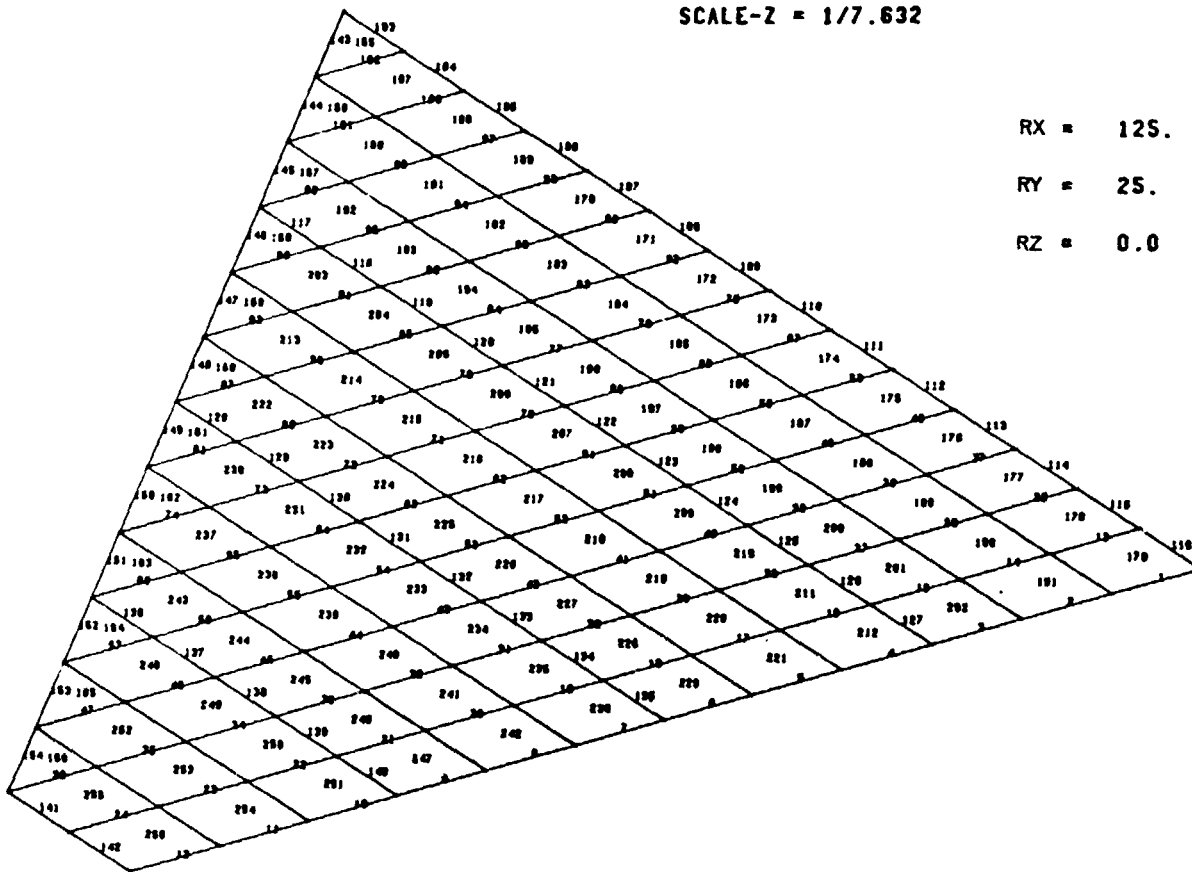
SCALE-Z = 1/7.632



RX = 125.

RY = 25.

RZ = 0.0



EXNAME = K1E1

KSET = 1

NSUB = N111

ESUB = E001

LSUB = KGRID

Figure 228-4. Stiffness Element Grid with Element Numbers

GNAME = GEOM01

PLOT ID = K1E1

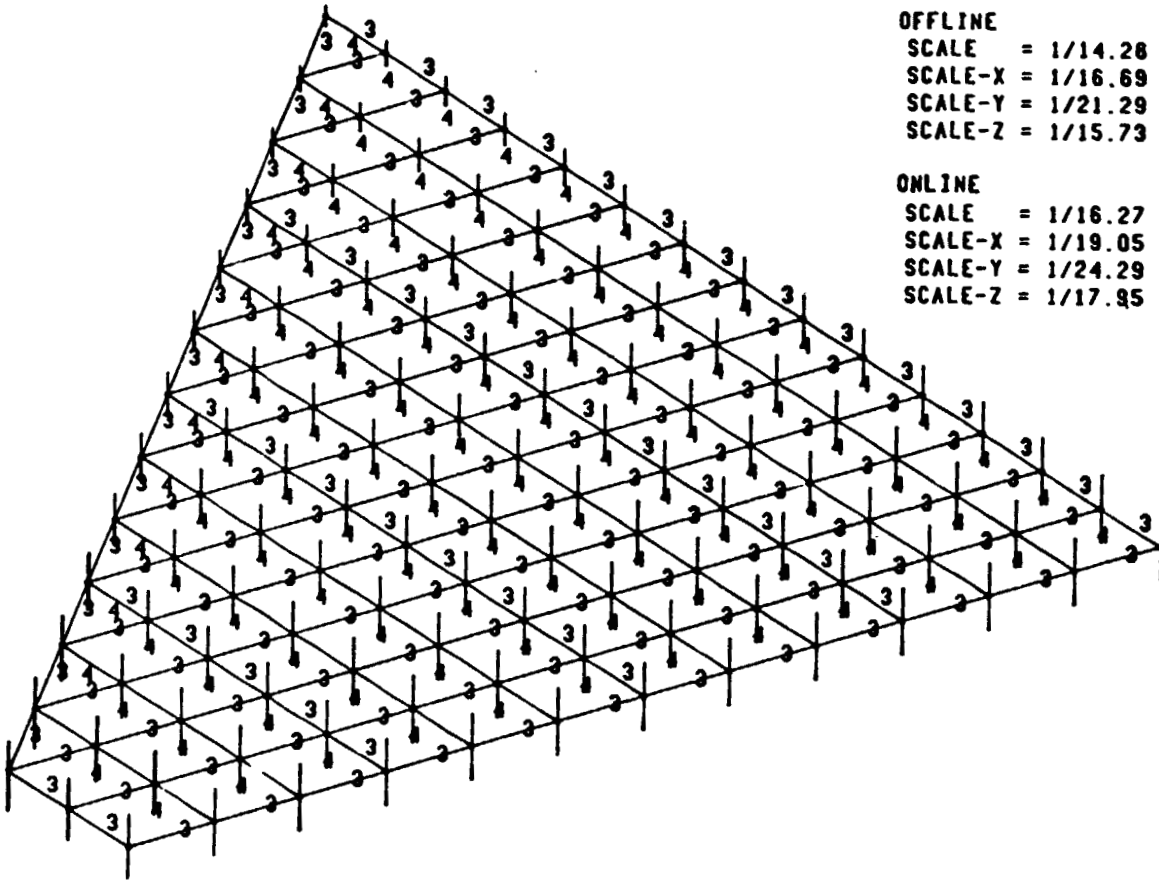
GEOMETRY



RZ = 125.
RY = 25.
RX = 0.0

OFFLINE
SCALE = 1/14.28
SCALE-X = 1/16.69
SCALE-Y = 1/21.29
SCALE-Z = 1/15.73

ONLINE
SCALE = 1/16.27
SCALE-X = 1/19.05
SCALE-Y = 1/24.29
SCALE-Z = 1/17.95

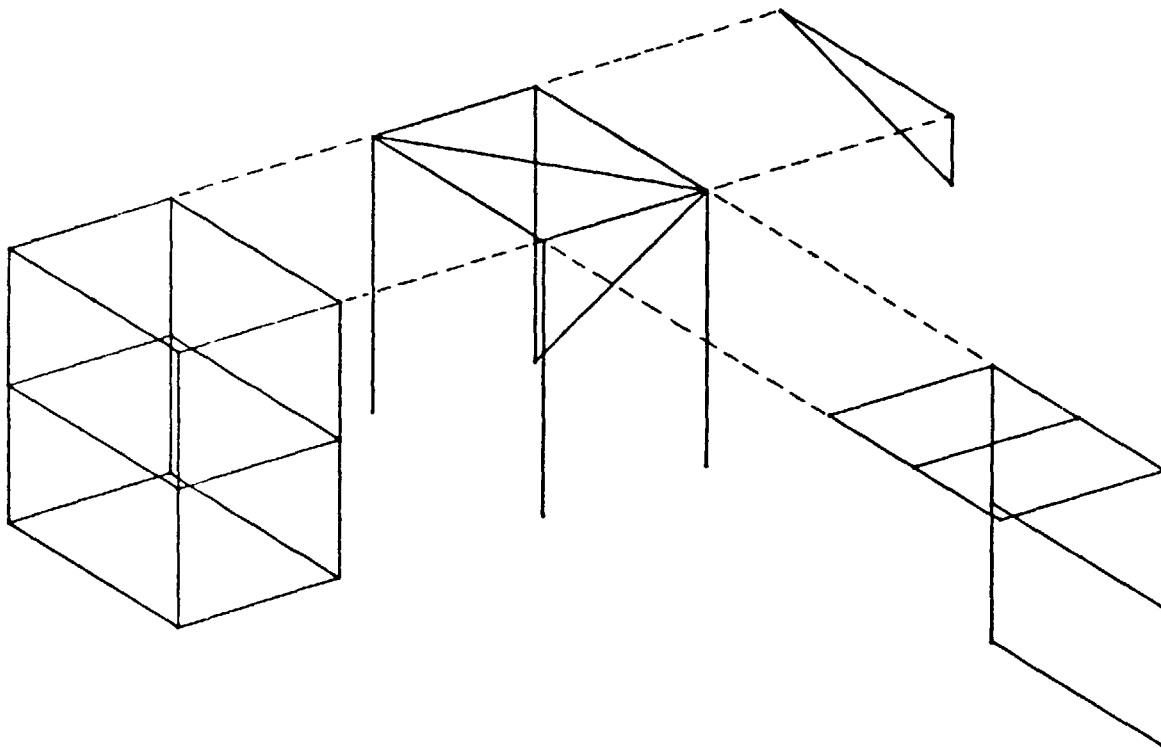


EXNAME = K1E1 KSET = 1 NSUB = N111
ESUB = E001 LSUB = KGRID

Figure 228-5. Stiffness Element Grid with Element-Type Numbers

GNAME = GEOM02

PLOT ID = EXPLODED GEOMETRY



EXNAMES = K3E1 . K3E2 . K3E3 . K3E4 .

Figure 228-6. Exploded Geometry Plot

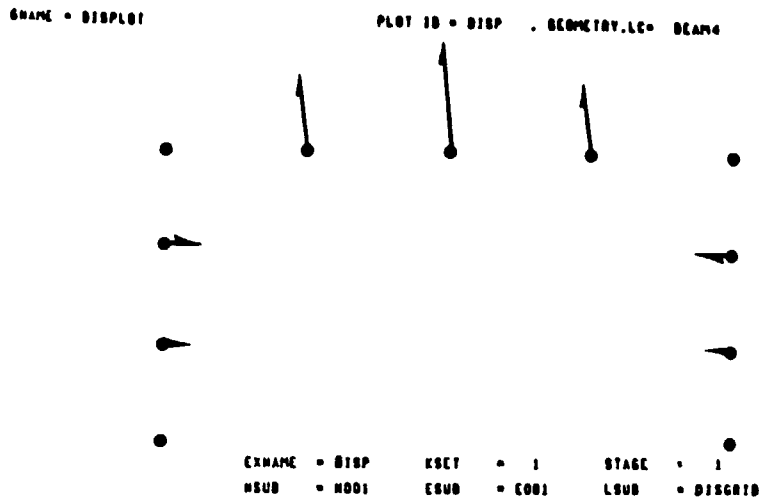


Figure 228-7. Deformed Geometry Plot -- Points Plus Vectors

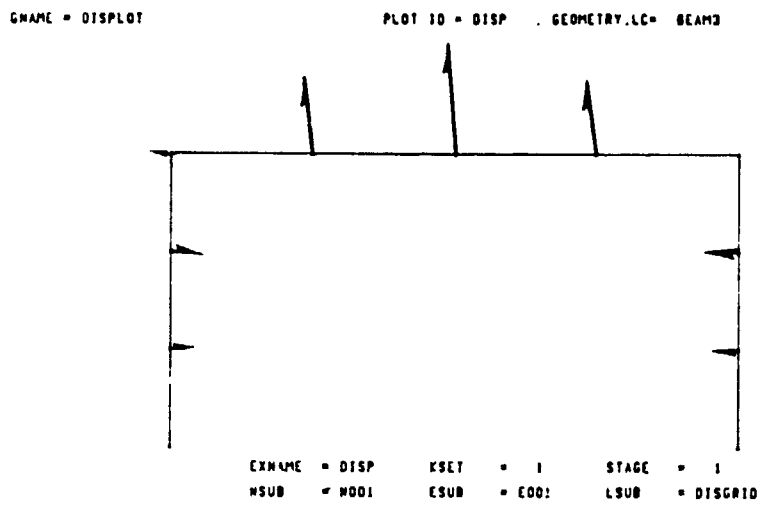


Figure 228-8. Deformed Geometry Plot -- VECTOR3 Option

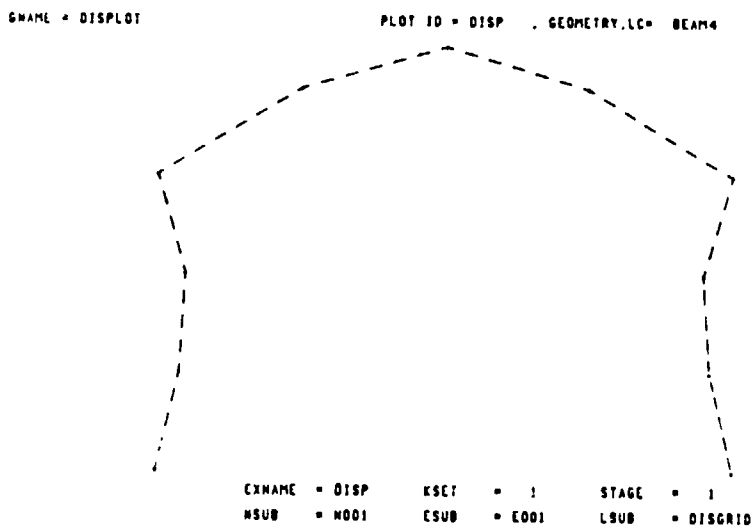
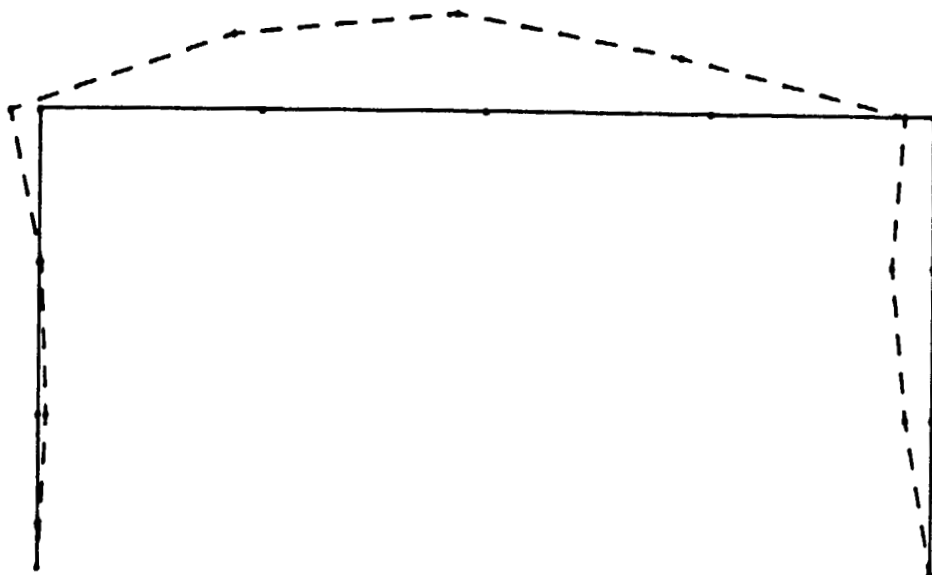


Figure 228-9. Deformed Geometry Plot -- VECTOR4 Option

GNAME = DISPLOT

PLOT ID = DISP . GEOMETRY,LC= BEAM2

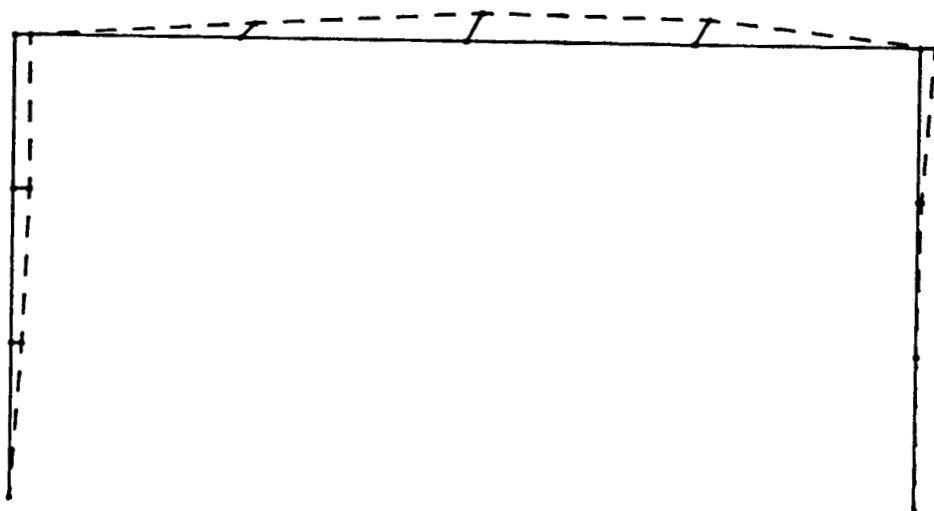


EXNAME = DISP KSET = 1 STAGE = 1
 NSUB = NOO1 ESUB = E001 LSUB = DISGRID

Figure 228-10. Deformed Geometry Plot -- VECTOR2 Option

GNAME = DISPLOT

PLOT ID = DISP . GEOMETRY,LC= BEAM1

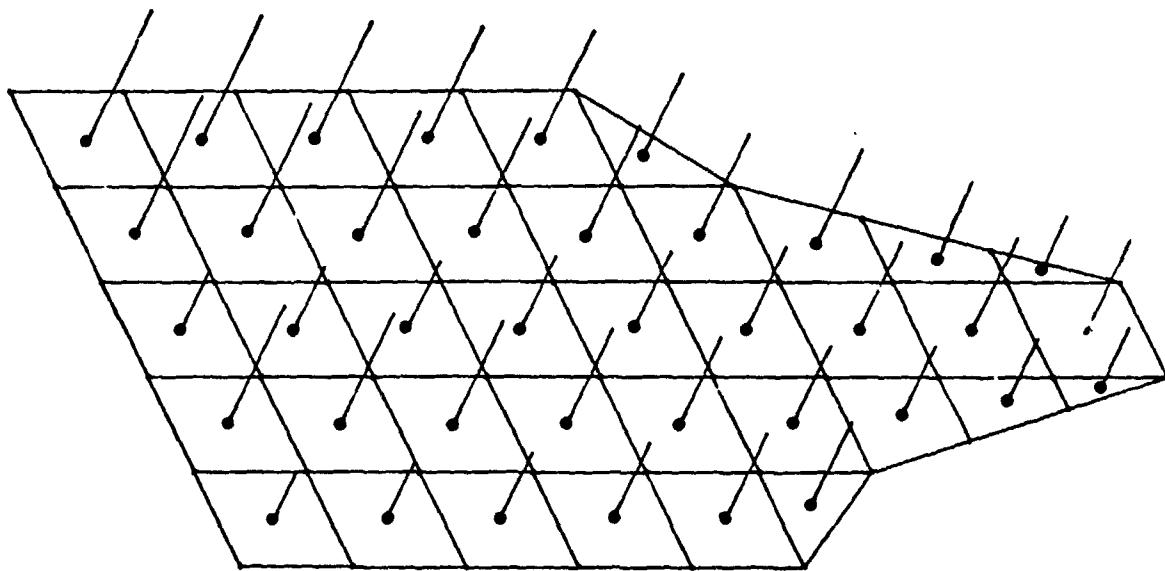


EXNAME = DISP KSET = 1 STAGE = 1
 NSUB = NOO1 ESUB = E001 LSUB = DISGRID

Figure 228-11. Deformed Geometry Plot -- VECTOR1 Option

GNAME = PLPROP

PLOT ID = PROP , PROPERTY FLT (0)

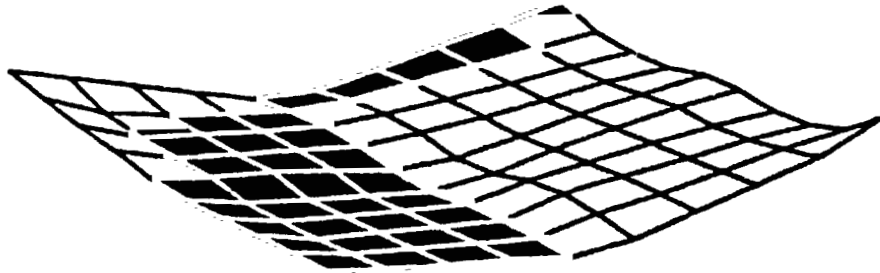


EXNAME	=	PROP	KSET	=	1	MSUB	=	NO01
ESUB	=	E001	BSUB	=	P001	LSUB	=	KPROP

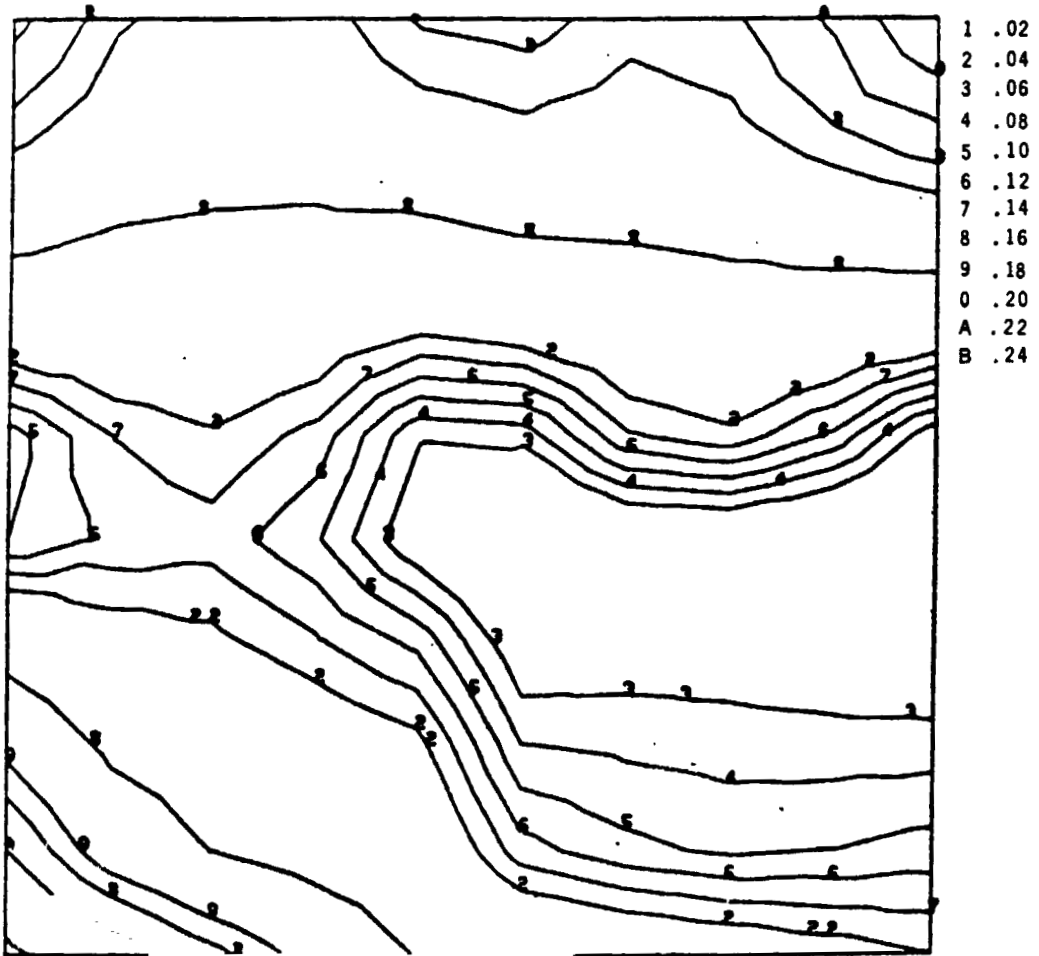
Figure 228-12. Scalar - Grid Plot

GNAME = PLPROP

PLOT ID = PROP , PLTS (1) CONTOURS



a) Displaced Grid

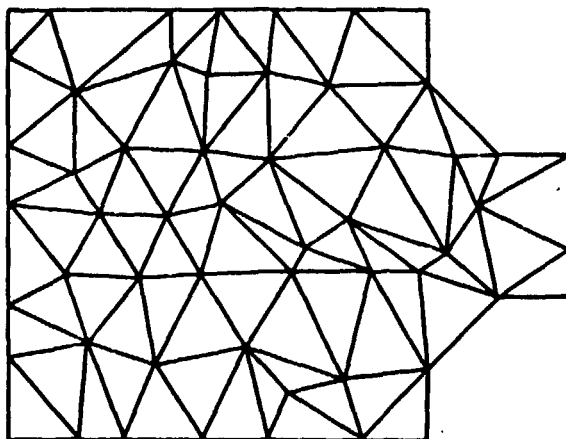


b) Contour Plot

Figure 228-13. Contour Plot -- Rectangular Mesh

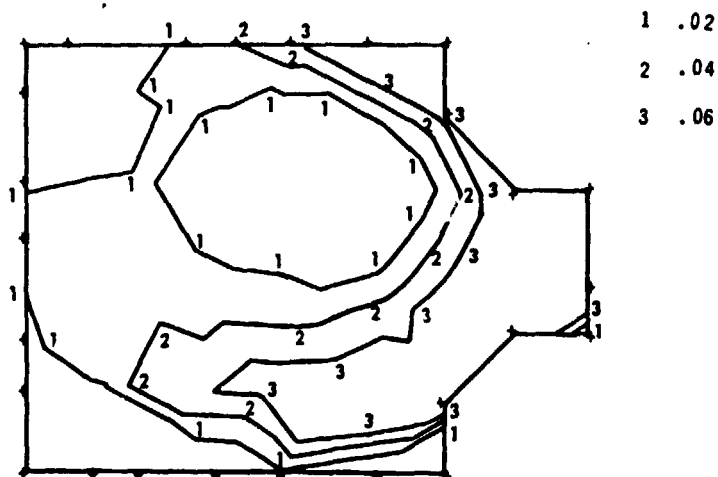
GNAME = PLPROL

PLOT 1 D = PROP , PLT (0) CONTOURS



a) Triangulated Grid

EXNAME = PROP KSET = 1 NSUB = N001
ESUB = E001 BSUB = P001 LSUB = KPROP



b) Contour Plot

Figure 228-14. Contour Plot -- Triangular Mesh

GNAME = STREPLOTT

PLOT ID = BMMZ(1) vs. LC. ELEM = 7

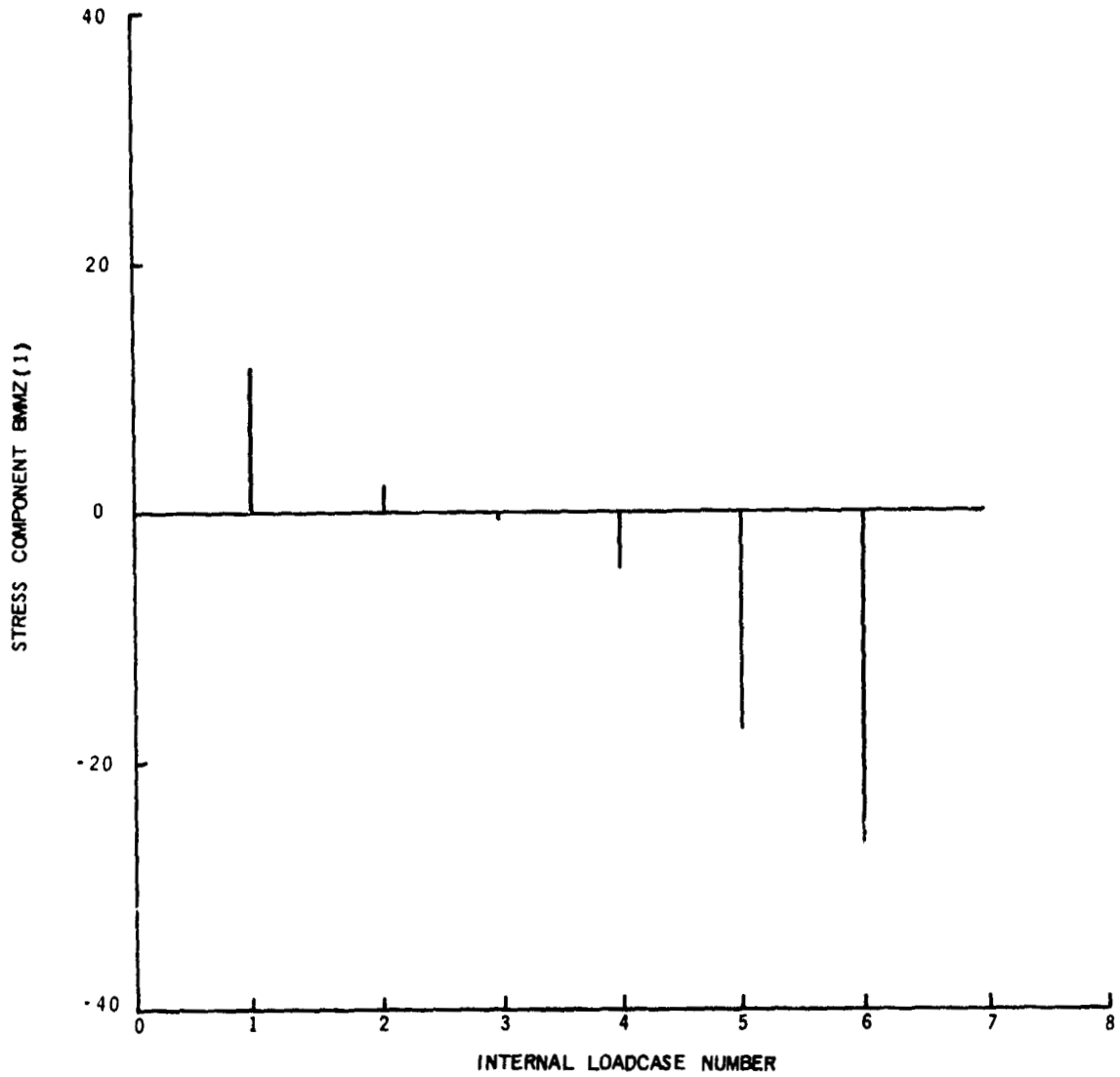


Figure 228-15. Graph Plot -- Stress vs Loadcases

GNAME = VGVFPLOTS

PLOT 1 D = V-G, CASE = 12, CO = 1, CS = 5, RS = 3

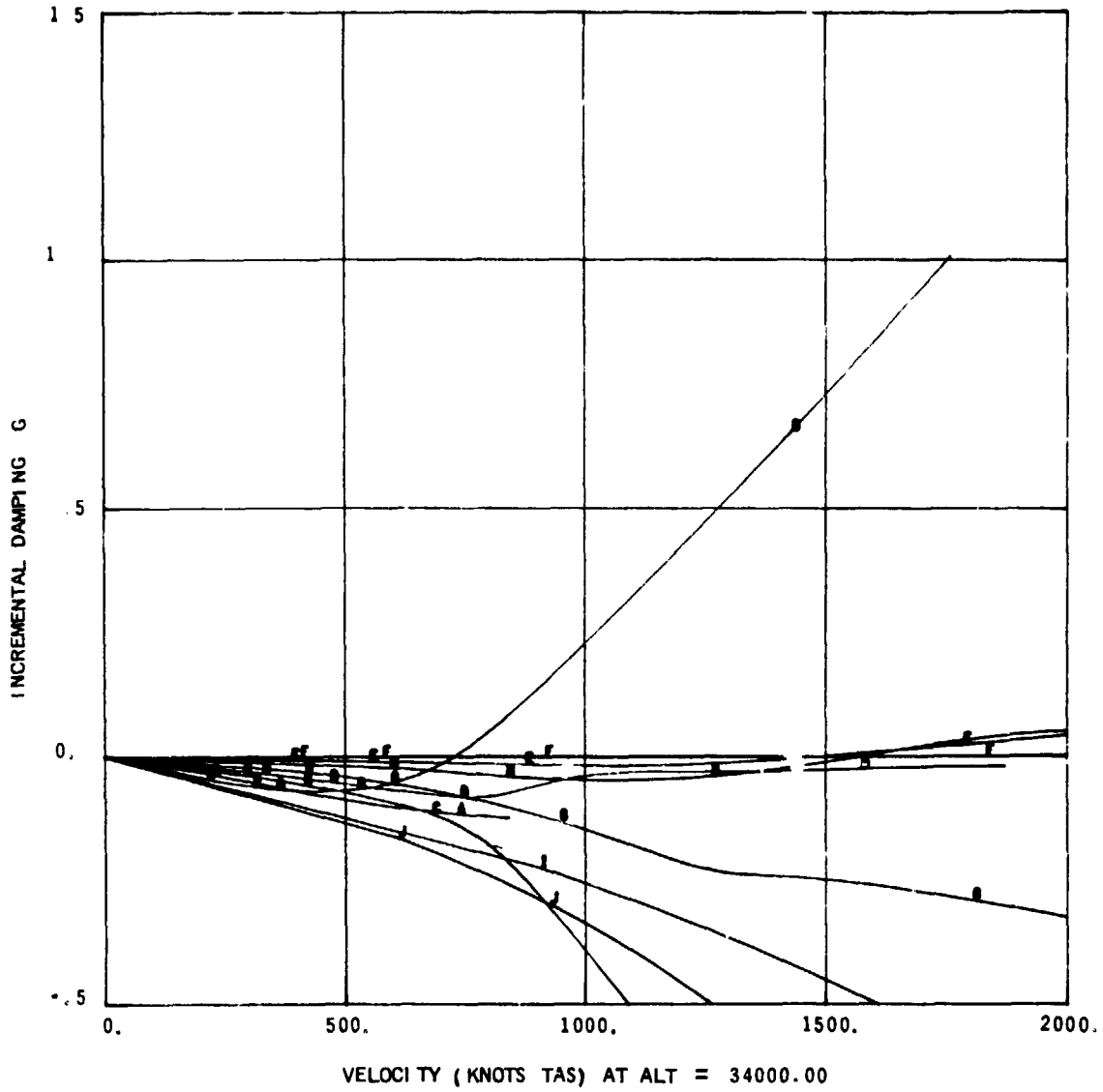


Figure 228-16. Flutter V-g Graph

GNAME = VGVFPLOTS

PLOT 1 D=V-F, CASE=12, CO=1, CS=5, RS=3

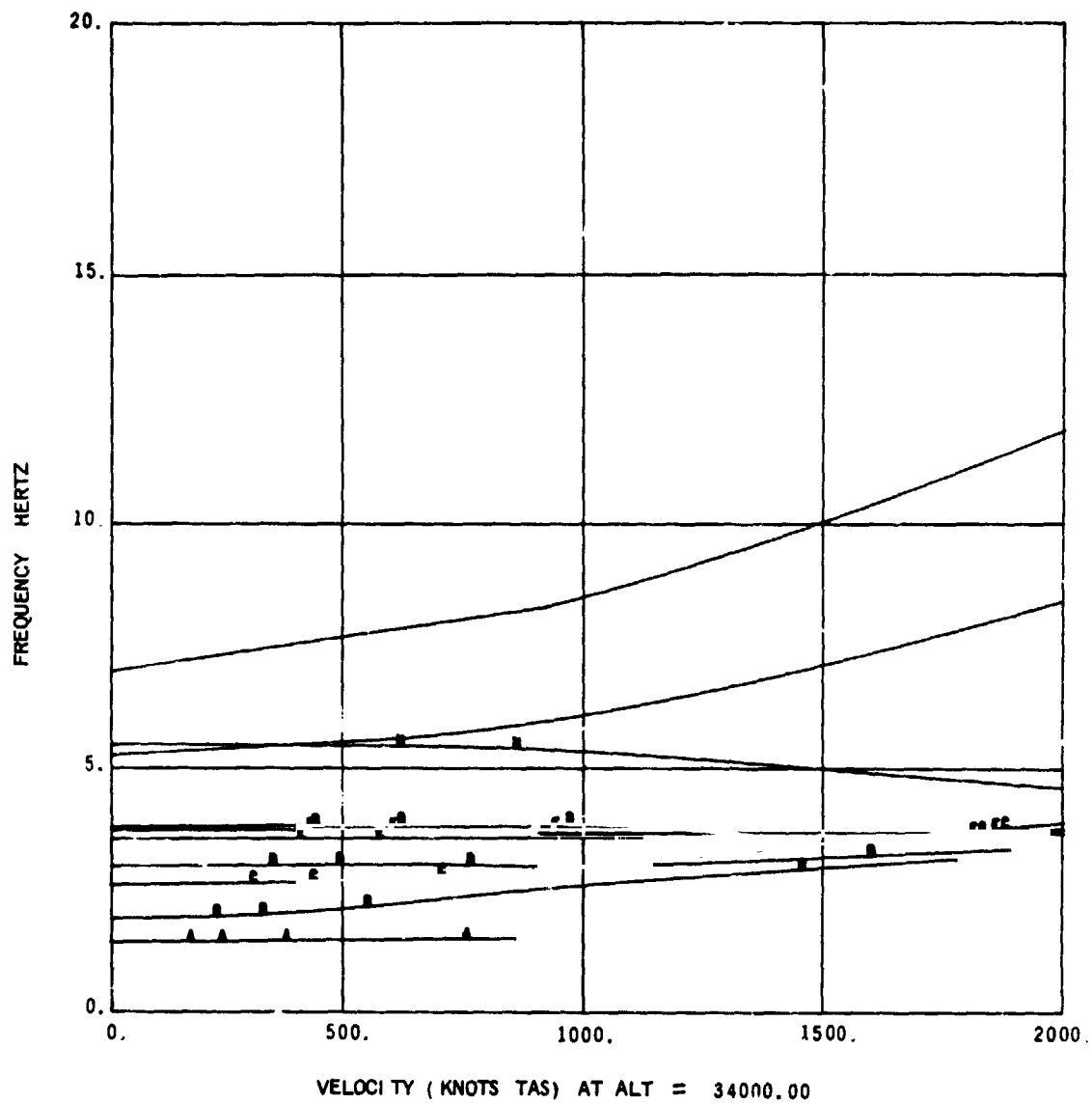


Figure 228-17. Flutter V-f Graph

GNAME = LOADABILITY

PLOT ID = LOADABILITY DIAGRAM

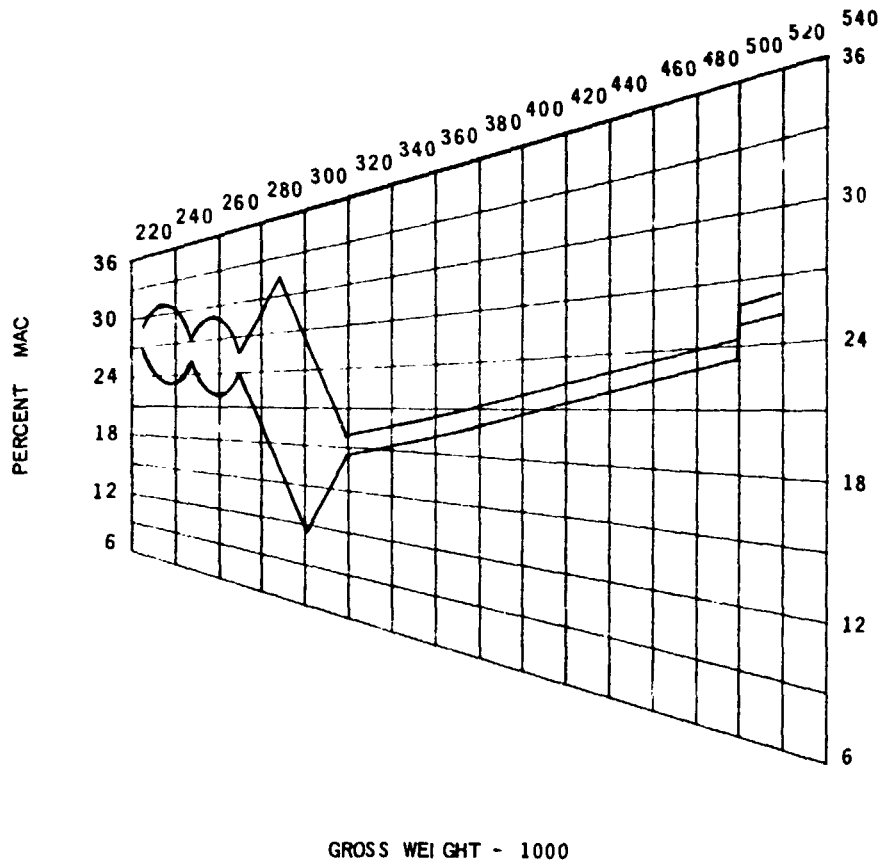


Figure 228-18. Loadability Diagram

RECORD 1 MATRIX NAME : MATRIX IN NUMERICAL FORM KEY : (1,1) = (1,1)

ROWS 100 COLUMNS 10

	1	2	3	4	5
1	1.	101.	201.	301.	401.
2	2.	102.	202.	302.	402.
3	3.	103.	203.	303.	403.
4	4.	104.	204.	304.	404.
5	5.	105.	205.	305.	405.
6	6.	106.	206.	306.	406.
7	7.	107.	207.	307.	407.
8	8.	108.	208.	308.	408.
9	9.	109.	209.	309.	409.
10	10.	110.	210.	310.	410.
11	11.	111.	211.	311.	411.
12	12.	112.	212.	312.	412.
13	13.	113.	213.	313.	413.
14	14.	114.	214.	314.	414.
15	15.	115.	215.	315.	415.
16	16.	116.	216.	316.	416.
17	17.	117.	217.	317.	417.
18	18.	118.	218.	318.	418.
19	19.	119.	219.	319.	419.
20	20.	120.	220.	320.	420.

Figure 228-19. Matrix Data Plot

228.2 ORIENTATION OF ORTHOGRAPHIC PROJECTIONS

Orientation of objects relative to the display plane for creation of orthographic plots (geometry, scalar-grid and contour plots) is selected by the user. The orientation of an object relative to the observer is defined either by a viewing position or by object-rotation angles.

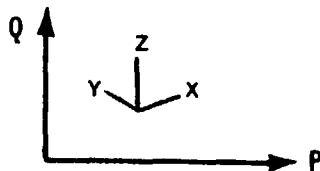
The display (orthographic projection) plane is defined as the P-Q plane in the observer's reference system R-P-Q shown in figure 228-20. This triad coincides with the initial orientation of the GLOBAL X-Y-Z reference frame. Viewing is always in the negative direction of the R-axis. The P-Q axes always define the horizontal and vertical coordinates, respectively, of the plots. The R-P-Q triad is a stationary reference frame, whereas the X-Y-Z triad associated with the object can be translated and rotated to effect any view of the object. The positioned object is projected orthographically onto the P-Q display plane during plot creation.

Orientation of an object to effect a certain pictorial view is specified by a sequence of rotations of the model relative to the R-P-Q axes. These rotations, RX, RY and RZ, are measured relative to the R-P-Q axes as shown in figure 228-20. The positive directions of the positioning angles are shown in this figure. The default sequence and magnitude of rotations of an object displayed via orthographic or contour plots are:

$$RZ = 125^\circ$$

$$RY = 25^\circ$$

$$RX = 0^\circ$$



Rotations are not commutative. This is illustrated in figure 228-21 by positioning a frame model by one set of sequential 30-degree rotations about the Q-P-R axes, respectively. The final position of the model, figure 228-21(d), illustrates the P-Q (display) plane orthographic projection which is the pictorial view that would be plotted. Note that each of the other five sets of sequential 30-degree rotations would effect different pictorial views than the one illustrated in figure 228-21. Thus, the order in which RX, RY and RZ are specified is crucial when establishing a pictorial view.

A second method of orienting an object for orthographic projection is to specify the viewing position in terms of the GLOBAL coordinates VX, VY and VZ of a point. A vector is automatically generated from the GLOBAL origin through this point. The object is rotated through an RZ angle followed by an

RY angle such that the vector coincides with the R-axis. The positioned object is projected orthographically onto the display plane. Figure 228-22 illustrates the two sequential rotations performed to orient an object by this method.

When an object is to be viewed in the direction of one of the GLOBAL axes, the VIEW parameter may be used in the EXECUTE GRAPHICS statement (sec. 228.4) in lieu of specifying 90° increments for RX, RY and RZ. The orientation of the GLOBAL axes relative to the P-Q plot plane for each of these GLOBAL-axis views is shown in figure 228-23.

Orientation of a model should be selected such that the desired view is effected and such that the plotting area is utilized advantageously. For example, when CALCOMP or GERBER plots are generated, the model should be oriented relative to the display plane such that the largest dimension of the plot paper is parallel or nearly parallel to the P-axis of the display plane.

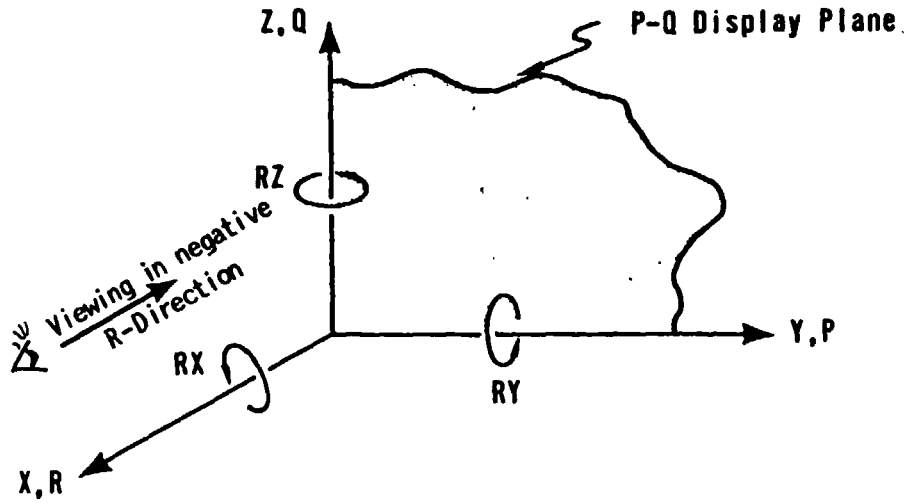


Figure 228-20. GLOBAL/Display Axes

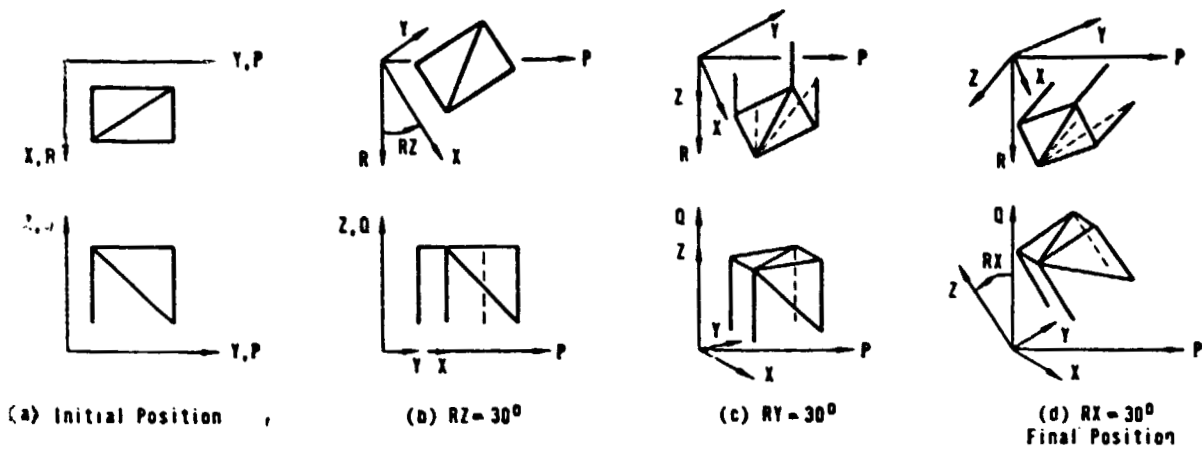


Figure 228-21. Model-Orientation Under Sequential Rotations

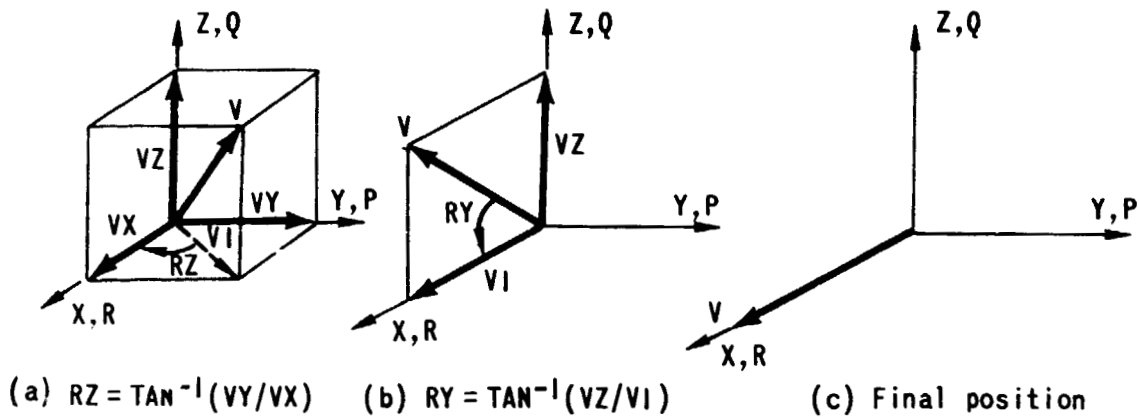


Figure 228-22. Equivalent Rotations for a VX-VY-VZ Orientation

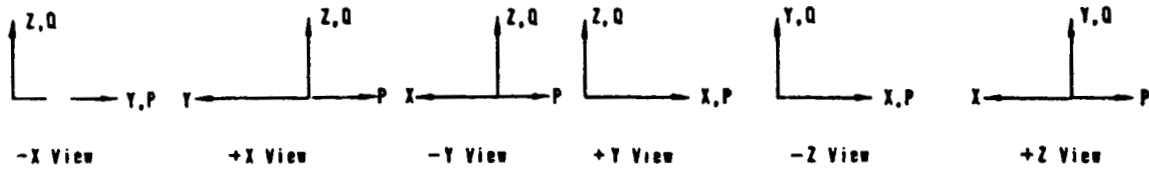


Figure 228-23. Global-Axis View Orientations

228.3 MANAGEMENT OF GRAPHICS DATA

All ATLAS input data and analysis data are stored in various files within the ATLAS data base (ref. sec. 200.2). Those data to be displayed graphically must first be extracted from the primary data base by execution of the EXTRACT Postprocessor (sec. 218.0). The extracted data blocks are stored in the file named EXTRRNF which is the only ATLAS data file interrogated by the GRAPHICS Postprocessor. Each extracted data block, identified by an EXNAME, may be used to create multiple plots.

Three different data files are generated sequentially by the GRAPHICS Postprocessor. The contents of these files and the management of graphics-related data are described below (ref. fig. 228-24).

- a) G/F (GRAPFIL) File -- Plot-definition (display) data for EXNAMES selected strictly from the EXTRRNF file. The types of plots to be created from an EXNAME are specified by the parameters of the EXECUTE GRAPHICS statement. A maximum of 100 different groups of plots (GNAMES) can be defined by the user. For each GNAME group, a maximum of 40 plots can be created.
- b) V/F (VECTFIL) File -- Display-vector data corresponding to plots previously identified via GNAMEs in the G/F file. The V/F data are generated strictly from the G/F data and are independent of any plotting hardware device to be used.
- c) P/F (TAPE99) File -- Hardware-dependent instructions for subsequent generation of offline plots by a particular plotting device (SC4020, CALCOMP or GERBER). Multiple P/F files can be generated from a single V/F file via multiple ATLAS jobs as discussed below.

Generation of either the G/F, the G/F and V/F, or the G/F, V/F and P/F files for the plots defined by an EXECUTE GRAPHICS statement is controlled by the user via the OFFLINE parameter in the GRAPHICS statement. The G/F and/or the V/F files can be used in subsequent ATLAS jobs during graphics-execution restarts as described in section 228.6.

When the GRAPHICS Postprocessor is executed in the batch mode, each of the graphics data files that is generated contains the respective information for each plot requested via the EXECUTE GRAPHICS statements. During interactive execution, however, the user controls which plots are to be selected from the G/F file, displayed on the terminal screen, and written onto the V/F file for subsequent processing. The user conducts a

conversational dialogue with ATLAS in the interactive mode by using menu options to select plots, generate new displays and enlarge selected regions of plots by zooming. Any new displays of selected plots created online can also be written on the V/F file. The online directives are defined via the keyboard and/or crosshairs provided by the console. Descriptions of the interactive graphics capabilities are presented in section 228.5.

The P/F data file, if requested, is always created at the end of an ATLAS job. Since the operating system of the computer supports the generation of only one PLOTFILE per job, only one offline plot device should be identified per job. If different devices are specified via multiple GRAPHICS-execution directives in one job, all offline plot data are generated for the first device identified by the sequence of directives.

When executing in the batch mode, the plot data in TAPE99 (P/F) are automatically transferred to magnetic tape for the selected offline processing by use of the PLOTFIL control statement described in section 11.2. During interactive processing, however, offline plot data are not managed automatically. If offline plots are to be generated after interactive execution of the GRAPHICS module, the user must first SAVE or REPLACE the TAPE99 file followed either by a SUBMIT command or by submittal of a regular batch job.

Consider, for example, an interactive execution of ATLAS during which a TAPE99 (P/F) file for CALCOMP plots was created and saved. A batch job comprised of the following CDC 6600 control cards could then be used to generate the desired plots.

```
Job Card
Account Card
GET,TAPE99
PLOTFIL(CALCOMP,TAPE99,0)
End-of-File
```

An alternate way of generating offline displays of plots created during an interactive execution of ATLAS is illustrated by the following steps.

- a) Identify selected plots to be stored in the V/F file, during interactive processing, for subsequent offline plot generation.
- b) Include the SAVE FILES(VECTFIL) statement in the Control Program (ref. sec. 228.6), after the EXECUTE GRAPHICS statements, to save the V/F data file.

c) Submit a batch job to restart execution of the GRAPHICS Postprocessor for generation of the P/F file and corresponding plots via one of the offline plot devices. In this case, the statements which must be included in the "restart" ATLAS Control Program are as follows:

```

BEGIN CONTROL PROGRAM
LOAD FILES (VECTFIL)
EXECUTE GRAPHICS (OFFLINE= { SC4020
                             CALCOMP } )
                             GERBER )
END CONTROL PROGRAM
    
```

A further description of the GRAPHICS-execution restart features is presented in section 228.6.

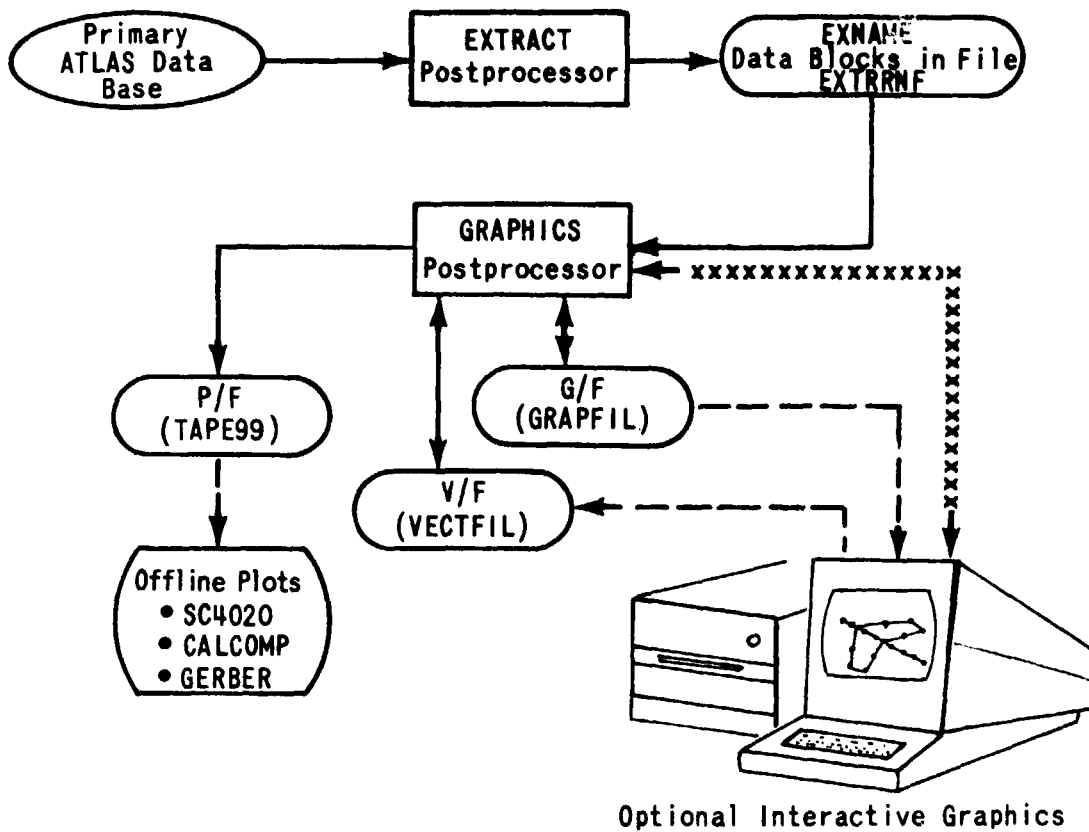


Figure 228-24. Management of Graphics-Related Data

228.4 EXECUTE GRAPHICS STATEMENT

Execution of the GRAPHICS Postprocessor is initiated by the following statement. Previous extraction of the data to be plotted is required (see sec. 218.0).

EXECUTE GRAPHICS (Plist)

GNAME=name An alphanumeric word of 1 to 10 characters that identifies the plot group name for the plots generated by this statement (see sec. 228.3). If the same "name" is specified in multiple GRAPHICS statements in a single job, all corresponding plots are identified by the same group name. Default: Warning. No plots are generated.

OFFLINE= (Mode<,NODISPLAY>)

This parameter defines the execution mode of the GRAPHICS Postprocessor. The options for the key-word "Mode" are:

OFF--Only the G/F file is generated.

ON--The G/F and V/F files are generated.

SC4020	}	-- The G/F, V/F and P/F for the selected offline plotting device are generated. In either the batch execution mode or the interactive execution mode, the P/F file is always created at the end of an ATLAS job even though multiple EXECUTE GRAPHICS statements have been processed. Since the computer operating system supports the creation of a P/F for only one plotting device per job, only one of these key-words should be used during a particular job. If, however, different devices are specified in different commands, only the first-specified device is acknowledged. Offline plots are generated for this device for all plot images written onto V/F during execution of the job.
CALCOMP		
GERBER		

NODISPLAY--This key-word suppresses activation of the interactive-graphics conversational mode when ATLAS is being executed via a terminal. If, for example, execution is being performed via a nongraphics terminal

GRAPHICS

(e.g., a Hazeltine 2000), this key-word should be used to prevent "garbage" output on the terminal.

Defaults: OFFLINE=ON in the batch mode
OFFLINE=OFF in the interactive mode

If ATLAS is being executed via a graphics terminal and the key-word NODISPLAY is not used, the interactive conversational mode is activated automatically for each GRAPHICS statement. At this point, additional directives for controlling which plot images are written onto V/F and P/F can be input by the key-board. Descriptions of the interactive graphics capabilities are presented in section 228.5.

EXNAME= { Name } The alphanumeric name of a data block
 { (List) } identified by a previous EXECUTE EXTRACT
 statement. The information contained in
 this data block is to be used for
 generation of plots. See the following
 discussion regarding the position of this
 parameter within Plist. One or more of
 these parameters may be used in a single
 statement. The Asterisk Name Option (see
 sec. 200.0) may be used to identify data
 block names.
 Default: Warning. No plots are generated.

The following parameters define what types of plots are to be generated for an EXNAME. These parameters remain effective for plotting of subsequent EXNAMES listed in Plist as it is interrogated in a left-to-right order until they are reset. Therefore, the following parameters are specified prior to the first EXNAME parameter in Plist. Except when the EXPLODE option is used for creating undeformed geometry plots, each plot of an EXNAME is displayed separately.

SIZE= (Hor,Ver) This parameter defines the horizontal and
 vertical plot-area dimensions (in inches)
 to be used. If the specified dimensions
 (integers or decimals) exceed the limits
 for the selected plotting device, the
 following maximum plot areas are used.

SC4020	-----	11.0	by	11.0
CALCOMP	-----	88.5	by	29.5
GERBER	-----	144.0	by	36.0
TEKTRONIX 4014	-----	11.0	by	11.0
ONLINE HARDCOPY	-----	8.5	by	8.5

Upon special request from the operator of the GERBER drafting machine, 60-inch width paper is available. In this case, the maximum limits specified for "Hor" and "Ver" are 144.0 and 60.0, respectively.

Default: The area required to satisfy the SCALE parameter described below.

The remaining options for Plist are described in the following sections according to the type of plot to be generated. These options are:

<u>Section</u>	<u>Plot Type</u>
228.4.1 --	Undeformed Geometry Plots
228.4.2 --	Deformed Geometry and Scalar-Grid Plots
228.4.3 --	Contour Plots
228.4.4 --	Graph Plots
228.4.5 --	Matrix Data Plots

The Plist parameters that are applicable to generation of each plot type are summarized in table 228-2.

228.4.1 Undeformed Geometry Plots

The GRAPHICS-statement Plist options are:

GNAME=name OFFLINE= (Mode<,NODISPLAY>) EXNAME= {<Name><(List)>} SIZE= (Hor,Ver)	}	Defined in section 228.1
--	---	--------------------------

TYPE=	{	<u>ORTHOGRAPH</u> (<u>ORTHOGRAPH</u> ,POINT<, <u>NODELTA</u> >) (<u>ORTHOGRAPH</u> ,GRID<, <u>NODELTA</u> ><, <u>NOOFFSET</u> >)	}
-------	---	--	---

This parameter specifies that an orthographic projection of the nodes or finite-element grid is to be created for the next EXNAME in Plist. The optional key-words denote the following:

POINT--Generate a point plot of the nodes

GRID--Generate the finite-element grid

NODELTA--If mid-surface nodes are displayed, plot the mid-surface points without deltas

NOOFFSET--If offset elements are displayed, only plot the structural node connectivity (see sec. 228.1.1).

Default: If POINT or GRID is not specified, a GRID plot is generated for the next EXNAME in Plist. If grid data are not included in EXNAME, no plots are generated. Mid-surface nodes are plotted with deltas (figs. 223-2 and 228-5) and offsets are plotted for offset elements (see sec. 228.1.1).

Defaults: Warning. No plots are generated.

EXPLODE

Key-word denoting that all the views of the EXNAMES specified in Plist are to be displayed on a single plot (fig. 228-6). Each EXNAME is plotted to the same scale. This option allows an "exploded view" of the model to be created. The relative position and orientation of each EXNAME are defined by the TX-TY-TZ, RX-RY-RZ, VX-VY-VZ, and/or VIEW parameters discussed below.

Default: Each view of each EXNAME is displayed on a separate plot.

TX=a
TY=b
TZ=c

The translations (integers or decimals) to be performed on the EXNAME data relative to the GLOBAL X, Y and Z axes, respectively. Translations are performed prior to any rotations of the data. These parameters are effective only when the key-word EXPLODE is used.

Default: The last specified value for the corresponding parameter or zero.

ORIGINAL PAGE IS
OF POOR QUALITY

The orientation of the data to be plotted is defined either by the following RX-RY-RZ parameters, the VX-VY-VZ parameters, or the VIEW parameter.

RX=a
RY=b
RZ=c

The rotations in degrees (integers or decimals) about the R-P-Q axes (the initially-oriented X-Y-Z axes),

respectively, to be performed on the EXNAME data. The sequence in which these parameters appear in Plist defines the order of the sequential rotations. If it is necessary to change the sequence or magnitude of rotation for subsequently listed EXNAMES, each of these parameters should be reset. If less than three rotations are specified, the order of rotations is assumed to be RZ, RY and RX, respectively, and the corresponding rotational values are assumed to be the last specified.

Default: The following sequence and magnitude of rotations are assumed: RZ=125, RY=25, RX=0.

Caution: If the following VX-VY-VZ parameters are used, the RY-RZ values are ignored.

VX=x
VY=y
VZ=z

The GLOBAL X, Y and Z coordinates (integers or decimals) that define a viewing vector.

Default: These parameters are effective only if at least one of them is input. In this case, the remaining coordinates are assumed to be the last specified values or zero.

Examples: The parameter VX=1.0 is equivalent to the three parameters RZ=0.0, RY=0.0, RX=0.0. The parameter VX=-1.0 is equivalent to the three parameters RZ=0.0, RY=180.0, RX=0.0. It should be noted that combinations of ±1.0 for VX, VY and VZ allow isometric views to be generated conveniently.

VIEW=Code

This parameter is used to generate GLOBAL-axis views or multiple views of the next EXNAME in Plist. Code is a 7 digit integer; each digit is set to 0 (no) or 1 (yes) to denote whether the corresponding view is desired. From left-to-right, the digits correspond to -X, +X, -Y, +Y, -Z, +Z and the pictorial view defined either by RX-RY-RZ or by VX-VY-VZ. Leading zeros need not be input. See figure 228-23 for the GLOBAL-axis view orientations.

Default: The view defined by the last specified "Code" or the view defined either by RX-RY-RZ or by VX-VY-VZ is generated.

Example: VIEW=101 denotes that a -Z GLOBAL-axis view plus a pictorial view of the next EXNAME in Plist are to be generated.

LABEL= $\left\{ \begin{array}{l} N \\ E \\ T \\ N+E \\ N+T \end{array} \right\}$

This parameter denotes that integer labels are to be plotted for node (N) numbers, element (E) numbers, or element-type (T) numbers. The combinations N+E and N+T, as shown, can also be used.

Default: The labels identified by the last parameter of this type in Plist are displayed or no labels are displayed.

SCALE=scale

The plot scale (integer or decimal) greater than or equal to zero. When the scale is zero, the size of the plotted data is adjusted so that the figure remains within the maximum plot area. Only one SCALE may be specified if the EXPLODE option is used. In this case, the specified scale must account for the absolute dimensions of the display including the effects of specified translations and/or rotations. If the specified scale requires a plot area larger than the limits for the plotting device, the scale is assumed to be zero. Default: The last specified value or SCALE=0.

Example GRAPHICS Statement:

EXECUTE GRAPHICS (GNAME=PLOTS, TYPE=ORTH, EXNAME=N3,
VIEW=101000, EXNAME=(N5,E100), VX=1.0, VY=1.0, VZ=1.0,
EXNAME=E1*, SCALE=.05, VIEW=1101, EXNAME=E3)

This statement initiates generation of the following views for the specified EXNAMES:

EXNAME	SCALE	-X	+X	-Y	+Y	-Z	+Z	Pictorial
N3	0.0							Default
N5	0.0	✓		✓				None
E100	0.0	✓		✓				None
E10	0.0							User
E11	0.0							User
•	•							•
•	•							•
•	•							•
E19	0.0							User
E3	.05				✓	✓		User

228.4.2 Deformed-Geometry and Scalar-Grid Plots

The GRAPHICS-statement Plist options are:

GNAME=name	}	Defined in section 228.1
OFFLINE= (Mode<,NODISPLAY>)		
EXNAME= {<Name><(List)>}		
SIZE= (Hor,Ver)		

The following parameters are the same as defined in section 228.4.1 except for the NODELTA key-word. In this case, only mid-surface points are displayed for mid-surface nodes.

TYPE=	{	ORTH	}
	{	(ORTH,POINT)	}
	{	(ORTH,GRID<,NOOFFSET>)	}
RX=a,RY=b,RZ=c			
VX=x,VY=y,VZ=z			
VIEW=Code			
LABEL=	{	N	}
	{	E	}
	{	T	}
	{	N+E	}
	{	N+T	}
SCALE=scale			

The following "VECTOR" and VSCALE parameters define the types of deformed geometry plots that are to be created.

{	VECTOR1	}	=	{	DISP	}
{	VECTOR2	}		{	BMODE	}
{	VECTOR3	}		{	VMODE	}
{	VECTOR4	}				

The plot formats identified by the "VECTOR" key-words are:

VECTOR1--Undeformed geometry plus deformed geometry plus vectors

VECTOR2--Undeformed geometry plus deformed geometry

VECTOR3--Undeformed geometry plus vectors

VECTOR4--Deformed geometry only

Displacement plots corresponding to static loading are generated if the DISP key-word is used, whereas mode shape plots are generated if the BMODE (buckling) or VMODE (vibration) key-word is used.

Default: Undeformed geometry plots are generated.

VSCALE=scale A scale factor (integer or decimal) to be applied to the "vector" values prior to generation of plots.
 Default: The last specified value or VSCALE=1.0

Static Displacement Plots

If an ordered node subset is included in the EXNAME for which a grid is to be plotted, only the grid defined by the node subset is displayed. Any of the optional forms of the parameters shown above can be used.

Mode Shape Plots

- a) If a grid is to be displayed for mode shape plots, the EXNAME must include an ordered node subset.
- b) The TYPE parameter must be input as TYPE=ORTH
- c) Only node numbers can be plotted via the LABEL=N parameter.

Scalar-Grid Plots

If scalar-grid plots are to be generated, the following two parameters are used in lieu of the foregoing "VECTOR" and VSCALE parameters. Otherwise, any of the optional forms of the parameters shown above can be used.

SCALAR=Label One of the data component labels selected from table 156-1 that identifies a stiffness-element property, a stress component, or a margin of safety.
 Default: Undeformed geometry plots are generated.

SSCALE=scale A scale factor (integer or decimal) to be applied to the "scalar" values prior to generation of plots.
 Default: SSCALE=1.0

228.4.3 Contour Plots

The GRAPHICS-statement Plist options are:

GNAME=name OFFLINE= (Mode<,NODISPLAY>) EXNAME= {<Name><(List)>} Size= (Hor,Ver)	}	Defined in section 228.1
--	---	--------------------------

TYPE=CONTOUR This parameter denotes that contour plots are to be generated.
Default: Warning. No plots are generated.

RX=a,RY=b,RZ=c VX=x,VY=y,VZ=z VIEW=Code	}	Defined in section 228.4.1
---	---	----------------------------

SCALAR=Label One of the data component labels selected from table 156-1 that identifies a stiffness-element property, a stress component, or a margin-of-safety scalar field.
Default: Warning. No plots are generated.

FLEV=val1 LLEV=val2	}	These parameters (integers or decimals) define the first or smallest value (FLEV) and the last or largest value (LLEV) of the contour levels for the scalar field. Default: Warning. No plots are generated.
------------------------	---	---

INTLEV=value The constant interval between adjacent contour levels for the scalar field.
Default: Warning. No plots are generated.

If an ordered node subset is included in the EXNAME for which a contour plot is to be generated, a triangulated grid is used automatically to interpolate the data points. Otherwise, a rectangular mesh is used for generation of the contour plot and the following parameter is applicable.

NMESH=value The integer number of divisions (≤ 40) in either direction that the rectangular area is to be divided prior to interpolating the data values for generation of isocurves.
Default: NMESH=10

228.4.4 Graph Plots

The GRAPHICS-statement Plist options are:

GNAME=name OFFLINE= (Mode<,NODISPLAY>) EXNAME= {<Name><(List)>} SIZE= (Hor,Ver)	}	Defined in section 228.1
--	---	--------------------------

TYPE=GRAPH This parameter denotes that graph plots are to be generated.
 Default: Warning. No plots are generated.

X-Y Graphs

$X = \left\{ \begin{array}{l} LC \\ CYCLE \\ V \end{array} \right\}$	The abscissa of the graph is identified by one of the key-words:
--	--

LC--The element stress component identified by the "Y1" and/or "Y2" parameters is to be plotted against selected loadcases.

CYCLE--The element margin of safety identified by the "Y1" and/or "Y2" parameters is to be plotted against selected design-cycle numbers.

V--The damping or frequency values identified by the "Y1" and/or "Y2" parameters are to be plotted against velocities for selected flutter solutions.

Default: Warning. No plots are generated.

XMIN=val1 } XMAX=val2 }	The minimum (XMIN) and maximum (XMAX) abscissa values (integers or decimals) to be used to generate the graph. Internal loadcase numbers are input if X=LC is used. Default: Warning. No plots are generated.
----------------------------	--

Y1=Name1 One of the element-stress or margin-of-safety component labels selected from table 156-1 if LC or CYCLE is input for X, respectively. If X=V, "Name" must be either the letter "G" or "F" denoting whether V-g or V-f flutter solution plots are to be generated.
 Default: Warning. No plots are generated.

Y1MIN=val1 }
Y1MAX=val2 }

The minimum (Y1MIN) and maximum (Y1MAX) ordinate values (integers or decimals) to be used to generate the graph.
Default: Warning. No plots are generated.

A second graph with the same abscissa established by X, XMIN and XMAX but with a different ordinate can be requested via input of the following three parameters.

Y2=Name2 }
Y2MIN=val3 }
Y2MAX=val4 }

Same as described for Y1, Y1MIN and Y1MAX.
Default: A second graph is not generated.

Loadability Diagrams

MAC=mac

The mean aerodynamic chord of the wing.
Default: Warning. No plots are generated.

LEMAC=lemac

The GLOBAL X-coordinate of the intersection of the MAC and the wing leading edge.
Default: Warning. No plots are generated.

OEW=wt

The operational empty weight. This represents the weight at which the loading will start.
Default: The weight of the mass and stiffness elements plus the concentrated mass subset specified by the CONMASS parameter.

OEWCG=cg

The GLOBAL X-coordinate of the OEW center of gravity. This represents the c.g. point at which loading will start.
Default: The center of gravity of the mass and stiffness elements plus any concentrated mass subsets specified by the CONMASS parameter.

CONMASS=Num

The number (integer) of a subset of concentrated masses, defined via the mass data, which are to be added to the mass and stiffness elements to establish the OEW. This parameter is applicable only if OEW and OEWCG are not specified.
Default: No concentrated mass subsets are included in the OEW.

CGTOL=tol The OEW center of gravity tolerance (% MAC). This tolerance results in two center of gravity points for the OEW, one at OEWCG+CGTOL and the other at OEWCG-CGTOL.
Default: CGTOL=0.

ZFW=zfw The maximum zero fuel weight. This represents the cutoff weight when loading payload (passengers and cargo).
Default: No limit on the amount of payload loaded.

OEWFAC=fac1 The factor by which the OEW is to be multiplied prior to establishing the starting point for the loading.
Default: OEWFAC=1.0

PASSFAC=fac2 The factor by which the passenger vectors are to be multiplied prior to plotting.
Default: PASSFAC=1.0

CARGOFAC=fac3 The factor by which the cargo vectors are to be multiplied prior to plotting.
Default: CARGOFAC=1.0

FUELFAC=fac4 The factor by which the fuel vectors are to be multiplied prior to plotting.
Default: FUELFAC=1.0

The following optional parameters are used to define the weight center-of-gravity grid (fan grid) that is to be used to plot the loading vectors (see fig. 228-25).

WTINC=inc The weight line-spacing on the grid.
Default: WINC=10000.

FWLNE=wt1 The first (smallest) weight line on the grid.
Default: The OEW rounded down to the nearest multiple of WTINC.

LWLNE=wt2 The last (largest) weight line on the grid.
Default: Twice the OEW or the largest mass condition weight rounded up to the nearest multiple of WTINC.

CGINC=inc The center of gravity line-spacing on the grid (% MAC)

Default: CGINC=1.0

FCGLNE=cg1 The first (forward most) center of gravity line on the grid (% MAC)
Default: FCGLNE=0.0

LCGLNE=cg2 The last (aft most) center of gravity line on the grid (% MAC)
Default: LCGLNE=50.0

DATUM=cgref The % MAC of the grid horizontal reference line (% MAC)
Default: DATUM=25.0

228.4.5 Matrix Data Plots

The GRAPHICS-statement Plist options are:

GNAME=name
OFFLINE= (Mode<,NODISPLAY>)
EXNAME= {<Name><(List)>}
SIZE= (Hor,Ver) } Defined in section 228.1

TYPE=MATRIX This parameter denotes that matrix data plots are to be generated.
Default: Warning. No plots are generated.

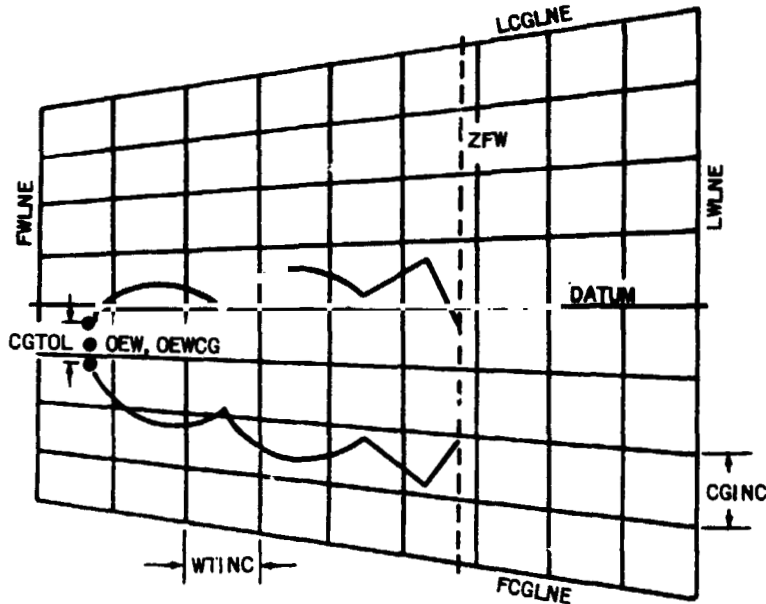


Figure 228-25. Center of Gravity vs Gross Weight Loadability Diagram

Table 228-2. Summary of Execution Parameters for the EXECUTE GRAPHICS Statement

EXECUTE GRAPHICS PLIST PARAMETERS	Type of Plot to be Generated							
	Undeformed Geometry	Static Displacements	Made Shapes	Scalar-Grid	Contour	X-Y Graph	Loadability Diagram	Matrix Data
GNAME = name	✓	✓	✓	✓	✓	✓	✓	✓
OFFLINE = Options	✓	✓	✓	✓	✓	✓	✓	✓
EXNAME = {<Name> <<List>>}	✓	✓	✓	✓	✓	✓	✓	✓
SIZE = (Hor, Ver)	✓	✓	✓	✓	✓	✓	✓	✓
TYPE = Options	✓	✓	✓	✓	✓	✓	✓	✓
TYPE = Options	✓	✓	✓	✓	✓	✓	✓	✓
TYPE = ORTHOGRAPHIC		✓	✓					
TYPE = CONTOUR					✓			
TYPE = GRAPH						✓	✓	
TYPE = MATRIX								✓
EXPLODE	✓							
TX = a, TY = b, TZ = c	✓							
RX = a, RY = b, RZ = c	✓	✓	✓	✓	✓			
VX = x, VY = y, VZ = z	✓	✓	✓	✓	✓			
VIEW = Code	✓	✓	✓	✓	✓			
LABEL = Options	✓	✓	✓	✓				
LABEL = N			✓					
SCALE = scale	✓	✓	✓	✓				
VECTOR = Options		✓	✓					
VSCALE = scale		✓	✓					
SCALAR = Label				✓	✓			
SSCALE = scale				✓				
PLEY = val1, LLEV = val2					✓			
INTLEV = value					✓			
NMESH = value					✓			
X = {<LC> <<CYCLE>> <<v>>}						✓		
XMIN = val1, XMAX = val2						✓		
Y1 = Name1						✓		
Y1MIN = val1, Y1MAX = val2						✓		
Y2 = Name2						✓		
Y2MIN = val3, Y2MAX = val4						✓		
MAC = fac							✓	
LFAC = lfmac							✓	
ZFW = wt							✓	
COMASS = Num							✓	
DEWCG = cg							✓	
CGTOL = tol							✓	
ZFW = zfw							✓	
DEWFAC = fac1							✓	
PASSFAC = fac2							✓	
CARGOFAC = fac3							✓	
TUELFAC = fac4							✓	
WTINC = inc							✓	
LWLINE = wt1, LWLNE = wt2							✓	
CGINC = inc							✓	
LCGLNE = cg1, LCGLNE = cg2							✓	
DATUM = cref							✓	

▷ OFFLINE = ({ OFF
ON
SC4020
CALCOMP
GERBER } <<NODISPLAY>>)

▷ LABEL = { N
E
T
N + E
N + T }

▷ TYPE = { ORTHOGRAPH
(ORTHOGRAPH, POINT <<NODELTA>>)
(ORTHOGRAPH, GRID <<NODELTA>> <<NOOFFSET>>) }

▷ { VECTOR1
VECTOR2
VECTOR3
VECTOR4 } = { DISP
BMODE
VMODE }

▷ TYPE = { ORTHOGRAPH
(ORTHOGRAPH, POINT)
(ORTHOGRAPH, GRID <<NOOFFSET>>) }

228.5 INTERACTIVE GRAPHICS

The GRAPHICS Postprocessor can be executed interactively via a Tektronix Model 4014 graphics terminal. The online graphics facilities provided by the interactive execution mode complement those graphical-display capabilities provided by the EXECUTE GRAPHICS statement (sec. 228.4). Functions that can be performed online include the following:

- a) Display selected plots on the terminal screen
- b) Generate instant hardcopy plots of selected displays shown on the terminal screen. A hardcopy unit that is compatible with the Tektronix 4014 must be used.
- c) Create new displays of selected plots via zooming and rotation.
- d) Enlarge selected regions of online plots by zooming.
- e) Select plots to be generated by one of the offline plotting devices (SC4020, CALCOMP or GERBER)

All plots displayed on the terminal screen are initially defined via the EXECUTE GRAPHICS statement and stored in the G/F file. This passive graphics capability allows selected plots to be viewed online, between execution steps, for verification of the analysis/design process.

The user conducts an interactive dialogue with the GRAPHICS Postprocessor via a function menu, two plot directories and a plot transformation menu. Zooming a selected region of an online plot, for example, can be performed by using either the plot transformation menu or the crosshairs provided by the Tektronix 4014. A description of the interactive dialogue, including the user-command format and the menu/directory functions, is presented in section 228.5.1.

If ATLAS is being executed via a graphics terminal, as discussed in section 228.4, the interactive conversational mode is activated automatically for each EXECUTE GRAPHICS statement in the Control Program. At this point, additional directives for controlling which plot images are written onto the V/F and P/F files can be input by the key-board provided the key-word NODISPLAY is not specified via the OFFLINE parameter. It should be noted that if the parameter OFFLINE=OFF is used in the GRAPHICS statement, the only plots written onto the V/F file are those specifically selected to be written as such. However, if the OFFLINE=ON parameter or the OFFLINE=Device parameter is used, all plots displayed on the terminal screen are written onto the

V/F file unless the appropriate interactive command is input to suppress this activity.

The interactive graphics capabilities can be used to display/manipulate any of the plot types identified by the TYPE parameter in the EXECUTE GRAPHICS statement.

TYPE= {
 ORTHOGRAPH
 CONTOUR
 GRAPH
 MATRIX

228.5.1 Interactive Dialogue

The menus and directories used during interactive-graphics processing are identified as:

- a) Function Menu -- A list of execution functions, as shown in figure 228-26, from which user-selections are made online.
- b) GNAME Directory (G-D) -- The G-D contains a list of the GNAMEs (100 maximum), as defined via the EXECUTE GRAPHICS statements, for which plots have been written onto the G/F file. A maximum of 40 plots can be associated with each GNAME (a plot grouping defined by the user). Figure 228-27 shows a typical G-D. After a GNAME is selected from the G-D, the corresponding Plot ID Directory is displayed.
- c) Plot ID Directory (P-I-D) -- A P-I-D is generated automatically for each GNAME. Each P-I-D contains a list of identifiers, one for each plot associated with the selected GNAME. A typical P-I-D is shown in figure 228-28. The plot that is to be viewed/manipulated online is selected from this directory.
- d) Plot Transformation Menu (P-T-M) -- A list of parameters that can be modified such that the selected plot can be rotated and/or zoomed to create a new display (see fig. 228-29).

The interactive-graphics execution mode is identified by the prompt character

1>

displayed at the beginning of each line on the terminal screen. At this point, a graphics execution directive is entered via the key-board. The formats of the user commands for making

selections from the menus and directories and for modifying the plot transformation parameters are summarized below.

<u>User Command</u>	<u>Applicable Menu</u>
FB,n	Function Menu
LP,n	G-D or P-I-D
Word1=x.x<,Word2=y.y>	Plot Transformation Menu (P-T-M)

The characters "FB" and "LP" are input as shown, in lieu of function buttons and a light pen which are associated with the more expensive, usually refreshable, graphics terminals. The integer that is input for "n" identifies a particular entry in the corresponding menu or directory previously selected via an "FB,n" command. A plot transformation parameter is input as a key-word followed by an equal sign and a decimal number, "x.x" or "y.y," that is to be assigned to the parameter.

Multiple commands can be concatenated by using a semicolon as a command delimiter. In this case, the string is limited to a maximum of 60 characters. Each command or string of commands is terminated by depressing the CR (carriage return) key. Typical command formats are:

```
I>FB,4      CR
I>LP,3;FB,1  CR
I>FB,9;RY=60.,RX=45.,RZ=0.0;FB,1  CR
```

If an "LP,n" command is input when the GNAME Directory or the Plot ID Directory is not displayed on the screen, it is ignored. However, an "FB,n" command can be input, when applicable, even though the Function Menu is not displayed at that time. "FB,n" commands that are applicable when one of the menus has been activated or when a graphical display is on the screen are summarized in table 228-3. Transformations that can be performed on the various plot types, as displayed on the screen, are summarized in table 228-4.

To illustrate the user commands and a typical interactive dialogue, assume the GNAMEs shown by figure 228-27 have been defined by previously executed GRAPHICS statements. Furthermore, assume that the displays for GNAME=GEOMETRY are identified by the P-I-D shown by figure 228-28.

Upon entering the interactive mode, the G-D menu is displayed on the screen (fig. 228-27). A maximum of 50 GNAMEs are displayed at one time. If more than 50 GNAMEs were written onto G/F, the second "page" of the G-D is displayed by depressing the CR key. If the CR key is depressed again, in this case, the first "page" of the G-D is regenerated. To view the plots

associated with the GEOMETRY plot grouping, for this example, the user enters the command

```
I>LP,1;FB,1 CR
```

The selected GNAME is identified by a cross appearing on the screen automatically followed by a display of the corresponding P-I-D (fig. 228-28).

All plots defined under the user-selected GNAME are identified in the P-I-D. This directory consists of a catalog of the plot titles written onto G/F under the selected GNAME. To display one of these plots on the screen, a selection is made by the "LP,n" command. Thus, for example, if the user enters

```
I>LP,4;FB,1 CR
```

the selected plot title is identified by a cross on the screen followed by generation of the online plot.

A plot that has been generated on the screen can be transformed (rotated and/or zoomed) to create a new display of the same data. The function commands FB,8, FB,9 or FB,10 as shown in table 228-4 are used to perform transformations. To display the Plot Transformation Menu, the following command is entered:

```
I>FB,8 CR
```

The values of the active orientation parameters and the X-Y-Z coordinate limits for the plot are shown in the P-T-M as illustrated by figure 228-30. The values for the X-Y-Z coordinate limits depend on what type of plot has been generated. For orthographic plots, these limits are the GLOBAL X-Y-Z coordinates. For contour plots and graphs, however, the X and Y limits are associated with the horizontal and vertical directions (the P-Q axes) of the plot. The Z-limits for a contour plot identify the numerical limits of the scalar field for which isocurves are generated, whereas the Z-limits are ignored for graphs.

To reset the values of the transformation parameters, the user may enter, for example, the following commands:

```
I>VX=-1.0,VZ=1.0,ZMAX=5.0;FB,1 CR
```

Updated values are displayed to the right of the original values, which are crossed out as shown by figure 228-30, and the new display is generated on the screen. For this example, a new viewing direction was selected in addition to zooming in on the

Z-coordinate limits of the plot. It should be noted that the zoom parameters are effective in three dimensions. Thus, the original plot from which a zoomed display is generated can be re-created by resetting the values of the transformation parameters.

Transformation of a plot that has been generated on the screen can be performed without displaying the P-T-M. For example, the commands

```
I>FB,9;VX=-1.0,VZ=1.0,ZMAX=5.0;FB,1 CR
```

could be used instead of the two foregoing lines of commands to effect the same result.

In lieu of specifying new values for the XMIN, XMAX, ..., ZMAX parameters to zoom a display shown on the screen, the user can employ the crosshairs provided by the Tektronix 4014. To activate the crosshairs, the following command is entered:

```
I>FB,10 CR
```

The horizontal and vertical crosshairs that appear on the screen are positioned at one corner of the rectangular area to be zoomed. The following steps are then followed:

- a) Type any numeric or alphabetic character followed by depressing the CR key.
- b) Wait for the crosshairs to reappear.
- c) Position the crosshairs at the diagonal corner of the area to be zoomed.
- d) Repeat (a)

The windowed portion of the display will then be generated such that the zoomed area covers the screen. Figure 228-31 illustrates a display that is zoomed by use of the crosshairs.

All plots that are displayed on the screen are written onto the V/F file if one of the following parameters is used in the EXECUTE GRAPHICS statement.

$$\text{OFFLINE} = \left\{ \begin{array}{l} \text{ON} \\ \text{SC4020} \\ \text{CALCOMP} \\ \text{GERBER} \end{array} \right\}$$

If, however, there are many plots on the G/F file which are to be written onto V/F or if there are displays which consist of

voluminous amounts of data that would require a large amount of clock time to display on the terminal, the function menu command FB,7 can be used. Thus, for example, if the P-I-D is displayed on the screen and the command

```
I>FB,9;LP,2;LP,3;LP,5;LP,7;FB,1  CR
```

is input, plots 2, 3, 5 and 7 would be written onto V/F without online displays.

Alternatively, if one of the foregoing OFFLINE parameters is used and only online displays are to be generated for selected plots, the FB,5 and FB,6 function menu commands can be used.

When the current interactive dialogue is completed, execution control is returned to the Control Program via the following command:

```
I>FB,12  CR
```

The control statement immediately following the last executed GRAPHICS statement is then performed.

The option provided by the FB,13 command allows the user to abort subsequent execution of the current ATLAS job. Generally, this option is used only when uncorrectable anomalies have been detected in the problem data by viewing the graphical displays online.

FUNCTION MENU

<u>KEY</u>	<u>DESCRIPTION</u>
FB,1	EXECUTE CURRENT MENU SELECTION(S)
FB,2	DISPLAY FUNCTION MENU
FB,3	DISPLAY GNAME DIRECTORY (G-D)
FB,4	DISPLAY PLOT ID DIRECTORY (P-I-D) FOR THE GNAME SELECTED FROM G-D
FB,5	DISPLAY REMAINING PLOTS FOR THIS STATEMENT ONLINE ONLY (ENTER FB,5 TWICE)
FB,6	DISPLAY NEXT PLOT ONLINE ONLY
FB,7	WRITE PLOTS SELECTED BY A STRING OF LP,N COMMANDS ONTO V/F WITHOUT ONLINE DISPLAY
FB,8	DISPLAY PLOT TRANSFORMATION MENU (P-T-M) TO ROTATE AND/OR ZOOM
FB,9	INPUT TRANSFORMATION PARAMETERS WITHOUT P-T-M DISPLAY
FB,10	ZOOM CURRENT PLOT VIA CROSSHAIRS
FB,11	DISPLAY DEFORMED GRID OR TRIANGULATED REGION FOR A CONTOUR PLOT
FB,12	RETURN TO ATLAS CONTROL PROGRAM
FB,13	STOP ATLAS EXECUTION (ENTER FB,13 TWICE)

Figure 228-26. Function Menu Options

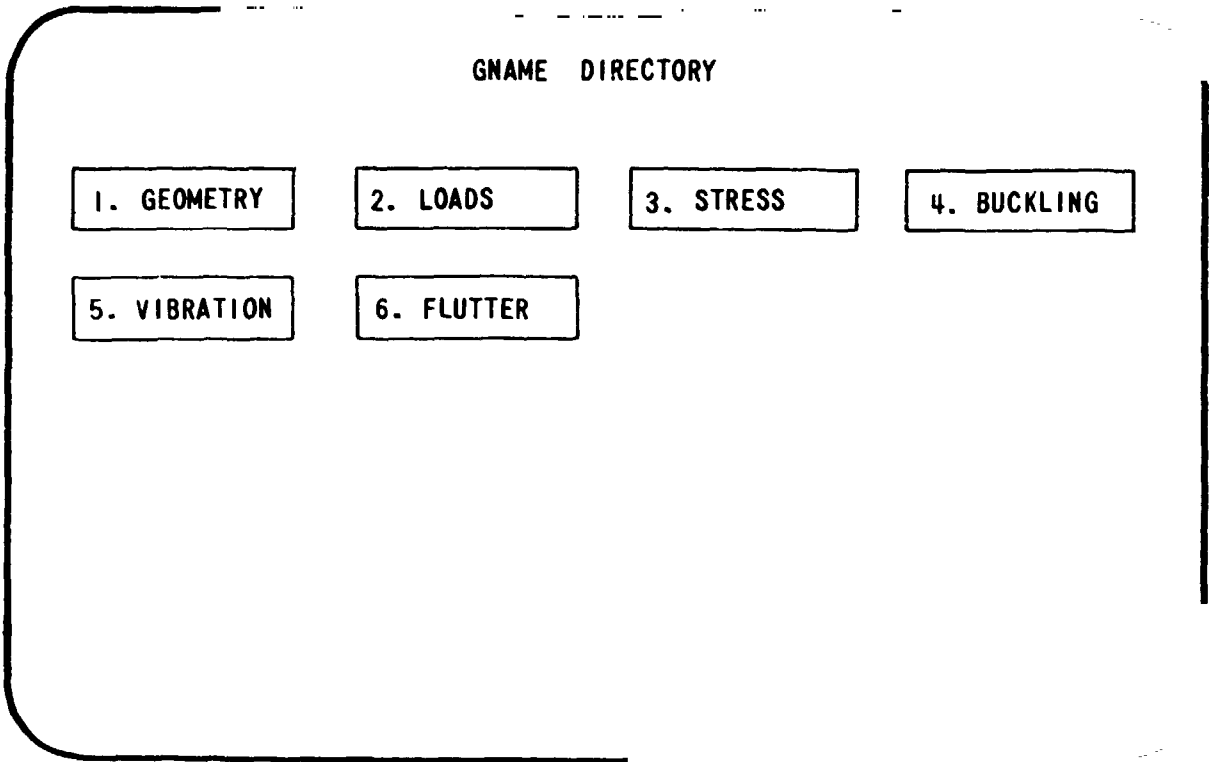


Figure 228-27. Typical GNAME Directory

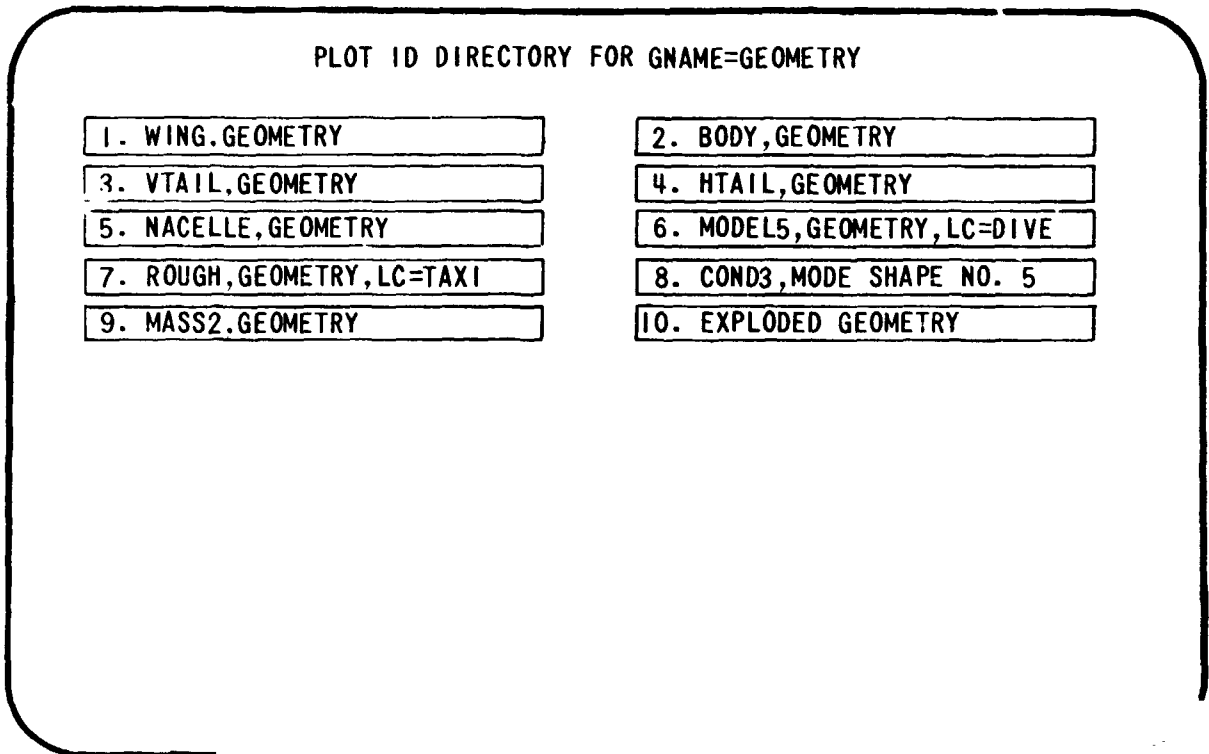


Figure 228-28. Typical Plot ID Directory

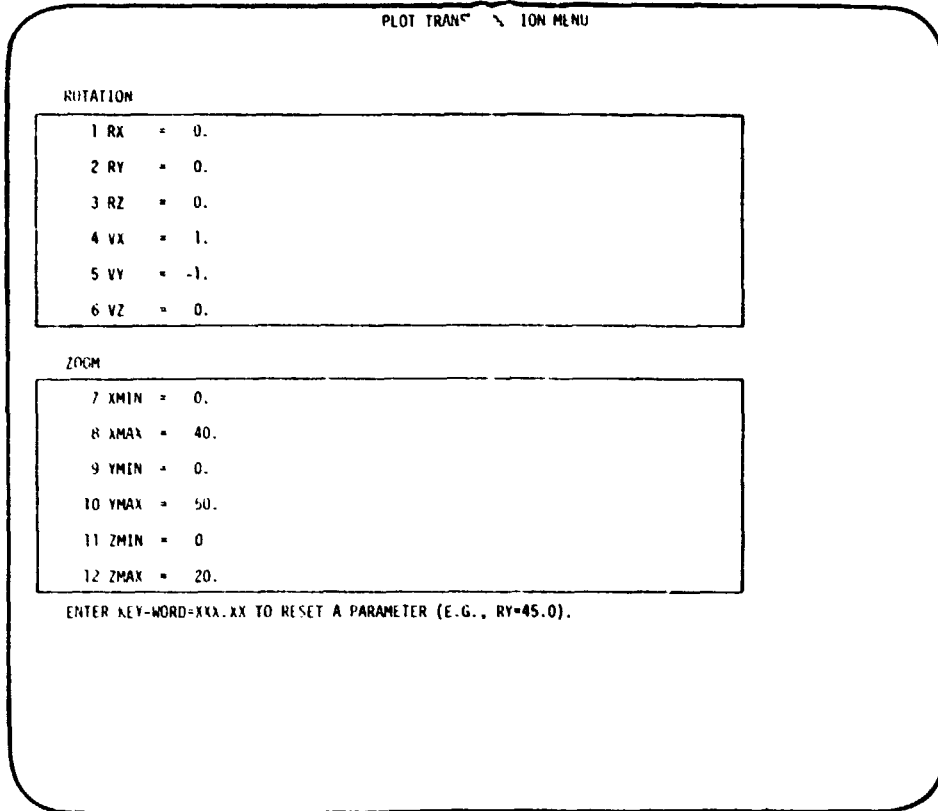


Figure 228-29. Plot Transformation Menu

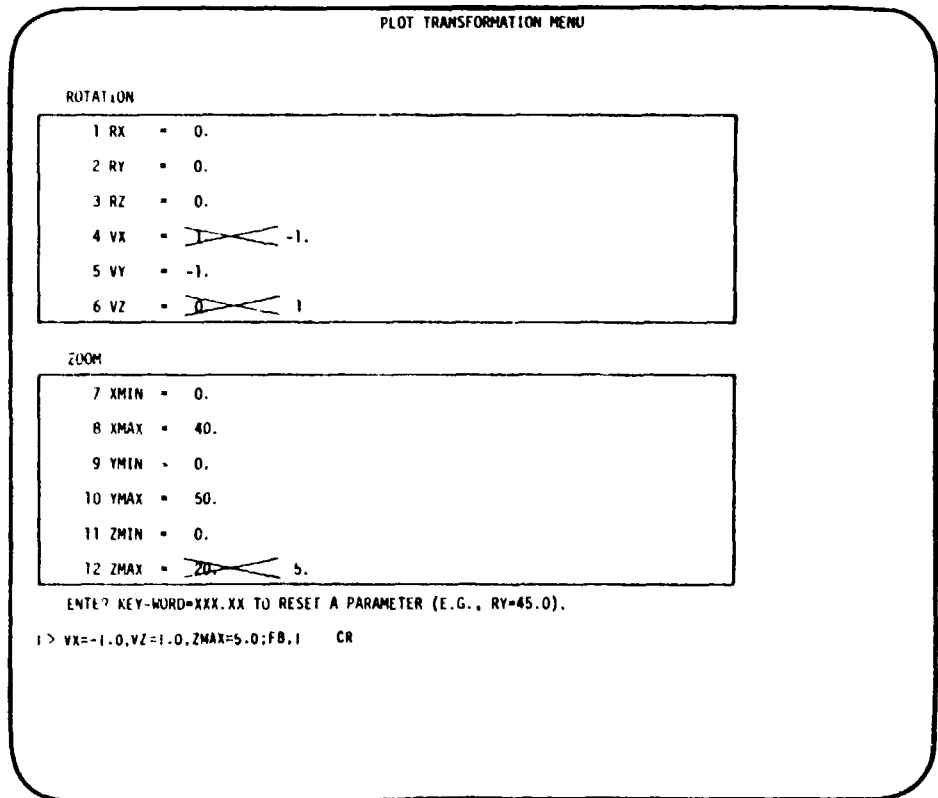


Figure 228-30. Example Use of the Plot Transformation Menu

Table 228-3. Applicable FB,N Commands when a Menu or Graphical Display is Activated

APPLICABLE FUNCTION MENU COMMANDS	ACTIVE MENU OR DISPLAY				
	FUNCTION MENU FB.2	GAME DIRECTORY (G-D) FB.3	PLOT ID DIRECTORY (P-I-D) FB.4	PLOT TRANS- FORMATION MENU (P-T-M) FB.8	GRAPHICAL DISPLAY LP.N
FB,1				✓	
FB,2	✓		✓	✓	✓
FB,3	✓		✓	✓	✓
FB,4				✓	✓
FB,5	✓	✓	✓	✓	✓
FB,6	✓	✓	✓	✓	✓
FB,7			✓		
FB,8					✓
FB,9					✓
FB,10					✓
FB,11					✓
FB,12	✓	✓	✓	✓	✓
FB,13	✓	✓	✓	✓	✓

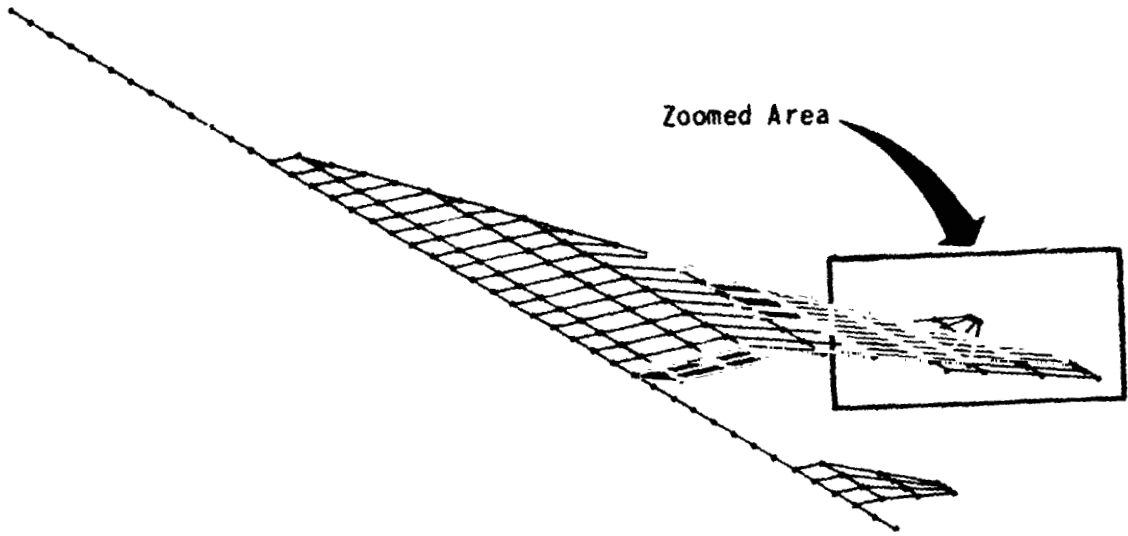
Table 228-4. Plot Transformations Effected via
FB,N Commands and Crosshairs

PLOT TYPE	ROTATION	ZOOM	
		P-T-M Menu	Crosshairs
<u>ORTHOGRAPHIC</u>	FB,8 or FB,9	FB,8 or FB,9	FB,10
<u>CONTOUR</u>		FB,8 or FB,9	FB,10
● <u>DEFORMED RECTANGULAR MESH</u>	FB,8 or FB,9		FB,10
● <u>TRIANGULATED REGION</u>	FB,8 or FB,9		FB,10
<u>GRAPH</u>		FB,8 or FB,9	FB,10
<u>MATRIX</u>			

GRAPHICS

GNAME = GEOMETRY

PLOT ID = GEOM . GEOMETRY

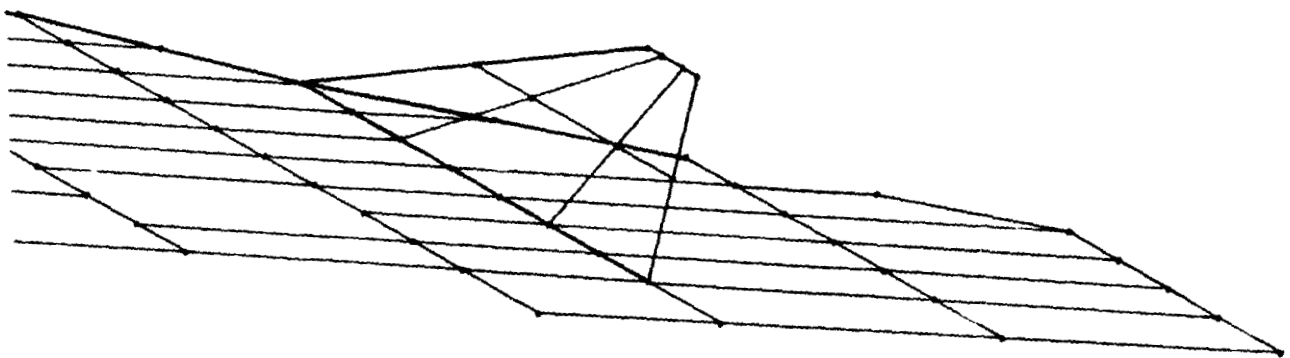


EXNAME = GEOM KSET = 1 NSUB = N001
ESUB = E001 LSUB = KGRID

a) Initial Display

GNAME = GEOMETRY

PLOT ID = GEOM . GEOMETRY



EXNAME = GEOM KSET = 1 NSUB = N001
ESUB = E001 LSUB = KGRID

b) Zoomed Display

Figure 228-31. Zoomed Online Display

228.6 RESTART OF GRAPHICS POSTPROCESSOR EXECUTION

The G/F and/or V/F graphics data files that are generated and saved by one ATLAS job can be loaded and used by subsequent ATLAS jobs. These files, in addition to the other data files generated during a job, can be saved and loaded by using the SAVE FILES and LCAD FILES statements (see sec. 200.3). If only the G/F and/or V/F files are to be saved or loaded, however, the execution control statements described below must be used. As discussed previously, a P/F file is created for generation of offline plots by a particular plotting device (SC4020, CALCOMP or GERBER). This file can not be saved and used by subsequent ATLAS jobs.

228.6.1 Saving Graphics Data

The following statement can be used to save the G/F (GRAPFIL) and/or V/F (VECTFIL) graphics files for subsequent restart of execution.

```
SAVE FILES ( <GRAPFIL> )
             <VECTFIL> )
```

The key-words GRAPFIL and/or VECTFIL can be used with the other SAVE FILES parameters described in section 200.3.

If only the G/F file is to be generated and saved, the OFFLINE=OFF parameter should be used in the EXECUTE GRAPHICS statement (sec. 228.4) and the SAVE FILES or SAVE FILES (GRAPFIL) statement should be used.

The G/F file and the V/F file are generated if one of the following parameters is used in any of the EXECUTE GRAPHICS statements in a Control Program.

```
OFFLINE= { ON
           SC4020
           CALCOMP
           GERBER }
```

In this case, either one or both graphics files can be saved by using one or more of the following control statements.

```
SAVE FILES
SAVE FILES (GRAPFIL)
SAVE FILES (VECTFIL)
SAVE FILES (GRAPFIL,VECTFIL)
```

228.6.2 Loading Graphics Data

The following statement can be used to load the G/F (GRAPFIL) and/or V/F (VECTFIL) graphics files to restart execution of the GRAPHICS Postprocessor.

```
LOAD FILES ( <GRAPFIL> )
            <VECTFIL> )
```

The key-words GRAPFIL and VECTFIL can be used with the other LOAD FILES parameters described in section 200.3.

The respective types of graphics data generated by the current ATLAS job are either added to the previously-saved graphics files or they are written onto new G/F and/or V/F files. The graphical display data generated by the GRAPHICS Postprocessor are dependent on the OFFLINE parameters used as well as the directives used during interactive execution. Examples are presented below to illustrate various types of execution restart of the GRAPHICS Postprocessor.

Create a V/F Corresponding to a G/F

The following two control statements can be used to create a V/F file that corresponds to a previously generated G/F file.

```
LOAD FILES (GRAPFIL)
EXECUTE GRAPHICS (OFFLINE=ON)
```

Note that only the OFFLINE parameter need be included in the EXECUTE GRAPHICS statement. In the batch execution mode, all plots on the G/F file would be written onto the V/F file. In the interactive mode, however, only those displays identified by the appropriate menu options would be written onto the V/F file.

Create a V/F and P/F Corresponding to a G/F

The following two control statements can be used to create a V/F file and a P/F file that correspond to a previously generated G/F file.

```
LOAD FILES (GRAPFIL)
EXECUTE GRAPHICS (OFFLINE= { SC4020
                             CALCOMP }
                  GERBER )
```

In the batch mode, all plots on the G/F file would be written onto the V/F and P/F files. In the interactive mode, however, only those displays identified by the appropriate menu options would be written onto both the V/F and the P/F files.

Create a P/F Corresponding to a V/F

The following two control statements can be used to create a P/F file that corresponds to a previously generated V/F file.

```
LOAD FILES (VECTFIL)
EXECUTE GRAPHICS (OFFLINE= { SC4020
                             CALCOMP } )
                             GERBER )
```

In the batch mode or in the interactive mode, all plots on the V/F file would be written onto the P/F file for the selected offline plotting device. As noted previously, multiple P/F files for different plotting devices can be generated from the same V/F via multiple ATLAS jobs.

Online Display of Plots from a G/F

The following two control statements can be used to select plots from a G/F for display on the terminal screen without creating the V/F and P/F files. Selection of plots is performed via the interactive-graphics menu options.

```
LOAD FILES (GRAPFIL)
EXECUTE GRAPHICS
```

Note that in this case, no parameters need be included in the GRAPHICS statement.

230.0 INTERACT POSTPROCESSOR

Printout of the nodal, boundary condition and loadcase data associated with substructures, as defined by the interact input data (sec. 130.0), may be requested from the INTERACT Postprocessor as described in the next section.

PRECEDING PAGE BLANK NOT FILMED

230.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the substructure interact data (sec. 130.0).

PRINT INPUT (INTERACT, Plist)

The optional parameters for Plist are as follows:

SS=CATlist ATLAS list of substructure numbers (integers) for which interact data are to be printed.
Default: Warning. No printout is generated.

Type Key-word that indicates what type of interact data is to be printed for each of the specified substructures. One or more of the following key-words may be used in a single statement.

NODES--Nodal data

CONNECTIVITY--Interaction nodes

BC--Boundary condition data

RETAINS--A list of the ordered retained freedoms and a list of the interacted nodal freedoms retained via the INTERACT Preprocessor

LOADCASES--Load case labels

Default: Warning. No printout is generated.

The foregoing types of data, except for the list of ordered retained freedoms and LOADCASES, are printed in internal node-number order.

Corresponding user node-numbers associated with the lowest-level substructures that form specified higher-level substructures are also displayed.

INTERFACE Key-word that is applicable only for printing of CONNECTIVITY and RETAINS data. If this key-word is input in addition to one or both of the key-words

INTERACT

CONN and RETA, these types of data are printed only for the interacted nodes common to two or more substructures specified via "SS."

Default: All data identified by the key-words CONN and RETA are printed for each substructure specified via "SS."

232.0 INTERPOLATION PROCESSOR

The primary function of the INTERPOLATION Processor is to transform vibration modes as calculated by the VIBRATION Processor (sec. 258.0) into an interpolation functional-coefficient form. These coefficients may be used by any of the unsteady aerodynamics processors to obtain modal displacements at the appropriate aerodynamic control points. A user-selected interpolation function is established which relates modal displacements calculated for retained kinematic degrees-of-freedom (see sec. 106.0) to displacements of aerodynamic control points.

The INTERPOLATION Processor can also generate interpolation coefficients in the absence of vibration modes by defining unit-displacement modes associated with selected, retained kinematic degrees-of-freedom. The resulting modes, generated by the "AIC option" of this processor, are basically vibration-mode substitutes. Furthermore, mode shapes may be generated by the INTERPOLATION Processor via user-specified polynomial terms.

Interpolation functional coefficients generated by any of the options discussed below may be used subsequently by one or more of the unsteady aerodynamics processors. The following method of analysis is required for these purposes.

- a) The analysis frames of the retained kinematic degrees-of-freedom must be rectangular x-y-z triads oriented such that the x-axis is parallel to the free-flow streamlines and either the y-axis or the z-axis is normal to the corresponding lifting surfaces defined via the aerodynamics data.
- b) Regions associated with one or more retained nodes must be defined via subsets of retained nodes by the SUBSET-DEFINITION Preprocessor (sec. 156.0). If structural vibration modes are essential to the selected interpolation function, corresponding subset mode-matrices must be calculated by the VIBRATION Processor (sec. 258.0).
- c) Interpolation-function coefficient (IFC) matrices are generated by the INTERPOLATION Processor based on the selected interpolation method for the subsets of retained nodes.
- d) An IFC matrix is associated with each aerodynamic surface of the model via the "Modal Data Subset" input

PRECEDING PAGE BLANK NOT FILMED

aerodynamics data. A particular IFC matrix may be associated with one or more aerodynamic surfaces. Reference sections 104.0, 116.0, 136.0 and 150.0.

Five interpolation methods generally referenced as surface spline, motion axis, motion point, polynomial and beam spline are available. Each of these methods is discussed briefly below.

- a) Surface spline (SURFSPLINE)--The input modal displacements must be associated with a subset of retained nodes that do not lie along a straight line but which define a planar or nearly-planar surface. For best results, the nodes should be evenly distributed over the surface such that little extrapolation is required. The input displacements are interpolated to arbitrarily-located points by an interpolation function based on the small-deflection bending-equation of a circular, uniform-thickness, thin plate of infinite radius simply-supported along its periphery (ref. 232-1).
- b) Motion axis (MOTIONAXIS)--The input modal displacements must be associated with a subset of two or more retained nodes that lie in a plane. A cubic curve is passed through these retained nodes to define a motion axis (e.g., the elastic axis of a high-aspect-ratio wing or the hinge line of an aerodynamic control surface). For best results, the displacement nodes should lie on or close to the axis. A motion axis may be defined such that it is (1) parallel to the x-axis, (2) strictly increasing in the y-direction within the x-y plane or (3) strictly increasing in the z-direction within the x-z plane. Displacements at intermediate points on the motion axis are calculated by using a cubic spline based on the input displacements. Displacements at points not on the defined axis are calculated by linear extrapolation of the axis displacements. Input modal components may include z-axis translation and rotations about the x-axis and y-axis or they may include y-axis translation and rotations about the x-axis and z-axis. To illustrate the MOTIONAXIS interpolation method, consider the high-aspect-ratio wing illustrated in figure 232-1, where it is assumed that the surface lies in the x-y analysis frame.

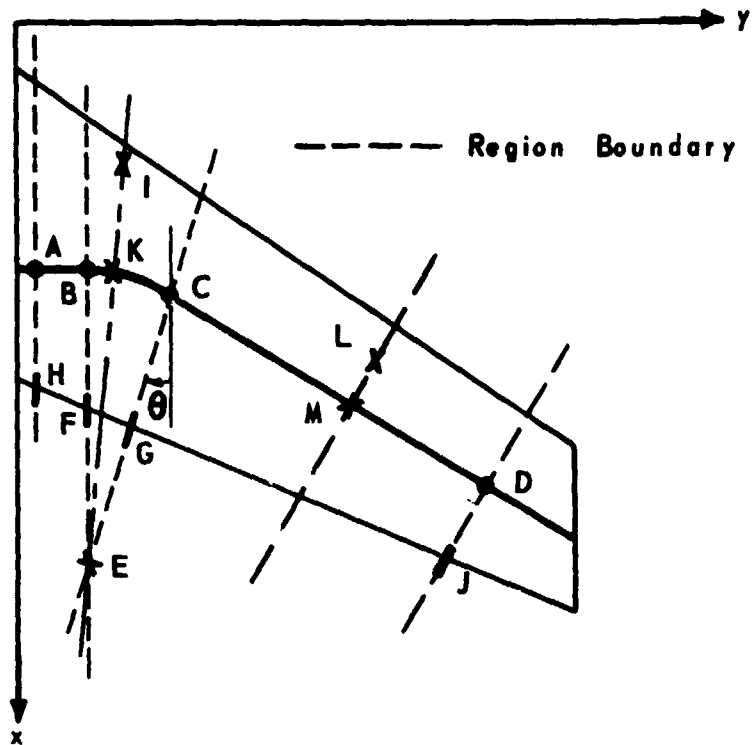


Figure 232-1. Motion-Axis Definition

The motion axis is defined by the retained nodes A, B, C and D. An interpolation region is associated with each segment of the axis. Each region boundary is defined by a plane which is perpendicular to the x-y plane and which intersects the x-z plane at a user-defined angle (e.g., θ as shown in figure 232-1). Displacements at an arbitrary output point on the planform, such as point I in the figure, are calculated in the following manner:

- 1) the output point I is associated with a particular region (axis segment)--region BC. If I falls within more than one region, for the case when the axis is strictly increasing in the y-direction, I is associated with the region defined by the smallest y-coordinate;
- 2) the intersection point of the boundaries of the particular region is determined--point E;
- 3) point I is connected to E by a straight line. The plane which is perpendicular to the x-y plane and which contains line I-E intersects the B-C segment at point K;
- 4) the three displacement components at K are calculated via a cubic spline over the axis-segment B-C;

- 5) the displacements at K are linearly extrapolated to point I.

If the region boundaries associated with the output point are parallel, the output point is projected onto the axis, parallel to the boundary planes. For this case, the foregoing steps (2) and (3) are omitted. Thus, point L in figure 232-1, is projected onto the axis to define point M. Steps (4) and (5) are then used to calculate the displacements at point L.

Displacements at an output point outside of a region boundary are calculated by extrapolating the axis displacements associated with the closest region. These displacements are then linearly extrapolated, parallel to the closest region boundary, to the output point.

In general, it is undesirable to define region boundaries that intersect at a point within the region of output points. This may, for example, result in streamwise discontinuities in the interpolated vibration modes used by the aerodynamics processors.

- c) Motion point (MOTIONPT)--The input modal displacements are associated with the degrees-of-freedom of a single retained node included in a node subset. These displacements are extrapolated linearly to define the motion of other selected points. The interpolation function is a rigid-body motion transformation.
- d) Polynomial (POLYNOMIAL)--The interpolation function is a polynomial of degree 0 to 5 in x and y with coefficients specified by the user. Each polynomial defines a mode (generalized coordinate).
- e) Beam spline (BEAMSPLINE)--The input modal displacements must be associated with a subset of retained nodes that lie in a planar or nearly-planar surface. Subsets of these retained nodes are used to define a lattice of beams comprised of two or more cubic curves. For best results, the beams should be straight or nearly-straight and they should be evenly distributed over the surface. Overlapped beams should be avoided. Each beam is generated such that it is either (1) strictly increasing in the y-direction within the x-y plane or (2) strictly increasing in the z-direction within the x-z plane. Only the displacements at the nodes used to define the beams are used in the interpolation process. Displacements at intermediate

INTERPOLATION

points on a beam are calculated by using a cubic spline based on distance along the beam. Displacements at points not on a defined beam are calculated by linear extrapolation of the curve displacements. Input modal components may include z-axis translation and optionally the rotations about the x-axis and y-axis or they may include y-axis translation and optionally the rotations about the x-axis and z-axis.

232.1 EXECUTE INTERPOLATION STATEMENT

The following statement initiates execution of the INTERPOLATION Processor. Previous execution of the VIBRATION Processor is required for generation of subset mode-matrices if interpolation function coefficients are to be generated from vibration mode shapes via the SURFSPLINE, MOTIONAXIS, MOTIONPT or BEAMSPLINE interpolation function.

EXECUTE INTERPOLATION (Plist)

The order in which the parameters are presented below is the same order in which they must be specified within Plist.

VSET=Vs The vibration set number (integer) identifying the structural model being analyzed (see EXECUTE VIBRATION statement; sec. 258.0). This parameter is not applicable when the AIC or the POLYNOMIAL options are used.
Default: VSET=1

Coefficients associated with a selected interpolation function, as calculated by the INTERPOLATION Processor, are based on either a) the natural, coupled, vibration mode-shapes (generalized displacements) or b) an identity matrix associated with unit-displacement generalized coordinates selected via the AIC key-word presented below. Both types of generalized displacements are associated with selected, retained kinematic freedoms. Subsets of retained nodes are defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0).

AIC<=Nxxx> This parameter denotes that the unit-displacement generalized coordinates associated with the subset of retained kinematic freedoms Nxxx are to be used to calculate the interpolation functional coefficients.
Default: If Nxxx is not specified by this parameter, the array of freedoms to be used to calculate the interpolation coefficients is defined by the following "Nxxx" parameter. Use of the following "Nxxx" parameter permits several sets of interpolation coefficients to be used to build an array of aerodynamic influence

coefficients via the aerodynamic processors.

Default: Vibration mode shapes generated previously by the VIBRATION Processor for the specified VSET are used.

{ Nxxx=Type
 Nxxx=(Type,Name) }

Nxxx is the name of a node subset defined previously via the SUBSET-DEFINITION Preprocessor unless the POLYNOMIAL option is used. In this case, the 1 to 3 digits xxx are used only to form the name of the calculated interpolation-coefficient matrix (see "Name" below). A subset mode-matrix must have been previously extracted by the VIBRATION Processor for the retained nodes associated with Nxxx unless the AIC and/or POLYNOMIAL options are used.

Type--A key-word or equivalent integer which identifies the type of interpolation function:

<u>Key-Word</u>	<u>Integer-Equivalent</u>
SURFSPLINE	1
MOTIONAXIS	2
POLYNOMIAL	3
MOTIONPT	4
BEAMSPLINE	5

Caution: Only one type of interpolation function can be used with the AIC option.

Name--A 1 to 7-character alphanumeric name to be assigned to the interpolation-coefficient User Matrix (see sec. 200.0). If "Name" is specified, the indicated pair of parentheses must be punched. The selected name is used as an input item within the "Modal Data Subset" associated with each of the unsteady aerodynamics data sets.
Default: The interpolation-coefficient matrix is assigned the name Cxxx where xxx is the corresponding node subset

number, zero-filled and right-adjusted. Thus, C010 is generated for N10 and C100 is generated for N100.

One of the input forms of this parameter, Nxxx=Type or Nxxx=(Type,Name), must be specified. Both forms may be used in a single statement. Immediately following each "Nxxx" parameter, a set of additional parameters associated with the selected type of interpolation function must be specified as discussed below.

The "Type" need not be specified explicitly for the second or subsequent parameters of this type in a statement. Thus, parameters of the form Nxxx=(, Name) or just Nxxx denote that the interpolation function type for this Nxxx is the same as the last specified "Type."

Default: Error. Execution is terminated.

The following parameter is required after the "Nxxx" parameter if the SURFSPLINE, MOTIONAXIS, MOTIONPT or BEAMSPLINE interpolation function is selected.

DOF=xyzxyz

The code xyzxyz is a 1 to 6 digit integer denoting which retained freedoms associated with subset "Nxxx" are to be expressed via the selected interpolation function. Each digit in this code is set to 0 or 1 denoting whether the x-axis translation, y-axis translation, z-axis translation, x-axis rotation, y-axis rotation and z-axis rotation, respectively, are to be included. A one indicates the corresponding degree-of-freedom is to be included, whereas a zero indicates the freedom is not to be included. Leftmost zero-value digits of this code need not be input.

The number and type of freedoms which may be denoted by DOF depends on the selected type of interpolation function.

INTERPOLATION

In all cases, at least one freedom must be selected. One to 6 freedoms may be selected for processing by the MOTIONPT function, whereas only 1 of the 6 freedoms may be selected for processing by the SURFSPLINE function. One, 2 or 3 freedoms may be expressed by the MOTIONAXIS or BEAMSPLINE functions via one of the following combinations: a) the y-axis translation or the z-axis translation, b) the y-axis translation and one or both of the x and z-axis rotations, c) the z-axis translation and one or both of the x and y-axis rotations. The MOTIONAXIS function may also be used with any one of the three rotation freedoms. An x-axis translation may not be selected for processing by the MOTIONAXIS or BEAMSPLINE function.

Default: If the DOF parameter is not input immediately after an "Nxxx" parameter which specifies either the SURFSPLINE, MOTIONAXIS, MOTIONPT or BEAMSPLINE function, execution is terminated.

The following two parameters (DEFNPTS and ANGLES) are required after the DOF parameter only if the MOTIONAXIS function is selected.

DEFNPTS= (Node1, Node2, ..., Noden)

A list of 2 to 26 user node numbers which define a motion axis. These nodes must all have been assigned the same x-y-z rectangular analysis frame; they must have been retained and they must be included in subset "Nxxx." These nodes must all lie either in the x-y plane or in the x-z plane. The motion axis is generated by passing a cubic spline through these nodes. The nodes are connected in the order of strictly increasing y-coordinates or strictly increasing z-coordinates. If more than one node in the list has the same y-coordinate, execution is terminated

unless all the nodes have the same y-coordinate.
Default: Error. Execution is terminated.

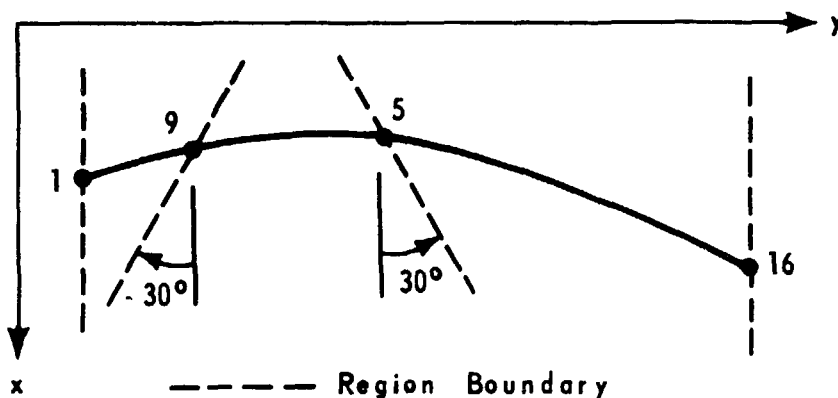
ANGLES=(a1, a2, ..., an)

A list of 2 to 13 angles in degrees measured according to the right-hand rule relative to the axis normal to the surface. These angles define the motion-axis region boundaries at each of the axis-definition points specified via DEFNPTS. The sequence and number of items in this list must correspond with the sequence and number of items in the DEFNPTS list.

Default: If the motion axis is strictly increasing in the y-direction, a boundary that is perpendicular to the axis is defined for each curve segment at the segment definition point with the greatest y coordinate. The boundary through the node with the smallest y coordinate is parallel to the x-axis. If the axis lies in the x-z plane, z coordinates are used similarly to the y coordinates in the foregoing. If the motion axis is parallel to the x-axis, boundaries are defined through the definition points such that they are perpendicular to the axis.

Caution: If a region boundary is located so that it does not cross the motion axis at the definition point, execution is terminated.

Example: DEFNPTS=(1,9,5,16),
ANGLES=(0.,-30.,30.,0.) (see the following sketch)



INTERPOLATION

If the POLYNOMIAL option is selected and the AIC key-word is not input, the following sequence of parameters is required after the "Nxxx" parameter.

MODE, Aij=c,... The key-word MODE denotes that the polynomial terms follow. Each polynomial (mode) is comprised of terms of the form $cx^i y^j$. The powers and coefficient of each term are specified by a parameter of the form "Aij=c" where $0 \leq i$ and $j \leq 5$ such that $(i+j) \leq 5$. This parameter "Aij=c" may be repeated to specify one complete polynomial. The sequence of parameters "MODE, Aij=c,..." may be repeated to define a maximum of 21 polynomials. The user is cautioned, however, that Plist may contain a maximum of 28 parameters. Default: Error. Execution is terminated.

Example: If the sequence of parameters

```
N4=3,  
MODE,A00=1.,A10=2.,A01=3.,A23=-4.,  
MODE,A50=5.,A41=6.,A13=7.
```

is specified, two polynomial modes, expressed as "m1" and "m2" below, are generated.

```
m1=1.0 + 2.0x + 3.0y - 4.0x2y3  
m2=7.0xy3 + 6.0x4y + 5.0y5
```

If the AIC key-word is input and the POLYNOMIAL function is selected, 21 polynomials (modes) are generated automatically from the terms of a general fifth-degree polynomial. The coefficient of each term of each polynomial is one.

The following parameter (BEAMxx) is required after the DOF parameter only if the BEAMSPLINE function is selected.

```
BEAMxx= { (Nxxx,Nyyy,...) }  
        { Nxxx TO Nyyy }
```

Each beam to be used in calculating the BEAMSPLINE interpolation coefficients is described via a subset of retained stiffness nodes defined previously via the SUBSET-DEFINITION Preprocessor (sec. 156.0). A beam is generated for each specified

node subset by passing a cubic spline through the nodes in the order of strictly increasing y or z coordinates. Subsets may be identified by any combination of the two options. The Asterisk Name Option (see sec. 200.0) may be used to identify subsets. The characters "xx" in BEAMxx are used to specify how a beam may be used in extrapolating displacements. "xx" is either blank or is comprised of 1 or 2 letters as follows:

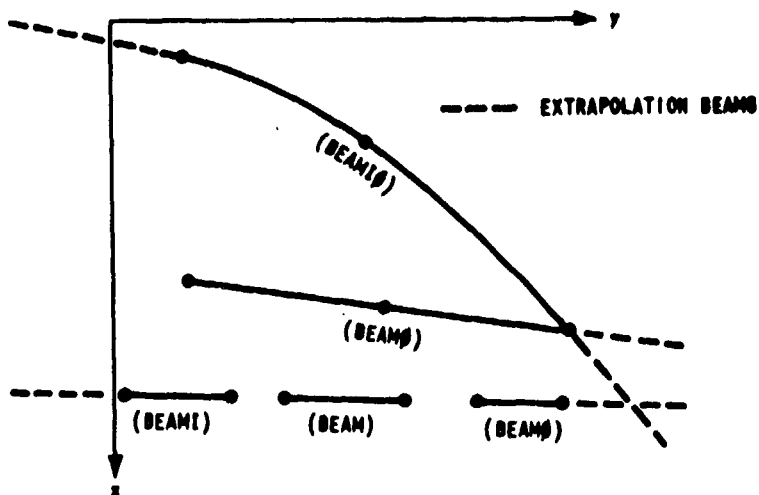
blank--The beam may not be used for extrapolation
 I--Only inboard extrapolation allowed
 O--Only outboard extrapolation allowed
 IO--Inboard and outboard extrapolation allowed

This parameter is repeated to define all beams required for interpolation. The user is cautioned that Plist may contain a maximum of 28 parameters. All of the nodes included in the specified subsets must be contained in the node subset identified by the "Nxxx=Type" parameter. Any modal displacements not associated with the beams are ignored.

Default: Error. Execution is terminated.

Example: The following parameters define the beams illustrated in the sketch below.

BEAM=N11, BEAMI=N21, BEAMO=(N31,N32),
 BEAMIO=N41, ...



234.0 LOADS MODULES

All loads input to an ATLAS structural model must be prepared by the LOADS Processor prior to assembling the loads and specified displacement matrices via the MERGE Processor (sec. 242.0). The LOADS Processor (sec. 234.2) performs the following functions:

- a) Processes the nodal loads as specified by the loads data (sec. 134.0).
- b) Processes the nodal displacements as specified for supported nodal freedoms via the boundary condition data (sec. 106.0) and/or the loads data.
- c) Generates equivalent nodal loads for element-distributed loadings as defined by the loads data. Consistent nodal loads are calculated for BRICK elements, whereas statically equivalent loads are calculated for all other element types (ref. 234-1).
- d) Calculates nodal loads and initial stresses for the nodal and element thermal loads specified by the loads data.
- e) Calculates equivalent nodal loads for inertial forces caused by rigid-body rotations as defined by the loads data.

Groups of input loads are identified by a user-selected load case label. Multiple groups of loads identified with the same label are processed as a single load case. All input loads associated with a load case are cumulative. The user has the option via the EXECUTE LOADS statement to retain selected types of loads for loads and stress processing.

Printout of the input nodal and element-distributed forces, specified temperature increments and specified rotational kinematics may be requested via the statement discussed in section 234.1. Printout of the cumulated applied nodal loads and total applied-load resultants may be requested via the statement discussed in section 234.3.

PRECEDING PAGE BLANK NOT FILMED

234.1 PRINT INPUT STATEMENT

The following statement is used to request printout of selected load case data (sec. 134.0). Structural models for which loads data are to be printed are identified by either a stiffness data-set number and boundary-condition stage number or a lowest-level substructure number. Only one of the identifier types (SET/STAGE or SS described below) may be used in a single statement.

PRINT INPUT (LOADS, Plist)

List of optional parameters for Plist:

SET=Se	The stiffness data set number (integer) for which loads data are to be printed. Default: SET=1
STAGE=St	A boundary condition stage number (integer) associated with set "Se." Default: STAGE=1
SS=CATlist	ATLAS list of lowest-level substructure numbers (integers) for which loads data are to be printed. Default: A SET/STAGE model.
LC=CATlist	ATLAS list of load case identifiers (integers or alphanumeric words) or the key-word ALL denoting which loads data are to be printed. Default: LC=ALL; all load cases associated with the specified SET/STAGE or SS model.
Loadtype	Key-word denoting that a certain type of input loads is to be printed. Several of these key-words may be used in a single statement. The available options are: <u>E</u> LEMENT --Element-distributed loads <u>N</u> ODTHERM --Nodal thermal loads <u>E</u> L THERM --Element thermal loads <u>I</u> NERTIA --inertia loads Default: All load types associated with the specified load cases.

LINES=lines Maximum number of lines (integer≤56)
including page heading to be printed
on a page.
Default: LINES=56

The following parameters are applicable for printing thermal loads data associated with node and element subsets previously-defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0). The Asterisk Name Option (sec. 200.0) may be used to identify subset names. Several of the following parameters may be used in a single statement.

Nxxx Name of a node subset for which nodal
thermal loads are to be printed.
Default: Nodal thermal loads
associated with all the
nodes of the specified
SET/STAGE or SS are printed.

Exxx Name of an element subset for which
element thermal loads are to be
printed.
Default: Element thermal loads
associated with all the
elements of the specified
SET/STAGE or SS are printed.

234.2 EXECUTE LOADS STATEMENT

The following statement initiates execution of the LOADS Processor. If thermal loads are to be processed, the STIFFNESS Processor (sec. 252.0) must be executed prior to execution of this processor. If inertia loads are to be processed, execution of the MASS Processor with OPTION=4 (sec. 238.2) must be performed prior to execution of this processor. Structural models for which loads are to be processed are identified by either a stiffness data-set number and boundary-condition stage number or a lowest-level substructure number. Only one of the identifier types (SET/STAGE or SS described below) may be used in a single statement.

EXECUTE LOADS (Plist)

List of optional parameters for Plist:

SET=CATlist	ATLAS list of stiffness data set numbers (integers) for which loads data are to be processed. Default: SET=1
STAGE=CATlist	ATLAS list of boundary condition stage numbers (integers) associated with the stiffness data sets for which loads data are to be processed. Default: STAGE=1
SS=CATlist	ATLAS list of lowest-level substructure numbers (integers) for which loads data are to be processed. Default: A SET/STAGE model.
LC=CATlist	ATLAS list of load case identifiers (integers or alphanumeric words) or the key-word ALL denoting which loads data are to be processed. Default: LC=ALL; all load cases associated with the specified SET/STAGE or SS models.
Loadtype	Key-word denoting that a certain type of input load is to be processed. Several of these key-words may be used in a single statement to include multiple types of loads for the specified load cases. The available options are:

NODE --Nodal loads
 DISP --Specified nodal
 displacements
ELEMENT --Element-distributed loads
NODTHERM --Nodal thermal loads
ELTHERM --Element thermal loads
INERTIA --Inertia loads

Default: All load types associated with the specified load cases.

MATERIAL= {CONSTANT }
 {VARIABLE }

This parameter is used only for calculation of equivalent nodal loads and initial stresses for thermal loads on BRICK elements. If the parameter MATERIAL=CCONSTANT is used, the elastic properties are based on the element base temperature (the element temperature defined via the stiffness data--sec. 152.0). If the parameter MATERIAL=VARIABLE is used, the elastic properties are based on the final nodal temperature (the sum of the element base temperature and the incremental nodal temperatures defined via the loads data--sec. 134.0). Initial stresses are based on the final nodal temperature.

Default: MATERIAL=CONSTANT

LUMP=factor

The fraction ($0.0 \leq \text{factor} \leq 1.0$) of SPAR-element lumped flange areas as specified via the stiffness input data (see Appendix B) to be used in calculating SPAR-element initial stresses and equivalent nodal forces. This parameter is used only if SPAR element thermal loads are to be processed. If the lumping factors specified via the input data are to be effective, the parameter LUMP=1.0 must be included in Plist.

Default: The value assigned to the parameter LUMP in the Plist of the EXECUTE STIFFNESS statement for the specified structural model.

234.3 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the cumulated nodal loads and total applied-load resultants for selected load cases. In general, the LOADS and MERGE Processors must be executed prior to use of this statement. If only the resultants of applied mechanical loads are to be printed, only the LOADS Processor need be executed prior to using this statement. Structural models for which data are to be printed are identified by either a stiffness data-set number and boundary-condition stage number or a lowest-level substructure number. Only one of the identifier types (SET/STAGE or SS described below) may be used in a single statement. The printed data may also be associated with node subsets defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0).

PRINT OUTPUT (LOADS, Plist)

List of optional parameters for Plist:

- | | |
|--|--|
| SET=Se | The stiffness data set number (integer) for which applied nodal loads are to be printed.
Default: SET=1 |
| STAGF=St | A boundary condition stage number (integer) associated with set "Se".
Default: STAGE=1 |
| SS=CATlist | ATLAS list of lowest-level substructure numbers (integers) for which loads data are to be processed.
Default: A SET/STAGE model. |
| LC=CATlist | ATLAS list of load case identifiers (integers or alphanumeric words) to be processed.
Default: All load cases associated with the specified SET/STAGE or SS model. |
| \langle L1=Name1
L2=Name2
L3=Name3 \rangle | Alphanumeric names of the load-matrix partitions (User Matrices) generated by the MERGE Processor for the specified SET/STAGE or SS. These matrices are associated with the three partition rows (FREE, RETAIN and SUPPORT kinematic freedoms) of the gross stiffness matrix (see sec. 106.1). Several of these parameters may be used |

in a single statement. Zero matrix partitions need not be specified. Nodal loads associated with the FREE and SUPPORT partitions are printed in an ascending internal node number order, whereas nodal loads associated with the RETAIN partition are printed in the order of specified retained nodal freedoms.

If the standard ATLAS control procedures are used (ref. appendix E and sec. 200.0), the names L11, L21 and L31 are assigned to Name1, Name2 and Name3 for SET/STAGE structural models, and the names L11Sxxx, L21Sxxx and L31Sxxx are assigned for lowest-level substructures. The characters xxx denote the substructure number, right-adjusted and zero-filled.

Defaults: L1=L11, L2=L21, L3=L31 for SET/STAGE models;
L1=L11Sxxx, L2=L21Sxxx, L3=L31Sxxx for substructure models.

FORMAT=Code

This parameter defines the format of each printed value. The options available for the integer "Code" are:

- 0--Optimum format. Data values are printed as either F-format or E-format decimal numbers. This option allows the maximum number of digits per value to be printed in the field width specified by the parameter FIELDW=width.
- 1--F-format. Data values are printed as decimal numbers with decimal points aligned in the columns of printed data.
- 2--I-format. Data values are printed as integer numbers.
- 3--E-format. Data values are printed as decimal numbers with exponents.

Default: FORMAT=0

FIELDW=width Number of digits (integer ≤ 12) available for each printed value. The letter E, decimal points and signs each require one digit space.
Default: FIELDW=8

LINES=lines Maximum number of lines (integer ≤ 56) including page headings to be printed on a page.
Default: LINES=56

RESULTANTS Key-word denoting that only the resultants of all loads applied to the structural model are to be printed. Resultants are calculated relative to the origin of the GLOBAL reference frame.
Default: Load resultants and the specified load-matrix data are printed.

Nxxx Node subset name defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0) for which cumulated nodal loads are to be printed. The Asterisk Name Option (sec. 200.0) may be used to identify subset names. Several parameters of this type may be used in a single statement.
Default: Loads are printed for all nodes in the SET/STAGE or SS model that have freedoms associated with the specif. load-matrix partitions.

236.0 MACHBOX AERODYNAMICS MODULES

The MACHBOX Processor calculates unsteady aerodynamic loads on rigid or elastic, single planforms or non-coplanar wings or wing/tail configurations in supersonic flow. Wing/tail configurations with or without vertical separation, longitudinal separation and dihedral on either surface may be examined. A non-intersecting wing/vertical-tail combination may also be investigated. The analysis technique is a three-dimensional Mach Box method with a refinement option to correct pressure distributions by airfoil thickness-slope factors. Irregular fluctuations in the calculated pressure distributions may optionally be reduced via a box-grid subdivision technique and/or a velocity potential smoothing technique. Development of the theoretical basis is presented in reference 136-1.

Generalized airforce matrices required for flutter analyses via the ADDINT and FLUTTER Modules (secs. 202.0 and 222.0) are generated by execution of the MACHBOX Processor (sec. 236.2). These data are calculated by a modal solution based on the aerodynamic model and structural modes defined by the user (sec. 136.0). If the MACHBOX Processor is executed in conjunction with the AIC option of the INTERPOLATION Processor (sec. 232.0), the generalized airforce matrices will represent Aerodynamic Influence Coefficient (AIC) matrices. Other data calculated by this processor include aerodynamic pressure distributions, velocity potentials, box lifts and AGARD (Advisory Group for Aeronautical Research and Development) generalized aerodynamic coefficients. Perturbation velocity components in the flow field (off-planform wash samples) of a single surface may also be generated.

Printout of the MACHBOX input data and data calculated by the MACHBOX Processor may be requested from the MACHBOX Postprocessor as described in sections 236.1 and 236.3, respectively.

PRECEDING PAGE BLANK NOT FILMED

236.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the aerodynamic input data associated with a particular MACHBOX data case (sec. 136.0).

PRINT INPUT (MACHBOX, Plist)

The optional parameter for Plist is:

CASE = Ca The aerodynamic data case number
 (integer) for which input data are to
 be printed.
 Default: CASE=1

236.2 EXECUTE MACHBOX STATEMENT

The following statement initiates execution of the MACHBOX Processor. Previous execution of the INTERPOLATION Processor is required when structural mode shapes generated via the VIBRATION Processor are used in the aerodynamic analysis or Aerodynamic Influence Coefficients are to be generated (see sec. 136.0).

EXECUTE MACHBOX (Plist)

List of optional parameters for Plist:

- COND=co A condition number (integer) in the range 1 to 36 which identifies the remaining parameters of this statement. If multiple executions are required in a single job, the condition number specified by each statement of this type must be unique.
Default: COND=1
- CASE=Ca The MACHBOX data case number (integer) to be processed.
Default: CASE=1
- MACH=(list) A list of 1 to 20 free-stream Mach numbers, M. Each Mach number must be greater than 1.0 and ≤ 5.0 . Valid solutions can be expected for Mach numbers in the range $1.2 \leq M \leq 3.0$.
Default: Error. Execution is terminated.
- KVAL=(list) A list of 1 to 20 k-values (reduced frequencies of oscillation expressed as $k = (\omega b/V)$ where ω is the circular frequency of oscillation, b is a reference length and V is the free stream velocity) greater than or equal to zero. The reference length associated with the specified k-values is defined by the parameter BREF. Each of the specified k-values may either be 0.0 or lie in the range 10^{-5} to 1000.0.
Default: Only the Mach Box aerodynamic grid is generated.

BREF=b

A reference length greater than zero associated with the reduced frequencies specified via KVAL.

Default: Specified k-values are assumed to have a reference length equal to the chordwise box length.

< ANTI
NONS >

Key-word denoting that an ANTI-symmetrical or a NONSymmetrical analysis is to be performed. No aerodynamic contributions of the surface(s) in the negative-y reference-frame half-space are included in a NONSymmetrical analysis.

Default: Symmetric aerodynamic analysis

< DIH1
DIH2 >

Key-word denoting that the dihedral angle of Surface 1 (DIH1) or the dihedral angle of Surface 2 (DIH2) is not to be used in calculation of the aerodynamic influence of the corresponding surface on itself. Both of these parameters may be used in a single statement.

Default: If neither one of these key-words is input, the dihedral angle of Surface 1 is included in calculation of the aerodynamic influence of Surface 1 on itself and on Surface 2. Additionally, the dihedral angle of Surface 2 is included in calculation of the aerodynamic influence of Surface 2 on itself.

SUBD<=n>

This parameter denotes that the Mach Box grid-subdivision refinement technique is to be used to reduce any irregular fluctuations in the calculated pressure distributions (see ref. 136-1). The integer "n" must be in the range $0 \leq n \leq 12$. This integer defines the maximum number of unsubdivided rows in the "effective area." If "n" is defaulted or the parameter "SUBD=0" is specified, one-third the total number of spanwise box rows on the planform is used.

Subdivision refinement consists of automatically dividing the local sending Mach Box grid into an even finer grid while the receiving grid of control points remains unchanged. This technique may be used only when the receiving and the sending boxes lie in the same plane. Thus, subdivision can be applied to a wing-tail configuration only when the surfaces are coplanar.

Default: Subdivision of boxes is not performed in the calculation of normal wash.

<SURFIT<=n>
<CHORDFIT<=n>

This parameter indicates that a least squares surface (SURFIT) or chordwise (CHORDFIT) polynomial of degree "n" ($0 \leq n \leq 10$) is to be fitted to the velocity potential distribution to minimize any irregular fluctuations in the calculated pressure distribution. If "n" is defaulted or specified as zero, the degree of the interpolation polynomial is automatically calculated as a function of the number of boxes on the surface or on the chords. The surface refinement is performed separately to each lifting surface. Interpolated values of the velocity potential are used in calculation of the generalized forces.

Default: Generalized forces are calculated from the unsmoothed velocity potential distribution.

AICTOL

Key-word denoting that the velocity-potential AIC (aerodynamic influence coefficient) values are to be calculated with an integration tolerance of 0.01 percent.

Default: AIC values are calculated with a tolerance of 1.0 percent.

PLYWOOD

Key-word denoting that the aerodynamic lifts and generalized forces are to be calculated using full box areas everywhere.

Default: Lift and generalized force calculations are based on the full areas and fractional areas, where applicable, of the boxes contained within the planform.

LEVEL= (List)

A list of 1 to 5 retention level indicators (integers) in the range 1 to 5 denoting which generated data are to be saved on the random file MACHRNF. Only the data that are retained can be interrogated subsequently during re-execution of this processor and for printout purposes (see sec. 236.3). The following table illustrates the type of data associated with each LEVEL indicator. See reference 1-2 for detailed descriptions of the data matrices.

Default: Mach Box grid data, generalized airforces, aerodynamic control matrix and AIC arrays stored in matrices BOXijmW, GF0ijmk, ACMij, AICPxxx, AICINDx and ACNijmk are always saved on file MACHRNF.

LEVFL	Description of Data	Matrix Names
1	Box data, generalized airforces, aerodynamic coefficients and the aerodynamic control matrix	BOXijmW, BOXijmT, GF0ijmk, SF0ijmk, GACijmk, GCiiijmk, SACijmk, SCiiijmk, MONijmk, ISPiijmk, MPTijmk, ACMI
2	Box lifts, sectional lifts and sectional moments	BLnijmk, SBnijmk, SLnijmk, SSnijmk, CMnijmk, SMnijmk
3	Velocity potentials and pressure difference coefficients	VPnijmk, SVnijmk, PCnijmk, SPnijmk
4	AIC matrices	AICPxxx, AICCxxx, AICWxxx, AICVxxx, AICMxxx, AICINDx, ACNijmk
5	Normal wash and off-planform sampling data	LNnijmk, Unnijmk, LTnijmk, UTnijmk, SWnijmk, STnijmk, DWPijmk, PSWijmk, PSTijmk, WSnijmk

Extensive amounts of data may be generated by the MACHBOX Processor depending on the aerodynamics model and the specified execution parameters. The maximum number of AIC matrices that may be generated for a particular CASE is restricted to 500. A box pattern data matrix, generalized airforce matrices and AIC arrays are generated for each combination of Mach number and reduced frequency specified by Plist for a particular data case. As a guide to the user, the number of AIC matrices (NAIC) generated for each combination of Mach number and reduced frequency by a single execution for various configurations is presented below.

- a) Single surface with zero dihedral (and off-planform sampling)
 - NAIC = 1 (+ NSC)
 - where NSC = number of sampling chords

- b) Single surface with non-zero dihedral (and off-planform sampling)
 - NAIC = 1 + NWC (+ 2*NSC)
 - where NWC = number of planform chords

- c) Two coplanar or parallel surfaces with zero dihedral
 $NAIC = 1 + NCO$
 where NCO = number of chords on Surface 2
- d) Two surfaces with the same dihedral or two surfaces with non-zero dihedral only for Surface 1
 $NAIC = 1 + NWC + 2*NTC$
 where NTC = number of Surface 2 chords
- e) Two surfaces where only Surface 2 has non-zero dihedral
 $NAIC = 2*NTC$
- f) Two surfaces with different non-zero dihedral
 $NAIC = NWC + 3*NTC$

Calculation of the AIC matrices is the most time-consuming part of a MACHBOX aerodynamic solution. To minimize computer time requirements, the MACHBOX Processor automatically reuses previously-calculated AIC matrices rather than recalculating them under certain circumstances. An AIC matrix saved on file MACHRNF via a preceding EXECUTE MACHBOX statement or via an ATLAS LOAD (sec. 200.0) statement may be reused. An AIC matrix is reused if the associated aerodynamic model has the same horizontal and/or vertical planform separation, the same Mach Box grid size calculated at the same Mach number and k-value with the same AICTOL integration tolerance. This capability allows the user to add and/or alter the structural modes used in the analysis in addition to applying thickness pressure-correction data or requesting off-planform wash samples based on previously-calculated AIC matrices.

236.3 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the aerodynamic data calculated by the MACHBOX Processor.

PRINT OUTPUT (MACHBOX, Plist)

List of optional parameters for Plist:

- CASE=Ca The aerodynamic data case number (integer) for which generated data are to be printed.
Default: CASE=1
- COND=Co The aerodynamic condition number (integer) specified by an EXECUTE MACHBOX parameter list for which generated data are to be printed.
Default: COND=1
- MACH=(List) A list of 1 to 20 Mach numbers associated with the selected COND number.
Default: Data generated for all Mach numbers associated with COND are printed.
- KVAL=(List) A list of 1 to 20 reduced frequency values associated with the selected COND number.
Default: Data generated for all k-values associated with COND are printed.
- LEVEL=(List) A list of 1 to 5 integers in the range 1 to 5 denoting which group(s) of generated data are to be printed. See description of LEVEL presented in section 236.2.
Default: Box pattern data and generalized airforces are printed.

238.0 MASS MODULES

The MASS Processor calculates all mass/weights data associated with an ATLAS mass finite-element model established via the mass data (sec. 138.0). Execution of the MASS Processor as discussed in section 238.2 generates one or more of the following types of data.

- a) A non-diagonal mass matrix whose rows correspond with the retained nodal degrees-of-freedom ordered via the BC data. This matrix is generated in the following manner: (1) a polyhedral region is associated with each retained node such that the enclosed volume is closest to the retained node, (2) the mass and inertia within each polyhedron are lumped at its center of gravity and then transferred to the associated retained node, (3) the mass matrix terms corresponding to the retained degrees-of-freedom are assembled to form the non-diagonal mass matrix.
- b) A diagonal mass matrix. This matrix is generated similarly to the one discussed in (a) except the mass and inertia contained within each polyhedron are lumped directly at the retained node.
- c) Lumped mass matrices associated with structural and non-structural finite elements of a mass model. These elemental matrices may be merged via the MERGE Processor (sec. 242.0) to form a gross mass matrix. This gross matrix may be reduced via the MULTIPLY Processor (sec. 244.0) to effect a Guyan-reduced mass matrix (ref. 238-1). All finite elements must be defined via Structural nodes (sec. 146.0).
- d) A panel-weight matrix associated with user-defined auxiliary panels (e.g., airload panels) specified via the mass data.
- e) Detailed weight-statement data associated with the printout levels specified via the mass data.
- f) Fuel data
- g) Payload data

Display of mass model data is supported by several ATLAS modules. Printout of the mass input data may be requested from the MASS Postprocessor (sec. 238.1). Plots of these data can

also be effected via the EXTRACT and GRAPHICS Postprocessors as discussed in sections 218.0 and 228.0.

Printout of the data calculated by the MASS Processor may be requested from the MASS Postprocessor (sec. 238.3). These data may also be displayed graphically via the EXTRACT and GRAPHICS Postprocessors.

238.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the mass input data (sec. 138.0). The printed data may be associated with a complete mass data set or with selected element, fuel, payload, concentrated mass and/or auxiliary panel data subsets.

PRINT INPUT (MASS, Plist)

The available options for Plist are described in sections 238.1.1 and 238.1.2.

238.1.1 Data-Set Print Option

This option allows the user to print all the mass input data or selected mass data subsets of a data set. The optional parameters for Plist are:

- | | |
|---------------|---|
| SET=Se | The mass data set number (integer) for which input data are to be printed.
Default: SET=1 |
| ELEMORDER=Ord | This parameter defines the ascending element-number order in which the element data are to be printed. Optional key-words for "Ord" are:

USERID -- User number order
INPUT -- Input record number order
INTERNAL -- Internal element number order

Default: ELEMORDER=USERID |
| NOELEMENTS | Key-word that suppresses printing of all non-structural element data.
Default: All non-structural element data associated with data set "Se" are printed. |
| NO{Eltype} | Key-word that suppresses printing of data for a selected non-structural element type. One or more of these key-words may be used in a single statement. The options are:

NORODS NOSPARS NOPLATES
NOBEAMS NOCOVERS NOSCALARS |

Default: Data are printed for all element types in set "Se."

NO{Subset}

Key-word that suppresses printing of a selected mass input-data subset. One or more of these key-words may be used in a single statement. The options are:

NOCONDITION -- Mass-distribution
condition data
NOFUEL -- Fuel condition data
NOPAYLOAD -- Payload condition data
NOMASS -- Concentrated mass data
NOPANELS -- Auxiliary panel data
NOLUMPING -- Mass lumping data
NOFACTOR -- Weight factor data
NOLABEL -- Weight-statement label
data

Default: All the foregoing types of data for set "Se" are printed.

238.1.2 Data-Subset Print Option

This option allows the user to print the mass data associated with selected mass data subsets. Additionally, the data associated with non-structural mass element subsets of a mass data set which have been identified via the SUBSET-DEFINITION Preprocessor (sec. 156.0) may be printed. The data associated with each subset are printed separately. The optional parameters for Plist are:

SUBSETS

Key-word required for subset data printing.

Default: The data set print option is assumed.

SET=Se

The mass data set number (integer) for which input data are to be printed.

Default: SET=1

Data

Key-word that initiates printing of a particular subset of mass input data. One or more of these key-words may be used in a single statement. The options are:

CONDITION --Mass-distribution
condition data

FUEL --Fuel condition data
 PAYLOAD --Payload condition data
 LUMPING --Mass lumping data
 FACTOR --Weight factor data
 LABEL --Weight-statement label data

Default: None of the foregoing types
 of data are printed.

Cxx, Gxx

These parameters denote that a concentrated mass (C) subset and/or auxiliary-panel subset geometry (G) are to be printed. Each of these parameters is comprised of a letter followed by a one or two-digit integer defined by the mass data. One or more of these parameters may be used in a single statement. The Asterisk Name Option (see sec. 200.2) may be used to identify subsets.

Default: Concentrated mass or panel subset geometry data are not printed.

Exxx<=Ord>

The data associated with the element subset identified by Exxx are printed in the order denoted by the key-word Ord:

$$\text{Ord} = \left\{ \begin{array}{l} \text{USERID} \\ \text{INPUT} \\ \text{INTERNAL} \end{array} \right\}$$

These print-orders are interpreted in a manner similar to the ELEMORDER key-word for data set printout.

Default: Ord=USERID

One or more of these parameters may be used in a single statement. A print-order remains in effect for the subsequent subset names in Plist as it is interrogated in a left-to-right order until it is reset.

The Asterisk Name Option (sec. 200.2) may be used to identify subset names.

Default: Element subset data are not printed.

NO{Eltype}

Key-word that suppresses printing of data for a selected element type that may be in subsets Exxx. One or more of the following optional key-words may be used in a single statement.

NORODS	NOSPARS	NOPLATES
NOBEAMS	NOCOVERS	NOSCALARS

Default: Data are printed for all element types included in subsets Exxx.

Example: Assume the non-structural mass element data and the associated MASS subsets Exxx have been defined in regards to the following Control Program statement.

```
PRINT INPUT (MASS, SUBSETS, E2, E201, E5=INPUT, E1*2=USERID)
```

This statement initiates printing of subsets E2, E201, E102, E112, ..., E192 in USERID order and E5 in INPUT order.

Printout of fuel, payload, concentrated mass and/or panel geometry subset data may be requested via the same statement illustrated above or may be requested separately from the mass element subset printout. For example, the statement

```
PRINT INPUT (MASS, SUBSETS, FUEL, PAYLOAD, C*, G5)
```

initiates printing of the following subsets: C1, C2, ..., C9, G5 and the fuel and payload data.

238.2 EXECUTE MASS STATEMENT

The following statement initiates execution of the MASS Processor. No other processor need be executed prior to this one.

EXECUTE MASS (Plist)

List of optional parameters for Plist:

SET=Se The mass data set number (integer) to be processed.
Default: SET=1

CONDITION=Co The mass distribution condition-number (integer) to be processed. The mass matrix or the panel-weight matrix for the specified CONDITION is to be calculated via OPTION 2 or 3.
Default: All mass conditions associated with data set "Se" are processed.

OPTION=Type This parameter defines which type of weight data or mass matrices are to be calculated. The optional values for "Type" (integer) are:

- 1--Element weights, inertias and totals thereof
- 2--Diagonal, reduced mass matrix
- 3--Non-diagonal, reduced mass matrix
- 4--Lumped mass matrices associated with each of the structural and non-structural finite elements. All elements must be defined via Structural nodes (sec. 146.0). This option is used when a Guyan-reduced mass matrix is to be generated for set "Se."

Default: OPTION=1

Caution: The mass associated with BRICK structural elements is ignored if OPTION=2 or OPTION=3 is used. For all options, the BRICK element weights are not included in the total structural weight.

CONMASS=Num The concentrated mass subset number (integer) for which element mass matrices are to be generated. This parameter is only effective for **OPTION=4**.
Default: Element mass matrices are generated for all concentrated mass subsets when **OPTION=4** is specified.

INERTIAS Key-word that causes moments of inertia to be calculated for each auxiliary weight panel. This parameter is only applicable when a panel-weight matrix is requested via the specified **CONDITION** number.
Default: Weight-panel moments of inertia are not calculated.

The mass associated with a substructured model (see sec. 130.0) may be calculated in one of the following ways:

- a) If the mass associated with the structural elements of lowest-level substructures is to be considered, all mass data must be defined relative to these equivalent **SET/STAGE** models. In this case, the same mass **CONDITION** number must be assigned to each of these mass data sets (sec. 138.0). Mass execution options 2, 3 or 4 may be used for these **SETS**. Reduced mass matrices for those **SETS** executed via **OPTION=4** are calculated most conveniently via the cataloged control statements presented in appendix E. The reduced mass matrices for higher-level substructures and the highest-level substructure are calculated by Guyan-reduction (see **M-REDUCE** and **REDUCE** procedures in appendix E).
- b) Mass data may be defined directly for a highest-level substructure strictly via a mass data set (sec. 138.0) for the equivalent **SET/STAGE** model. In this case, generation of the reduced mass matrix for the highest-level substructure is effected for the mass **SET** via mass execution option 2 or 3.

238.3 PRINT OUTPUT STATEMENT

Printout of the following data calculated by the MASS Processor may be requested:

- a) Detailed weight statements
- b) Structural and non-structural element-subset masses and geometry data
- c) Summary of total mass properties and condition data
- d) Mass matrices associated with a set of retained nodal degrees-of-freedom
- e) Panel-weight matrices
- f) Fuel and payload data

PRINT OUTPUT (MASS, Plist)

List of optional parameters for Plist:

Set=Se The mass data set number (integer) for which generated data are to be printed.
Default: SET=1

STATEMENT<=Code> This parameter denotes that the detailed weight-statement, as defined by the mass data (sec. 138.0), is to be printed according to the 3-digit integer Code, ABC, defined as follows:

A--Detail indicator denoting what type of data are to be printed for each component of the weight-statement.
The options are:

- 1--Total mass properties
- 2--Total mass properties and element mass properties
- 3--Total mass properties, element mass properties and element geometric data

B--Element data order indicator. This digit is ignored if A=1 but must be input. The options are:

- 1--Ascending user number order
- 2--Ascending internal number order
- 3--Ascending input-record number order

C--Inertia switch. The options are:

- 1--No inertias are printed
- 2--Inertias relative to the GLOBAL reference frame are printed
- 3--Inertias are printed relative to a coordinate system whose origin is at the center of gravity of the corresponding subset or element and whose orientation is the same as the GLOBAL reference frame.

Default: If the key-word STATEMENT is not used, no weight-statement is generated. If only the key-word is input, the total mass properties are printed (ABC=111).

<EKxxx<a><=Code>>
<EMxxx<a><=Code>>

These parameters initiate printing of the mass and geometric data for the elements in the stiffness (K) or mass (M) subset denoted by EKxxx and EMxxx, respectively. The characters xxx denote an element subset number defined via the SUBSET-DEFINITION Preprocessor.

The optional <a> is effective only for any SPAR, COVER and CCOVER element types (see appendices B and C) which may be included in the specified subset. This portion of the key-word is input as the 3 characters (U), (L) or (W). These characters denote that only the upper (U) surface and flange data, only the lower (L) surface and flange data, or only the SPAR web (W) data for the corresponding elements in the subset are to be printed.

"Code" is an optional 3-digit integer ABC defined as above for the STATEMENT parameter. This code is effective for

the subsequent subsets identified in Plist as it is interrogated in a left-to-right order until it is reset.

One or more of these parameters may be used in a single statement. The Asterisk Name Option (see sec. 200.2) may be used to identify subsets.

Defaults: If the words EKxxx or EMxxx are not input, no subset data printout is generated. The default for Code is 111.

SUMMARY

Key-word that initiates printing of the total mass-property data and a summary of the mass-distribution condition data.

Default: No summary data are printed.

MDC=MDCxxxY

This parameter initiates printing of the mass and/or panel-weight matrices identified by the name MDCxxxY for a mass-distribution CONDITION. One or more of these parameters may be used in a single statement. The Asterisk Name Option (see sec. 200.2) may be used to identify these matrices.

Default: No mass or panel-weight matrices are printed.

FUEL

Key-word which must be specified if any of the following parameters TABLES or TANKS is used. If only this key-word is specified, the total mass properties of the fuel-distribution conditions associated with data set "Se" are printed.

Default: No fuel-table data or fuel-distribution data are printed.

TABLES<=Par>

This parameter denotes which types of fuel-table data are to be printed. "Par" is a 2-digit integer, AB, defined as follows:

A--Fuel-tank geometry switch. The options are:

- 1--No geometries are printed
- 2--Geometries are printed

B--Inertia switch. The options are:

- 1--No inertias are printed
- 2--Inertias relative to the GLOBAL reference frame are printed
- 3--Inertias are printed relative to the c.g. of each fuel tank

Default: TABLES=11 if the key-word FUEL is specified.

TANKS<=Par>

This parameter denotes which types of fuel-distribution data are to be printed. "Par" is a 2-digit integer, AB, defined as follows:

A--Fuel-tank mass property switch. The options are:

- 1--Total mass properties
- 2--Mass properties for the individual tanks

B--Inertia switch. The options are:

- 1--No inertias are printed
- 2--Inertias relative to the GLOBAL reference frame are printed
- 3--Inertias are printed relative to the c.g. of the total fuel-distribution data (if A=1) or relative to the c.g. of each tank (if A=2)

Default: TANKS=11 if the key-word FUEL is specified.

PAYLOAD<=Par> This parameter denotes which types of payload-distribution data are to be printed. "Par" is a 2-digit integer, AB, defined as follows:

A--Payload (passengers and/or cargo holds) mass property switch. The options are:

- 1--Total mass properties
- 2--Mass properties for the individual passengers and/or cargo holds

B--Inertia switch. The options are:

- 1--No inertias are printed
- 2--Inertias relative to the GLOBAL reference frame are printed
- 3--Inertias are printed relative to the c.g. of each passenger and/or cargo hold

Default: PAYLOAD=11

Example PRINT statements:

```
PRINT OUTPUT (MASS, STATEMENT=213)
PRINT OUTPUT (MASS, EK1=313, EM14 (L), SUMMARY)
PRINT OUTPUT (MASS, MDC=MDC01**)
```

240.0 MATERIAL POSTPROCESSOR

Printout of the elastic properties and/or the design allowable stresses associated with the ATLAS Standard materials and user-defined Special or Composite materials (sec. 140.0) may be requested from the MATERIAL Postprocessor as described in section 240.1. Options are provided to display the following types of data for selected materials:

- a) The elastic properties associated with the first principal direction of Standard and Special isotropic materials
- b) The elastic properties associated with each principal direction of the materials
- c) Only the ultimate or only the yield allowable stresses
- d) Elastic properties and allowable stresses

PRECEDING PAGE BLANK NOT FILMED

240.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the Material data.

PRINT INPUT (MATERIAL, Plist)

List of optional parameter for Plist:

⟨STANDARD
Mxx
Cxx⟩

This parameter denotes the material reference codes for which elastic properties and/or design allowable stresses are to be printed. Optional values of this parameter are:

<u>STANDARD</u>	--All Standard materials
M1,M2,....,M10	--Selected Standard materials
M51,M52,....,M99	--Selected Special materials
C1,C2,....,C31	--Selected Composite materials

One or more of these parameters may be used in a single statement. The Asterisk Name Option (sec. 200.0) may be used to identify material codes.

Default: Material data are printed for all Special and Composite materials that have been defined.

⟨ELASTIC
ALLOWABLES ⟨=U⟩
⟨=Y⟩⟩

This parameter denotes that only the ELASTIC properties or only the design ALLOWABLES are to be printed for the specified materials. If ALLOWABLES is used, printout may be further limited to only the ultimate (U) or only the yield (Y) stress data. One or more of these parameters may be used in a single statement.
Default: ELASTIC properties and ALLOWABLE stresses are printed.

ISOTROPIC

Key-word denoting that only the elastic properties associated with the first principal direction of the material are to be printed. This key-word is not applicable to Composite materials.
Default: Elastic property data associated with each of the principal material directions are printed.

242.0 MERGE PROCESSOR

The MERGE Processor assembles elemental and substructure elastic and geometric stiffness, mass, loads and displacement matrices to generate gross structural matrices or selected submatrices thereof for subsequent analyses. Additionally, this processor performs specialized functions that support back-substitution operations required for stress/displacement analyses of substructure models. Execution of this module to effect structural analyses via the ATLAS System is illustrated by the cataloged control statements presented in appendix E. These cataloged statements are a user convenience that obviates the need, in most cases, to use the explicit EXECUTE MERGE statement.

C-7

PREVIOUS PAGE BLANK NOT FILMED

242.1 EXECUTE MERGE STATEMENT

The following statement initiates execution of the MERGE Processor. Previous execution of other processors is required to generate the data matrices used by this processor. In all cases, the STIFFNESS Processor must be executed prior to execution of this processor.

EXECUTE MERGE (Plist)

List of optional parameters for Plist:

STIFFNESS
GSTIFFNESS
MASS
LOADS
DISPLACEMENTS
PARTITION
EXPANSION

Key-word denoting that elastic or geometric stiffness (STIFFNESS or GSTIFFNESS), MASS, LOADS or DISPLACEMENT matrices are to be assembled (merged) or that substructure back-substitution operations (PARTITION and EXPANSION) are to be performed. Use of the latter two key-words is discussed in detail below. Only one of these key-words may be used in a single statement.
Default: Error. Execution is terminated.

If the MASS option is selected, there must be a mass property associated with each retained kinematic freedom for the specified "STAGE." If this is not the case, execution is terminated subsequent to printing a list of all such deficiencies. Elemental mass matrices and concentrated masses (see CONMASS parameter) contribute to the mass properties associated with the retained freedoms.

Matrices to be assembled must be associated with one or more structural model components each identified by either a data set number and boundary condition (BC) stage number (SET/STAGE), a substructure number (SS) or a buckling set number (BSET). The component identifier SET/STAGE may not be used with SS or BSET in Plist

SET=se

Either the stiffness data-set number (integer) for which STIFFNESS, LOADS or DISPLACEMENT matrices are to be assembled or the mass data set number (integer) identifying which MASS matrices are to be assembled. Finite-element, elastic stiffness matrices must have

been calculated via the STIFFNESS Processor (sec. 252.2), loads and displacements prepared via the LOADS Processor (sec. 234.1) and finite-element mass matrices calculated by OPTION=4 of the MASS Processor (sec. 238.2).
Default: SET=1

STAGE=St

The BC stage number (integer) associated with the specified stiffness or stiffness/mass data set for which the specified type of matrices is to be assembled.
Default: STAGE=1

SS=Sub

A substructure number (integer), assigned via the interact data (sec. 130.0), for which the specified operation is to be performed.
Default: SET/STAGE component matrices are to be assembled.

BSET=Num

The buckling set number (integer) for which GSTIFFNESS matrices are to be assembled. Finite-element, geometric stiffness matrices must have been calculated via the STIFFNESS Processor (sec. 252.2). If this parameter is used with the "SS" parameter, the component geometric stiffness matrices to be assembled must be User Matrices identified as GREDxxx, where "xxx" is a component substructure number.
Default: BSET=1 if GSTIFFNESS is specified.

CONMASS=Num

The number (integer) of a subset of concentrated masses, defined via the mass data (sec. 138.0), which are to be assembled. Concentrated masses may be assembled with elemental mass matrices previously generated for the SET/STAGE component. This parameter is only applicable when the MASS option is selected.
Default: Nc concentrated masses are assembled.

RETAIN=Code

This parameter denotes the action to be taken when an inactive kinematic

freedom (one that has zero stiffness) has been retained. The options available for the integer "Code" are:

- 1--Error. Execution is terminated.
- 2--Warning. The requested operation is executed.

Regardless of the selected "Code," if an inactive freedom has been retained for subsequent analyses or if a load has been applied to an inactive freedom, a list of all such freedoms is printed.

Default: RETAIN=1

OPTION=Code

This parameter denotes the action to be taken when the name of a matrix specified via the "Name" parameter is the same as the name of a previously-generated User Matrix. The options for the integer "Code" are:

- 1--The specified operations are performed and the contents of the previously-generated matrix "Name" are over-written.
- 2--The specified operations are not performed. Subsequent control statements, however, are executed. This option allows the user to avoid recalculation of previously-generated matrices as in the case of problem-execution restart or during substructure analyses.

In both cases, a warning message is issued.

Default: OPTION=1

Name=Index

This parameter allows the user to specify that only a particular submatrix of a gross matrix for a SET/STAGE or BSET component is to be generated. "Name" and "Index" are as follows:

Name--Alphanumeric name of the resulting User Matrix (see sec. 200.0).

Index--A two-digit integer identifying which submatrix of a particular gross matrix (elastic or geometric stiffness, mass, loads or displacements) is to be generated. Each of these digits is either 1, 2 or 3 corresponding to the freedom activity labels FREE, RETAIN and SUPPORT, respectively. The first digit defines the row-partition number, whereas the second digit defines the column-partition number (e.g., 11=FREE-FREE, 23=RETAIN-SUPPORT, etc.) of the matrix that is to be generated. The DISPLACEMENT matrix must be identified by 31 and the LOADS matrices must be identified by 11, 21 and 31.

Whenever a parameter of this type is input, only the specified matrices are assembled. If more than one matrix is to be generated, this parameter is repeated as necessary. All nine partition matrices of a general gross matrix may be requested. This parameter is not applicable to assembling substructure matrices.

Default: If this parameter is not input when SET/STAGE or BSET component matrices are to be assembled, all diagonal and above-diagonal partition matrices of the corresponding gross matrix are assembled. Each partition (User Matrix) is automatically assigned a name ABCXXYY where

A--The letter D, G, K, L or M associated with the type of matrices being assembled:

D--DISPLACEMENTS
G--GSTIFFNESS
K--STIFFNESS
L--LOADS
M--MASS

BC--Two characters that identify a particular partition matrix. For the G, K and M type matrices, each character is the first letter of one of the kinematic-

freedom activity labels (see sec. 106.1). These letters are F, R and S for the labels FREE, RETAIN and SUPPORT, respectively. The first letter in this pair identifies the row address of the matrix, whereas the second letter identifies the column address of the matrix (e.g., FF=FREE-FREE, RS=RETAIN-SUPPORT, etc.). The DISPLACEMENT matrix is identified by the character pair S1, whereas the LOADS matrices are identified by F1, R1 and S1.

XX--The STAGE number or BSET number.

YY--The SET number or blank if BSET is specified.

Substructure (SS) component User Matrices assembled by the MERGE Processor are identified by system-standard 7-character names of the form ABCSXXX where

A--The letter D, G, K, L or M associated with the type of matrices being assembled:

D--DISPLACEMENTS
G--GSTIFFNESS
K--STIFFNESS
L--LOADS
M--MASS

BC--A two-digit integer identifying the row address and column address, respectively, of the partition matrix. Each of these digits is either 1, 2 or 3 corresponding to the kinematic-freedom activity labels FREE, RETAIN and SUPPORT, respectively. The digit C is always 1 for LOADS and DISPLACEMENT matrices and the digit B is always 3 for DISPLACEMENT matrices.

S--The letter S.

XXX--The substructure number.

The diagonal and above-diagonal partition matrices of the corresponding gross matrix for the specified "SS" are always generated via a single statement. No matrix-partition selectivity is available.

The following two parameters (PARTITION and EXPANSION) are applicable only to support back-substitution operations

required for stress/displacement analyses associated with a higher-level substructure denoted by the parameter "SS=Sub." See appendix E and F for applications of these parameters.

PARTITION

Key-word denoting that the displacement User Matrices DBS2xxx, DBS2yyy, etc., associated with the retained kinematic freedoms of substructures xxx, yyy, etc., which were interacted to define "SS" are to be generated. These matrices are generated by appropriate partitioning of the displacement matrices DBS1---, DBS2--- and DBS3--- previously calculated for "SS" by an interaction solution. The digit 1, 2 or 3 in a matrix name denotes the partition-matrix row address, whereas --- is the integer number of "SS."

Default: Partitioning of the displacement matrices is not performed.

EXPANSION, Out1=In1, Out2=In2, ...

This list of parameters denotes that the selected load-case dependent User Matrices associated with components xxx, yyy, etc., of the specified "SS" are to be expanded column-wise, as necessary, for compatible back-substitution calculations. An EXPANSION of the selected matrices (In1, In2, etc.) is performed such that the column order and dimension of the resulting matrices (Out1, Out2, etc.) are compatible with the column dimension of the displacement matrices DBS2xxx, DBS2yyy, etc., previously calculated for the corresponding components of "SS" (see the PARTITION option). Proper expansion of load-case dependent matrices is required because different load cases may be associated with the different substructure components.

Outi--Four alphanumeric characters to be used in forming the 7-character names of the resulting set of User Matrices. Each of the matrix names is comprised of these four characters followed directly by the corresponding substructure component number right-

adjusted and zero-filled. See the OPTION parameter regarding uniqueness of these matrix names.

Ini--The first four alphanumeric characters of a previously-generated, load-case dependent User Matrix which is to be expanded. The types of matrices which may be expanded are exemplified by the following matrices generated via the substructure-analysis catalogued statements presented in appendix E: LREDxxx, ESUBxxx, CSUBxxx and D31Sxxx.

Default: If none of these parameters is input, expansion of load-case dependent matrices is not performed. If, however, only the key-word EXPANSION is input, execution is terminated.

Example: The matrices of specified SUPPORT-displacements associated with the components of substructure 10 are identified by D31Sxxx, D31Syyy, etc. The column dimension of D31Sxxx, for example, is equal to the number of load cases associated with component xxx. The matrix of retained-freedom displacements for substructure 10, DBS2010, however, will generally have a larger column dimension than D31Sxxx, etc. A larger column dimension of matrix DBS2010, as calculated via the PARTITION option, is due to additional load cases associated with the components of substructure 10. The statement

```
EXECUTE MERGE (EXPANSION, SS=10, DBS3=D31S)
```

causes the D31Sxxx, D31Syyy, etc., matrices associated with the components of substructure 10 to be expanded accordingly, named DBS3xxx, DBS3yyy, ..., respectively, and saved for subsequent analyses.

The remaining five parameters (COND, COER, COWR, COEL and COWL) are applicable only if the STIFFNESS option is used. These parameters allow several types of ill-conditioning checks to be performed on the diagonal partition-matrices of the gross elastic-stiffness matrix. These checks provide the user with indicators of the level of ill-conditioning of the structural,

mathematical model prior to solving the stiffness equilibrium equations via the Cholesky Processor (sec. 210.0).

Two levels of conditioning referenced as the "single pass" and "double pass" checks may be performed. In the following discussion, K_{ij} denotes a submatrix of the gross matrix, wherein the "i" rows and "j" columns correspond to the kinematic freedoms of internal nodes "i" and "j", respectively.

The "single-pass" check is performed by calculating the eigenvalues of each diagonal submatrix K_{ii} . Based on the minimum and maximum eigenvalues (L_{mins} and L_{maxs} , respectively) and the user-selected ratios "erat" and "wrat," the following action is taken:

- a) Error if; $L_{mins} \leq 0$.
or, $(L_{maxs}) / (L_{mins}) > erat$
- b) Warning if; $erat > (L_{maxs}) / (L_{mins}) > wrat$

The "double-pass" check, if selected, is performed subsequent to performing the "single-pass" check only if no errors were encountered. For each K_{ii} matrix, a modified submatrix \bar{K}_{ii} is formed,

$$\bar{K}_{ii} = K_{ii} - \sum_{i \neq j} K_{ij} * (K_{jj})^{-1} * K_{ji}$$

Based on the minimum and maximum eigenvalues of this matrix (L_{mind} and L_{maxd} , respectively) and the ratios "erat" and "wrat," the following action is taken:

- a) Error if; $L_{mind} \leq 0$.
or, $(L_{maxd}) / (L_{mind}) > erat$
- b) Warning if; $erat > (L_{maxd}) / (L_{mind}) > wrat$

Furthermore, if no errors were encountered, the smallest eigenvalue calculated by the "single-pass" (L_{mins}) is compared with the smallest eigenvalue calculated by the "double-pass" (L_{mind}) for each of the K_{ii} submatrices. Based on these eigenvalues and the user-selected ratios "emin" and "wmin," the following action is taken:

- a) Error if; $(L_{mins}) / (L_{mind}) > emin$
- b) Warning if; $emin > (L_{mins}) / (L_{mind}) > wmin$

The conditioning-check parameters are as follows:

COND=Type The type of conditioning-check to be performed. The three options for the integer "Type" are:

- 0--No conditioning check
- 1--"Single-pass" check
- 2--"Double-pass" check

Default: COND=0

COER=erat Number that defines the error condition for the ratio $(L_{maxs})/(L_{mins})$ and for the ratio $(L_{maxd})/(L_{mind})$.
Default: COER= 10^{14}

COWR=wrat Number that defines the warning condition for the ratio $(L_{maxs})/(L_{mins})$ and for the ratio $(L_{maxd})/(L_{mind})$.
Default: COWR= 10^4

CCEL=emin Number that defines the "double-pass" error condition for the ratio $(L_{mins})/(L_{mind})$.
Default: CCEL= 10^{14}

COWL=wmin Number that defines the "double-pass" warning condition for the ratio $(L_{mins})/(L_{mind})$.
Default: COWL= 10^4

244.0 MULTIPLY PROCESSOR

The MULTIPLY Processor evaluates a user-defined matrix expression. The expression may denote the addition of one matrix to the sum of one to five matrix products. Execution of this module to effect structural analyses via the ATLAS System is illustrated by the cataloged control statements presented in appendix E. These cataloged statements are a user convenience that obviates the need, in most cases, to use the explicit EXECUTE MULTIPLY statement.

~~PRECEDING PAGE BLANK NOT FILMED~~

244.1 EXECUTE MULTIPLY STATEMENT

The following statement initiates execution of the MULTIPLY Processor. Previous execution of other processors is required to generate the data matrices used by this processor.

EXECUTE MULTIPLY (Plist)

List of optional parameters for Plist:

Matexp This parameter denotes the matrix expression to be evaluated. The expression must be defined in the form

$$\text{Res}=[\pm \text{Mat1} \pm \text{Mat2} * \text{Mat3} \pm \dots \text{Mat10} * \text{Mat11}]$$

where the brackets enclosing the matrix operations on the right-hand side of the equal sign must be punched. The remaining items are:

Mati--Alphanumeric name of a previously-generated User Matrix (sec. 200.0) to be processed.

Res--Name of the resulting User Matrix defined via this statement. See the OPTION parameter below regarding uniqueness of this matrix name. If the add matrix and each of the matrix products involve a null matrix, the result matrix is null. A non-zero result matrix is stored in rectangular format unless a certain condition is satisfied, in which case, it is stored in the triangular User-Matrix format. Each of these conditions is as follows:

- a) There is no add matrix and all matrix products are of the type $A(T) * A$ where A is a rectangular matrix.
- b) There is a triangular (symmetric) add matrix and all matrix products involve only rectangular matrices.
- c) The three characters (S) are appended to the result matrix name, all matrix products involve only rectangular

MULTIPLY

matrices and the add matrix, if included in the expression, is triangular (symmetric).

Default: Error. Execution is terminated.

Within the bracketed expression, one to five simple matrix products with or without one simple matrix add may be defined. The following rules must be satisfied:

- a) Allowable operation symbols are + (plus), - (minus), * (multiply) and T (transpose). A leading plus operation symbol may be omitted. The sequence of a matrix add and the matrix products is arbitrary. A matrix may be transposed prior to post-multiplication by another matrix. This is denoted by appending the three characters (T) to the name of the left matrix of a matrix product. For example, matrix B pre-multiplied by the transpose of matrix A is denoted by $A(T)*B$.
- b) No diagonal matrices are allowed. Any one of the "Mat" User Matrices may be rectangular or null. Only the add matrix or the first matrix of a product may be stored in the triangular format.
- c) Compatibility of the dimensions (orders) of the matrices to be processed is the user's responsibility.

OPTION=Code This parameter denotes the action to be taken when the name of the result matrix "Res" is the same as the name of a previously-generated User Matrix. The options for the integer "Code" are:

- 1--The specified operations are performed and the contents of the previously-generated matrix "Res" are over-written.
- 2--The specified operations are not performed. Subsequent control statements, however, are executed. This option allows the user to avoid recalculation of previously-generated matrices as in the case of problem-execution restart or during substructure analyses.

In both cases, a warning message is issued.

Default: OPTION=1

246.0 NODAL POSTPROCESSOR

Printout of the nodal data (sec. 146.0) associated with a structural or mass model may be requested from the NODAL Postprocessor as described in the following section. The printed information is comprised of node-number identifiers and coordinates.

REPRODUCED FROM ORIGINAL PAGE BLANK NOT FILMED

246.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the nodal input data. The printed data may be associated with a complete nodal data set or with nodal subsets.

PRINT INPUT (NODAL, Plist)

The available options for Plist are described in sections 246.1.1 and 246.1.2.

246.1.1 Data-Set Print Option

This option allows the user to print the nodal data associated with a particular data set. The optional parameters for Plist are:

SET=Se	The nodal data set number (integer) to be processed. Default: SET=1
NODEORDER = Ord	This parameter defines the ascending node-number order in which the nodal data are to be printed. Optional key-words for "Ord" are: USERID -- User number order INPUT -- Input record number order INTERNAL -- Internal node number order Default: NODEORDER=USERID

246.1.2 Data-Subset Print Option

This option allows the user to print the nodal data associated with specified node subsets extracted via the SUBSET-DEFINITION Preprocessor (sec. 156.0). The data associated with each subset are printed in a separate block. The optional parameters for Plist are:

SUBSETS	Key-word required for subset data printing. Default: The data set print option is assumed.
SET=Se	The nodal data set number (integer) to be processed. Default: SET=1

Nxxx<=Ord>

The data associated with the node subset identified by Nxxx are printed in the order denoted by the key-word Ord:

$$\text{Ord} = \left\{ \begin{array}{l} \text{USERID} \\ \text{INPUT} \\ \text{INTERNAL} \end{array} \right\}$$

These print-orders are interpreted in a manner similar to the NODEORDER key-word for data set prints.

Default: Ord=USERID

One or more of these parameters may be used in a single statement. A print-order remains in effect for the subsequent names in Plist as it is interrogated in a left-to-right order until it is reset.

The Asterisk Name Option (see sec. 200.0) may be used to identify subset names.

Default: The nodal data for all node subsets of set "Se" as defined via the SUBSET-DEFINITION Preprocessor are printed in the USERID order.

248.0 REACTION POSTPROCESSOR

Printout of reaction forces, reaction-force resultants and equilibrium checks of all the externally applied forces on a structural model may be requested from the REACTION Postprocessor as presented in the following section.

~~FOLLOWING PAGE BLANK NOT FILMED~~

248.1 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the reaction forces and the reaction-force resultants associated with a structural model. Additionally, an equilibrium check of all external forces applied to a model may be requested. These data may be printed for a structural model identified by either a stiffness data-set number and boundary-condition stage number or a lowest-level substructure number. Only one of the identifier types (SET/STAGE or SS described below) may be used in a single statement. Execution of this postprocessor is allowed only after the reactions are calculated (see the PERFORM STRESS control procedure in appendix E).

PRINT OUTPUT (REACTIONS, Plist)

List of optional parameters for Plist:

SET=Se	The number (integer) of the stiffness data set for which reaction forces are to be printed. Default: SET=1
STAGE=St	A boundary condition (BC) stage number (integer) associated with set "Se." Default: STAGE=1
SS=CATlist	ATLAS list of lowest-level substructure numbers (integers) for which reaction forces are to be printed. Default: A SET/STAGE model.
LC=CATlist	ATLAS list of load case identifiers (integers or alphanumeric words) to be processed. Default: All load cases that have been processed previously for the specified SET/STAGE or SS model.
EQCHK=<Node>	This parameter denotes that the equilibrium check is to be performed. The resultants of the reaction forces are compared to the applied load resultants. The resultants are calculated relative to the node identified by the user node

number "Node" (integer). If only the key-word "EQCHK" is input, the resultants are calculated relative to the origin of the GLOBAL triad.

Default: No equilibrium check is performed.

< R2=Name2 >
< R3=Name3 >

Alphanumeric names of the reaction-force matrix partitions (User Matrices) generated for the specified SET/STAGE or SS. These matrices are associated with the RETAIN and SUPPORT kinematic freedoms of the gross stiffness matrix (see sec. 106.1). Both of these parameters may be used in a single statement. Zero matrix partitions need not be specified. Reactions associated with the RETAIN partition are printed in the specified order of the retained nodal freedoms, whereas reactions associated with the SUPPORT partition are printed in an ascending internal node number order.

If the standard ATLAS control procedures are used (ref. appendix E and sec. 200.4), the name R31 is assigned to Name3 for SET/STAGE structural models, and the names RBS2xxx and RBS3xxx are assigned for lowest-level substructures. The characters xxx denote the substructure number, right-adjusted and zero-filled.

Defaults: R3=R31 for SET/STAGE models;
R2=RBS2xxx, R3=RBS3xxx for substructure models.

FORMAT=Code

This parameter defines the format of each printed value. The optional values available for "Code" (integer) are:

0--Optimum format. Data values are printed as either F-format or E-format decimal numbers. This option allows the maximum number of digits per value to be printed in the field width specified by the parameter FIELDW=width.

1--F-format. Data values are printed as decimal numbers with decimal points aligned in the columns of printed data.

2--I-format. Data values are printed as integer numbers.

3--E-format. Data values are printed as decimal numbers with exponents.

Default: FORMAT=0

FIELDW=width

Number of digits (integer ≤ 12) available for each printed value. The letter E, decimal points and signs each require one digit space.

Default: FIELDW=8

250.0 RHO3 AERODYNAMICS MODULES

The RHO3 Processor calculates unsteady aerodynamic loads on rigid or elastic, planar lifting surfaces with or without one to four trailing-edge controls in subsonic compressible flow. The solution technique is based on the subsonic kernel function and assumed pressure-mode approach presented in reference 150-1.

Generalized airforce matrices required for flutter analyses via the ADDINT and FLUTTER Modules (secs. 202.0 and 222.0) are generated by execution of the RHO3 Processor (sec. 250.2). These data are calculated by a modal solution technique based on the geometry and structural modes defined by the user (see sec. 150.0). If the RHO3 Processor is executed in conjunction with the AIC option of the INTERPOLATION Processor (sec. 232.0), the generalized airforce matrices will represent Aerodynamic Influence Coefficient (AIC) matrices. Optional calculated data include aerodynamic pressure and sectional generalized forces at selected locations on a lifting surface planform.

Printout of the RHO3 input data and data calculated by the RHO3 Processor may be requested from the RHO3 Postprocessor as described in sections 250.1 and 250.3, respectively.

~~REVISION PAGE BLANK NOT FILMED~~

250.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the aerodynamic input data associated with a particular RHO3 data case (sec. 150.0).

PRINT INPUT (RHO3, Plist)

The optional parameter for Plist is:

CASE=Ca

The aerodynamic data case number
(integer) for which input data are to
be printed.
Default: CASE=1

250.2 EXECUTE RH03 STATEMENT

The following statement initiates execution of the RH03 Processor. Previous execution of the INTERPOLATION Processor is required when structural mode shapes generated via the VIBRATION Processor are used in the aerodynamic analysis or Aerodynamic Influence Coefficients are to be generated (see sec. 150.0).

EXECUTE RH03 (Plist)

List of optional parameters for Plist:

- | | |
|---------------------------------|---|
| COND=co | A condition number (integer) in the range 1 to 36 which identifies the remaining parameters of this statement. If multiple executions are required in a single job, the condition number specified by each statement of this type must be unique.
Default: COND=1 |
| CASE=Ca | The RH03 data case number (integer) to be processed.
Default: CASE=1 |
| MACH=(list) | A list of 1 to 20 free-stream Mach numbers, M. Each Mach number must be in the range $0.0 \leq M < 1.0$.
Default: Error. Execution is terminated. |
| KVAL=(list) | A list of 1 to 20 k-values (reduced frequencies of oscillation expressed as $k = (\omega b/V)$ where ω is the circular frequency of oscillation, b is a reference length and V is the free-stream velocity) greater than or equal to zero. The specified k-values must satisfy the expression $k \leq 0.1*b$.
Default: Error. Execution is terminated. |
| <p>< ANTI
NON S ></p> | Key-word denoting that an ANTIsymmetrical or a NONSymmetrical analysis is to be performed. No contributions of the aerodynamic model in the negative-y half-space of the reference frame are included in a NONSymmetrical analysis.
Default: Symmetric aerodynamic analysis. |

Due to the form of the assumed pressure modes used in a solution, each aerodynamic-pressure/downwash-kernel influence coefficient matrix (C-matrix) is a function of planform geometry, downwash point locations, k-value and Mach number. Calculation of C-matrices is the most time-consuming part of a RHO3 aerodynamic solution. To minimize computer time requirements, the RHO3 Processor automatically reuses previously-calculated C-matrices rather than recalculating them under certain circumstances. That is, velocity profile variations, addition of different structural modes, and/or alterations of structural modes may be investigated by using previously-calculated C-matrices. A C-matrix saved on file RHO3RNF via a preceding EXECUTE RHO3 statement or via an ATLAS LOAD (sec. 200.0) statement is reused if it is associated with an aerodynamic model that has the same main surface identification, the same number of planform-definition and downwash points calculated with the same symmetry option at the same Mach number and k-value. The actual planform geometries and the downwash point distributions are not examined. Therefore, the main surface identification must be used to identify different problems with the same number of geometry-definition points and control points. If control surfaces are inherent to the model, the control surface identification and geometry must also be identical.

250.3 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the aerodynamic data calculated by the RHO3 Processor.

PRINT OUTPUT (RHO3, Plist)

List of optional parameters for Plist:

- CASE=Ca The aerodynamic data case number
(integer) for which generated data are
to be printed.
Default: CASE=1
- COND=Co The aerodynamic condition number
(integer) specified by an EXECUTE RHO3
parameter list for which generated data
are to be printed.
Default: COND=1
- MACH= (List) A list of 1 to 20 Mach numbers associated
with the selected COND number.
Default: Data generated for all Mach
numbers associated with COND
are printed.
- KVAL= (List) A list of 1 to 20 reduced frequency
values associated with the selected
COND number.
Default: Data generated for all k-values
associated with COND are
printed.
- LEVEL= (List) A list of 1 to 3 integers in the range
1 to 3 denoting which group(s) of
generated data are to be printed. Each
of the LEVEL indicators is as follows:
- 1-- Structural mode shapes interpolated
to the aerodynamic control points and
generalized airforces.
 - 2-- Sectional generalized force and pressure
report data.
 - 3-- Downwash matrices and C-matrices.
- Default: LEVEL=1

252.0 STIFFNESS MODULES

The STIFFNESS Processor generates the following types of element matrices for the stiffness elements of a structural model. These matrices are calculated relative to the analysis frames of the nodes used to define the elements.

- a) Elastic stiffness matrices. Used for all structural analyses. Rows and columns in each matrix for which the diagonal term is less than 10^{-12} times the largest diagonal term are automatically eliminated.
- b) Elastic stress matrices. Used by the STRESS Processor (sec. 254.0) for calculation of element stresses.
- c) Geometric stiffness matrices. Used for nonlinear analyses. See section 208.0 for solving general-instability problems.

Structural models defined by ATLAS stiffness elements (sec. 152.0 and appendix B) are analyzed via the ATLAS System by the finite-element displacement method (ref. 252-1). Gross, structural, stiffness matrices are formed by the MERGE Processor (sec. 242.0) and the equilibrium equations are solved via the CHOLESKY and MULTIPLY Processors (secs. 210.0 and 244.0).

Display of structural model data is supported by several ATLAS modules. Printout of nodal data from the NODAL Postprocessor (sec. 246.0), boundary condition data from the BC Postprocessor (sec. 206.0) and element data from the STIFFNESS Postprocessor (sec. 252.1) may be requested as necessary. Plots of these data can be effected via the EXTRACT and GRAPHICS Postprocessors as discussed in sections 218.0 and 228.0.

PREVIOUS PAGE BLANK NOT FILMED

252.1 PRINT INPUT STATEMENT

The following statement is used to request printout of the stiffness-element input data. The printed data may be associated with a complete stiffness data set or with element subsets.

PRINT INPUT (STIFFNESS, Plist)

The available options for Plist are described in sections 252.1.1 and 252.1.2.

252.1.1 Data-Set Print Option

This option allows the user to print the element data associated with a particular data set. The optional parameters for Plist are:

SET=Se The stiffness data set number (integer) for which input data are to be printed. Default: SET=1

ELEMORDER=Ord This parameter defines the ascending order in which the data are to be printed. Optional key-words for "Ord" are:

 USERID -- User element-number order
 INPUT -- Input record-number order
 INTERNAL -- Internal element-number order

 Default: ELEMORDER=USERID

NO{Eltype} Key-word that suppresses printing of data for a selected element type. One or more of these key-words may be used in a single statement. The options are:

NORODS	NOPLATES	NOSRODS
NOBEAMS	NOGPLATES	NOSPLATES
NOSPARS	NOBRICKS	NOCPLATES
NOCOVERS	NOSCALARS	NOCCOVERS

Default: Data are printed for all element types in data set "Se."

PROPERTY={VALUES }
 {CODE } This parameter defines how property data are to be displayed per element. If VALUES is selected, element property labels and the corresponding values are printed in a vertical column. If CODE is selected, the property data are identified in the printout only by the input section-property reference code, if used, or the property values are printed in horizontal lines, 4 values per line, without property labels. Fewer pages are generated for the CODE option.
 Default: PROPERTY=VALUES

ELEMNODES=Enode This parameter is applicable only if the PROPERTY=CODE parameter is used. It identifies which node numbers are to be displayed per element. Optional key-words for "Enode" are:

USER -- User node numbers only
 INTERNAL -- Internal node numbers only
 BOTH -- User and internal node numbers
 NONE -- Node numbers are not displayed

Default: ELEMNODES=USER if the parameter PROPERTY=CODE is used. Otherwise, the user and the internal node numbers are printed.

252.1.2 Data-Subset Print Option

This option allows the user to print the stiffness data associated with specified element subsets extracted from a stiffness data set via the SUBSET-DEFINITION Preprocessor (sec. 156.0). The data associated with each subset are printed in a separate block. The optional parameters for Plist are:

SUBSETS Key-word required for subset data printing.
 Default: The data-set print option is assumed.

SET=Se The stiffness data set number (integer) for which input data are to be printed.
 Default: SET=1

Exxx<=Ord>

The data associated with the element subset identified by Exxx are printed in the order denoted by the key-word Ord:

$$\text{Ord} = \left\{ \begin{array}{l} \text{USERID} \\ \text{INPUT} \\ \text{INTERNAL} \end{array} \right\}$$

These print-orders are interpreted in a manner similar to the ELEMORDER key-word for data-set prints.

Default: Ord=USERID

One or more of these parameters may be used in a single statement. A print-order remains in effect for the subsequent subset names in Plist as it is interrogated in a left-to-right order until it is reset.

The Asterisk Name Option (see sec. 200.0) may be used to identify subset names.

Default: The element data for all stiffness subsets of set "Se" as defined via the SUBSET-DEFINITION Preprocessor are printed in the USERID order.

NO (Eltype)

Key-word that suppresses printing of data for a selected element type which may be in subsets Exxx. One or more of the following optional key-words may be used in a single statement.

NORODS	NOPLATES	NOSRODS
NOBEAMS	NOGPLATES	NOSPLATES
NOSPARS	NOBRICKS	NOCPLATES
NOCOVERS	NOSCALARS	NOCCOVERS

Default: Data are printed for all element types included in subsets Exxx.

**PROPERTY={VALUES } Same as defined for the data-set
 {CODE } print option**

**ELEMNODES=Enode Same as defined for the data-set
print option**

STIFFNESS

Example: Assume the stiffness data and associated data subsets E1, E5, E10 and E12 have been defined in regards to the following Control Program statements.

```
PRINT INPUT (STIFFNESS, SUBSETS)
PRINT INPUT (STIFFNESS, SUBSETS, E5, E12=INPUT,
             E10=INTERNAL, E*, E5=INPUT)
```

The first print statement initiates printing of the stiffness data associated with subsets E1, E5, E10 and E12 in USERID order. The second statement initiates printing of the data associated with the subsets as listed below.

<u>Subset</u>	<u>Print-Order</u>
E5	USERID
E12	INPUT
E10	INTERNAL
E1	INTERNAL
E5	INTERNAL
E5	INPUT

252.2 EXECUTE STIFFNESS STATEMENT

The following statement initiates execution of the STIFFNESS Processor.

EXECUTE STIFFNESS (Plist)

Execution of this processor to calculate the elastic stiffness and stress matrices and the geometric stiffness matrices for a model are described in sections 252.2.1 and 252.2.2, respectively.

252.2.1 Elastic Stiffness and Stress Matrices

No other processor need be executed prior to using this execution option. The optional parameters for Plist are:

SET=Se	The stiffness data set number (integer) to be processed. Default: SET=1
RUN=Type	This parameter defines whether elastic stiffness and/or stress element matrices for the structural model are to be generated. The optional key-words for "Type" are: <u>STIFFNESS</u> -- Stiffness matrices only <u>STRESS</u> -- Stress matrices only <u>BOTH</u> -- Stiffness and stress matrices Default: RUN=BOTH
LUMP=factor	The fraction ($0.0 \leq \text{factor} \leq 1.0$) of SPAR-element lumped flange areas as specified by the stiffness input data (see sec. 152.0 and appendix B) to be used in calculating SPAR-element stiffnesses. If the lumping factors as specified by the input data are to be effective, the parameter LUMP=1.0 must be included in Plist. Default: LUMP=0.0

Caution: Printout of SPAR element flange areas are those specified by the stiffness input data.

BIGBRICK

Key-word denoting that each BRICK-element stress matrix should relate six stress components at each node.

Default: Only six stresses at the centroid of each BRICK element of the model are calculated (see Appendix E).

xxPRINT

Key-word that initiates printing of a particular type of element matrix. One or more of the following key words may be used in a single statement.

LKPRINT -- Stiffness matrices relative to element reference frames
LPPRINT -- Stress matrices relative to element reference frames
NAPRINT -- Transformation matrices relating element and analysis reference frames
GKPRINT -- Stiffness matrices relative to node analysis frames
GPPRINT -- Stress matrices relative to node analysis frames

Default: No element matrices are printed.

252.2.2 Geometric Stiffness Matrices

Geometric stiffness matrices are load-case dependent. Hence, previous execution of the STRESS Processor (sec. 254.0) for calculation of element stresses is required. Currently, geometric stiffness matrices may be calculated for the following element types: ROD, BEAM, PLATE, GPLATE and BRICK. If BRICK elements are included in the model, the parameter BIGBRICK, as discussed in section 252.2.1, may not be used in conjunction with calculation of geometric stiffness matrices. The optional parameters for Plist for this execution option are:

BSET=n

A buckling (large displacement) set number (integer) in the range 1 to 36. This number is used to identify the triplet of parameters SET, STAGE and LC specified by this statement. If multiple executions are required in a single job, the buckling set number specified by each statement of this type must be unique.
 Default: BSET=1

SET=Se The stiffness data set number (integer) to be processed.
Default: SET=1

STAGE=St A boundary condition stage number (integer) associated with set "Se" for which geometric stiffness matrices are to be calculated.
Default: STAGE=1

LC=Label Identifier of the load case (integer or alphanumeric word) associated with the specified SET and STAGE for which geometric stiffness matrices are to be calculated. Stresses for this load case must have been calculated previously.
Default: Error. Execution is terminated.

Generation of elastic stiffness and/or stress matrices may be requested by the same EXECUTE STIFFNESS statement used to generate geometric stiffness matrices. For this case, the "RUN=Type" parameter discussed in the preceding section must be specified explicitly in Plist. The optional parameters "LUMP=factor" and "xxPRINT" may then also be used.

254.0 STRESS MODULES

The STRESS Processor assembles displacements, computes stresses and computes element nodal forces for the load cases defined by the loads data (sec. 134.0) and the stress data (sec. 154.0).

For a BC (boundary condition) STAGE, this processor:

- a) Assembles the previously calculated structural displacement matrices D1, D2 and D3 (ref. sec. 106.0) into a single matrix containing the displacements for all active, structural freedoms (freedoms with non-zero stiffness).
- b) Computes stresses and internal, nodal forces for all elements in a structural model.

For a superposition stage, SUPSTAGE, this processor:

- a) Generates a single matrix of displacements by superimposing displacements for all active, structural freedoms. Load case components of a SUPSTAGE are defined by the stress data (sec. 154.0).
- b) Generates element stresses either by computation from the superimposed displacements or by superimposing previously-calculated stresses for selected SUPSTAGE components.
- c) Generates internal, element-nodal forces from the superimposed displacements.

Displacements and element nodal forces are measured relative to the selected node analysis frames (sec. 146.0), whereas stresses are measured relative to the corresponding element reference frames (see appendix B).

Printout of the assembled displacements, computed stresses and the internal, element nodal forces may be requested from the STRESS Postprocessor as described in section 254.2. Printed-page formats for these data may be specified by the user.

RECORDED PAGE BLANK NOT FILMED

254.1 EXECUTE STRESS STATEMENT

Execution of the STRESS Processor is initiated by the following statement. Previous calculation of the element stress matrices and the displacement matrix partitions is required. Execution of the STIFFNESS, LOADS, MERGE, CHOLESKY and MULTIPLY Processors is usually required prior to execution of this processor (see PERFORM STRESS in appendix E). Structural models for which the STRESS Processor may be executed are identified either by a stiffness data-set number and boundary-condition STAGE or SUPSTAGE number or by a lowest-level substructure number. Only one of the identifier types (SET/STAGE, SET/SUPSTAGE or SS as described below) may be used in a single statement.

EXECUTE STRESS (Plist)

List of optional parameters for Plist:

- | | |
|-------------------------------|--|
| SET=Se | The number (integer) of the stiffness data set to be processed.
Default: SET=1 |
| { STAGE }
{ SUPSTAGE } =St | The number (integer) of a BC (boundary condition) STAGE or a SUPSTAGE (superposition stage) associated with set "Se." Only one of these parameters may be specified. If SUPSTAGE is used, displacements for the SUPSTAGE components must have been assembled via previous execution of the STRESS Processor.
Default: STAGE=1 |
| SS=Sub | The number (integer) of a lowest-level substructure to be processed.
Default: A SET/STAGE or SET/SUPSTAGE model is assumed. |
| LC=CATlist | ATLAS list of load case identifiers (integers or alphanumeric words) for which stresses are to be calculated.
Default: All load cases associated with the specified SET/STAGE, SET/SUPSTAGE or SS. |

Note: The LC parameter is applicable to options 2, 3, 11 and 12 as identified by the OPT parameter described below.

OPT= { Num }
 { (List) }

This parameter identifies the types of calculations to be performed. Each option is identified by an integer number. One option may be specified by "Num" or several options may be specified by "List."

For SET/STAGE or SS models, the options available for "Num" and "List" are:

- 1--Assemble displacements for all load cases associated with the specified STAGE or SS.
- 2--Calculate stresses for the specified LC. If only this option is used, the associated displacements must have been assembled previously.
- 3--Calculate element nodal forces for the specified LC. If only this option is used, the associated displacements must have been assembled previously.

Default: OPT=(1,2)

For SET/SUPSTAGE models, the options available for "Num" and "List" are:

- 10--Superimpose displacements for all load cases associated with the specified SUPSTAGE.
- 11--Calculate stresses for the specified SUPSTAGE loadcases. If only this option is used, the associated displacements must have been superimposed previously.
- 12--Calculate element nodal forces for the specified SUPSTAGE loadcases. If only this option is used, the associated displacements must have been superimposed previously.
- 13--Superimpose stresses for the specified SUPSTAGE. If only this option is used, the associated stresses must have been calculated previously.

Defaults: OPT=(10,11) If the stresses associated
OPT=(10,13) with the specified SUPSTAGE
are not available on file
STRERNF, they are calculated.
Otherwise, they are
superimposed.

D1=Name1 }
D2=Name2 }
D3=Name3 }

Alphanumeric names of the displacement
User Matrices generated for the
specified SET/STAGE or SS. These
matrices are associated with the three
partition rows (FREE, RETAIN and SUPPORT
kinematic freedoms) of the gross
stiffness matrix (see sec. 106.1).
One or more of these parameters may
be used in a single statement. Zero
matrix partitions need not be specified.
These parameters are only applicable
to the OPT=1 parameter.

If the standard ATLAS control procedures are
used (ref. appendix E and sec. 200.0) the
names D11, D21 and D31 are assigned to Name1,
Name2 and Name3 for SET/STAGE structural
models, and the names DBS1xxx, DBS2xxx and
DBS3xxx are assigned for lowest-level sub-
structures. The characters xxx denote the
substructure number, right-adjusted and zero-
filled. ..

Defaults: An error message is issued
and execution is terminated
for a SET/STAGE model.
D1=DBS1xxx, D2=DBS2xxx,
D3=DBS3xxx for an SS model.

INTERNAL

Key-word that suppresses ordering of
the calculated data according to user
node-numbers and element-numbers. This
key-word allows for increased efficiency
when the calculated data are not to
be printed via the STRESS Postprocessor
(sec. 254.2). Ordering of displacements
and stresses is not required, for
example, for intermediate, resize design
CYCLES (see sec. 212.0 and appendix
F).

Default: The calculated data are ordered according to the user-defined node and element numbers.

Caution: If the calculated data are to be displayed by the ATLAS Postprocessors, the key-word INTERNAL should not be used.

254.2 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the nodal displacements, element stresses and internal, element nodal forces generated by the STRESS Processor. An option is provided in section 254.2.2 to request printout of lamina stresses and strains and margins of safety for CPLATE and CCOVER composite elements. If this option is not used, laminate stresses and strains are printed for composite elements. An additional option is provided in section 254.2.3 to request printout of the displacements and stresses contained in data blocks previously established via execution of the EXTRACT Postprocessor (sec. 218.0). For each option, the printed data may be associated with a complete structural model or with node and element subsets defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0).

Displacements and element nodal forces are measured relative to the node analysis frames (sec. 146.0), whereas stresses are measured relative to the element reference frames (see appendix B). Rotations are measured in radians.

PRINT OUTPUT (Plist)

The available options for Plist are described in sections 254.2.1, 254.2.2 and 254.2.3.

254.2.1 Displacement/Stress/Nodal-Force Print Options

Structural models for which data are to be printed are identified by either a stiffness data-set number and boundary-condition number or a lowest-level substructure number. Only one of the identifier types (SET/STAGE or SS as described below) may be used in a single statement. The optional parameters for Plist are:

DISPLACE
STRESSES
FORCES

Key-word denoting whether displacements, stresses or internal, element nodal forces are to be printed. Only one of these key-words may be specified. Default: Warning. No printout is generated.

Caution: If FORCES are to be printed, they must have been requested by using the OPT parameter in the EXECUTE STRESS or PERFORM STRESS statement.

- SET=Se** The number (integer) of the stiffness data set for which data are to be printed.
Default: SET=1
- STAGE=St** A boundary condition (BC) or superposition stage number (integer) associated with set "Se."
Default: STAGE=1
- SS=Sub** The number (integer) of a lowest-level substructure for which data are to be printed.
Default: A SET/STAGE model.
- LC=CATlist** ATLAS list of load case identifiers (integers or alphanumeric words) to be processed.
Default: All load cases associated with the specified SET/STAGE or SS.

The following two parameters are effective only for printing nodal displacements (DISPLACE option).

◊ REC
CYL
SPH ◊

Optional key-word denoting the kinematic-freedom labels (see sec. 106.0), associated with one of the three types of analysis reference frames (RECTangular, CYLindrical and SPHERical). The labels are used as headings of printed columns within a group of data. Only one set of column labels may be selected in a single statement.
Default: RECTangular analysis frame labels are used as column headings.

- Nxxx** Node subset name defined via the SUBSET-DEFINITION Preprocessor. The Asterisk Name Option (sec. 200.0) may be used to identify subset names. Several parameters of this type may be used in a single statement. Displacements are printed only for the nodes in the specified subset(s).
Default: All nodes in the specified SET or SS.

The following parameter is effective only for printing BRICK element stresses via the STRESS option.

$BIGBRICK = \left\{ \begin{array}{l} \underline{NODAL} \\ \underline{ELEMENT} \end{array} \right\}$ This parameter denotes that BRICK element nodal stresses are to be printed on a node-by-node basis (NODAL) or on an element-by-element basis (ELEMENT). This parameter is applicable only if nodal stresses were calculated for BRICK elements (see BIGBRICK option of the STIFFNESS Processor, sec. 252.0).
Default: BIGBRICK=ELEMENT if nodal stresses were calculated. If not, stresses relative to the centroids of the BRICK elements are printed (see appendix B).

The following parameter is effective only for printing element stresses or element nodal forces (the STRESS or FORCE option).

EXXX Element subset name defined via the SUBSET-DEFINITION Preprocessor. The Asterisk Name Option (sec. 200.0) may be used to identify subset names. Several parameters of this type may be used in a single statement. Stresses or nodal forces are printed only for the elements in the specified subset(s).
Default: All elements in the specified SET or SS.

LINES=lines Maximum number of lines (integer ≤ 56) including page headings printed on a page.
Default: LINES=56

The following six parameters are effective only for the DISPLACE or STRESS print options. These parameters allow the user to specify the printed-page format for groups of data each printed with NCOL columns. The number of columns (NCOL) within a group of displacement data is the maximum number of active freedoms in set "Se." The number of columns (NCOL) within a group of stress data is the maximum number of stresses per element type to be printed (see appendix B for the number of stresses calculated for each element type). Several groups of data (NGRO) may be printed across a page via use of the COMPACT and GROUPSP

parameters described below (NGRO is 1, by default). When establishing a printed-page format, the algebraic value of one of the following expressions must be less than or equal to 132 (standard computer printout width). If displacements are printed, the expression is

{MARGIN+30+NGRO[10+NCOL (COLSP+FIELDW)]+GROUPSP*NGRO}

If stresses are printed, the expression is

{MARGIN+NGRO[10+NCOL (COLSP+FIELDW)]+GROUPSP (NGRO-1)}

FORMAT=Code This parameter defines the format of each printed value within each group of data. The options available for the integer "Code" are:

0--Optimum format. Data values are printed as either F-format or E-format decimal numbers. This option allows the maximum number of digits per value to be printed in the field width specified by the parameter FIELDW=width.

1--F-format. Data values are printed as decimal numbers with decimal points aligned in the columns of printed data.

2--I-format. Data values are printed as integer numbers.

3--E-format. Data values are printed as decimal numbers with exponents.

Default: FORMAT=0

MARGIN=margin The number (integer) of blank spaces defining the left margin of each printed page.
Default: MARGIN=10 for DISP option
 MARGIN=1 for STRESS option

COLSP=space Number of blank spaces (integer) between columns of printed data values within each group of data.
Default: COLSP=2

FIELDW=width Number of digits (integer) available for each printed value. The letter E,

decimal points and signs each require one digit space.

Default: FIELDW=8

COMPACT Key-word denoting that multiple groups of data are to be aligned across the page.
Default: Only one group of data is printed across the page.

GROUPSP=group Number of blank spaces (integer) between groups of data printed across the page. This parameter is effective only if the key-word **COMPACT** is used.
Default: GROUPSP=5

254.2.2 Composite-Element, Lamina Stress/Strain Print Options

This option allows the user to print stresses, strains and margins of safety for each lamina of selected **CPLATE** and **CCOVER** composite elements. Stress/strain data for the laminates of composite elements are printed via the options described in sections 254.2.1 and 254.2.3.

The data printed via this option may be associated with selected load cases and element subsets. Structural models are identified by either a stiffness data-set number and boundary-condition number or a lowest-level substructure number. Only one of the identifier types (**SET/STAGE** or **SS** as described below) may be used in a single statement. The optional parameters for **Plist** are:

LAMINA \langle =STRESS \rangle This parameter identifies whether
 \langle =STRAIN \rangle lamina stresses (**STRESS**) or strains (**STRAIN**) are to be printed. Only one of these parameters may be specified.
Default: Stresses and strains are printed if only the key-word **LAMINA** is specified. Otherwise, no printout is generated.

SET=Se The number (integer) of the stiffness data set for which data are to be printed.
Default: SET=1

STAGE=St A boundary condition (**BC**) or superposition stage number (integer) associated with set "Se."
Default: STAGE=1

SS=Sub The number (integer) of a lowest-level substructure for which data are to be printed.
Default: A SET/STAGE model.

LC=CATlist ATLAS list of load case identifiers (integers or alphanumeric words) to be processed.
Default: All load cases associated with the specified SET/STAGE or SS.

Lxxx Element subset name defined via the SUBSET-DEFINITION Preprocessor. The Asterisk Name Option (sec. 200.0) may be used to identify subset names. Several parameters of this type may be used in a single statement. Data are printed only for the composite elements in the specified subset(s).
Default: All composite elements in the specified SET or SS.

The options provided below allow lamina stress and strain margins of safety to be printed. Strain margins of safety are based on a maximum strain criterion, whereas the stress margins of safety are based on Hill's criterion.

Lamina strain margins are defined as (ref. appendix B),

$$\begin{aligned} MS1 &= |(EPS1^+/EPS1)| - 1 \\ MS2 &= |(EPS2^+/EPS2)| - 1 \\ MS3 &= |(GAM12^+/GAM12)| - 1 \end{aligned}$$

where,

$$\begin{aligned} EPS1^+ &= FTU1/E1 - \nu_{12} * FTU2/E2 && \text{if } EPS1 \geq 0 \\ EPS1^+ &= FCU1/E1 - \nu_{12} * FCU2/E2 && \text{if } EPS1 < 0 \\ EPS2^+ &= FTU2/E2 - \nu_{12} * FTU1/E1 && \text{if } EPS2 \geq 0 \\ EPS2^+ &= FCU2/E2 - \nu_{12} * FCU1/E1 && \text{if } EPS2 < 0 \\ GAM12^+ &= FSU/G \end{aligned}$$

Lamina stress margins are defined as

$$MS-HILL = (1/R^{1/2}) - 1$$

where,

$$R = \frac{(SIG1/F1^+)^2 + (SIG2/F2^+)^2 + (TAU12/F3^+)^2}{(SIG1 * SIG2)/(F1^+)^2}$$

and

$$\begin{aligned} F1^+ &= FTU1 && \text{if } SIG1 \geq 0 \\ F1^+ &= FCU1 && \text{if } SIG1 < 0 \\ F2^+ &= FTU2 && \text{if } SIG2 \geq 0 \\ F2^+ &= FCU2 && \text{if } SIG2 < 0 \\ F3^+ &= FSU \end{aligned}$$

Both types of margin of safety data may be printed for the critical load cases via the following parameters:

MS $\left\langle \begin{array}{l} =\text{MAXSTRAIN} \\ =\text{HILL} \end{array} \right\rangle$

This parameter denotes that strain margins (MAXSTRAIN) or stress margins (HILL) are to be printed. Only one of these parameters may be specified. Defaults: Both types of margins of safety are printed if only the key-word MS is specified. Otherwise, no margins of safety are printed.

MSCUTOFF=max

Lamina stress, strain and margin of safety, MS, data are printed for the "LC" load cases for which $MS \leq \text{max}$. If the corresponding MS values are greater than "max," lamina stress, strain and MS data are printed for those load cases (a maximum of 3) which have the minimum margins of safety.

Defaults: If MS=MAXSTRAIN is specified, only the minimum value of MS1, MS2 and MS3 for each load case is printed.

If MS=HILL is specified, only the minimum value of MS-HILL for each load case is printed.

254.2.3 Printout of EXTRACTED Displacements and Stresses

This option allows the user to print the nodal displacements and element stresses previously generated by the STRESS Processor and stored in data blocks via execution of the EXTRACT Postprocessor (sec. 218.0). The printed data may be associated with selected load cases and node/element subsets. The optional parameters for Plist are:

$\left\{ \begin{array}{l} \text{EXDISPLACE} \\ \text{EXSTRESS} \end{array} \right\} = \text{Exname}$

This parameter denotes whether extracted displacements (EXDISPLACE) or stresses (EXSTRESS) are to be printed. "Exname" is the alphanumeric name previously assigned to the extracted data block via the EXECUTE EXTRACT statement (sec. 218.0). The data component labels

associated with "Exname" must include at least those identified by the standard label subset DISPRINT and STRPRINT for displacement and stress printout, respectively (see sec. 156.0). These names are generally used for "Exname." Only one of these parameters may be specified.

Default: Warning. No printout is generated.

LC=CATlist

ATLAS list of load case identifiers (integers or alphanumeric words) to be processed.

Default: All load cases associated with "Exname."

The following two parameters are effective only for printing nodal displacements (EXDISPLACE option).

REC
CYL
SPH

Optional key-word denoting the kinematic-freedom labels (see sec. 106.0), associated with one of the three types of analysis reference frames (RECTangular, CYLindrical and SPHERical). The labels are used as headings of printed columns within a group of data. Only one set of column labels may be selected in a single statement.

Default: RECTangular analysis frame labels are used as column headings.

Nxxx

Node subset name defined via the SUBSET-DEFINITION Preprocessor. The Asterisk Name Option (sec. 200.0) may be used to identify subset names. Several parameters of this type may be used in a single statement. Displacements are printed only for the nodes in the specified subset(s).

Default: All nodes associated with "Exname."

The following parameter is effective only for printing element stresses (EXSTRESS option).

Exxx

Element subset name defined via the SUBSET-DEFINITION Preprocessor. The

Asterisk Name Option (sec. 200.0) may be used to identify subset names. Several parameters of this type may be used in a single statement. Stresses are printed only for the elements in the specified subset(s).

Default: All elements associated with "Exname."

LINES=lines Maximum number of lines (integer ≤ 56) including page headings printed on a page.
Default: LINES=56

The following parameters allow the user to specify the printed-page format for groups of data each printed with NCOL columns. The number of columns (NCOL) within a group of displacement data is the maximum number of active freedoms associated with "Exname." The number of columns (NCOL) within a group of stress data is the maximum number of stresses per element type to be printed (see appendix B for the number of stresses calculated for each element type). Several groups of displacement data (NGRO) may be printed across a page via use of the COMPACT and GROUPSP parameters described below (NGRO is 1, by default). When establishing a printed-page format, the algebraic value of one of the following expressions must be less than or equal to 132 (standard computer printout width). If displacements are printed, the expression is

{MARGIN+30+NGRO[10+NCOL (COLSP+FIELDW)]+GROUPSP*NGRO}

If stresses are printed, the expression is

{MARGIN+ 10+NCOL (COLSP+FIELDW) }

FORMAT=Code This parameter defines the format of each printed value within each group of data. The options available for the integer "Code" are:

0--Optimum format. Data values are printed as either F-format or E-format decimal numbers. This option allows the maximum number of digits per value to be printed in the field width specified by the parameter FIELDW=width.

1--F-format. Data values are printed as decimal numbers with decimal points aligned in the columns of printed data.

2--I-format. Data values are printed as integer numbers.

3--E-format. Data values are printed as decimal numbers with exponents.

Default: FORMAT=0

MARGIN=margin

The number (integer) of blank spaces defining the left margin of each printed page.

Default: MARGIN=10 for EXDISPLACE option
MARGIN=1 for EXSTRESS option

COLSP=space

Number of blank spaces (integer) between columns of printed data values within each group of data.

Default: COLSP=2

FIELDW=width

Number of digits (integer) available for each printed value. The letter E, decimal points and signs each require one digit space.

Default: FIELDW=8

The following parameters are effective only for printing nodal displacements via the EXDISPLACE option.

COMPACT

Key-word denoting that multiple groups of displacement data are to be aligned across the page.

Default: Only one group of data is printed across the page.

GROUPSP=group

Number of blank spaces (integer) between groups of displacement data printed across the page. This parameter is effective only if the key-word COMPACT is used.

Default: GROUPSP=5

258.0 VIBRATION MODULES

The VIBRATION Processor solves the structural vibration eigenproblem. Natural, undamped mode-shapes, frequencies and generalized stiffness/mass matrices can be calculated for the real system of linear, homogeneous, algebraic equations denoted by $(K-E*M)*Q=0$ or $(F*M-E)*Q=0$ associated with a structural stiffness (K) or flexibility (F) formulation, respectively. Each of the matrices K, F and M is symmetric, where M is a diagonal or nondiagonal, positive-definite mass matrix. Matrices E and Q contain the eigenvalues and eigenvectors associated with the vibration frequencies and modes, respectively. The stiffness/mass formulation may represent a structural model which is free to assume 1 to 6 rigid-body motions or a model which is completely constrained from assuming any rigid-body motion. The flexibility/mass formulation, however, represents a completely constrained structural model.

Eigensolutions of both types of equation systems are effected by the following distinct steps:

- a) $M=L*L^T$; if the mass matrix is nondiagonal, it is decomposed via the Cholesky square root method (ref. 258-1) into a lower triangular matrix post-multiplied by its transpose.
- b) The eigenproblem is reduced to the standard, symmetric eigenproblem denoted by $D*X=E*X$, where $D=L^{-1}*K*(L^{-1})^T$ or $D=L^T*F*L$ for the stiffness and flexibility matrix formulations, respectively.
- c) The dynamic matrix D is tridiagonalized by the Householder-Givens technique (ref. 258-2) via similarity transformations.
- d) Eigenvalues are extracted from the tridiagonal matrix via either the symmetric Q-R orthogonal-matrix, transformation technique or the Sturm-sequencing bisection method (ref. 258-2).
- e) Eigenvectors are calculated by the Wielandt inverse iteration technique (ref. 258-2).
- f) Orthogonalization of the calculated eigenvectors is assured by use of the Gram-Schmidt technique (ref. 258-2).
- g) Each eigenvector is normalized to its largest component.

REPRODUCED FROM BLANK NOT FILMED

The numerical quality of an eigensolution is measured via several checks performed by the VIBRATION Processor (sec. 258.1). In one case, each calculated eigenvalue and associated eigenvector are substituted into the original equation to be solved. The deviation of the i -th resulting residual vector $R(i)$ from a null vector is measured by its unbiased root-mean-square value $RMS(i)$ calculated as the square root of $[R(i)^T * R(i)] / (n-1)$ where n is the order of matrix D . These quantities, which are a measure of the equilibrium or compatibility of the stiffness and flexibility-formulated system of equations, respectively, are printed for each eigensolution.

A second solution check, referenced as the orthogonality check, is a measure of how well the calculated eigenvector (mode) matrix succeeds in diagonalization of the mass matrix. That is, the off-diagonal elements of the generalized mass matrix calculated as $Q^T * M * Q$ are theoretically zero. These elements, however, as calculated by the triple matrix product may differ from zero. Therefore, a map of these elements is automatically displayed for each eigensolution. This solution check indicates how well the mode matrix Q transforms the coupled set of equations of motion, expressed via the physical coordinate system, into an uncoupled set of equations expressed in terms of the normal, generalized coordinate system.

Subsequent to an eigensolution, the VIBRATION Processor performs the following additional ill-conditioning check on the eigenproblem: (1) the "Norm" of the D matrix is calculated; the absolute values of the elements in the i -th row of this matrix are summed to form $s(i)$; $i=1, n$. The maximum $s(i)$ is defined as the "Norm" of the D matrix, (2) if the absolute value of the i -th calculated eigenvalue $e(i)$ is less than the quantity $(Norm * 10^{-12})$, it is automatically set to zero; the quantity within parentheses is referenced as the cutoff value. Clearly, if the ill-conditioning is caused by a rigid-body mode inherent to the vibration problem, a zero eigenvalue is just what is required. However, if it is caused by an inferior mathematical model of the physical problem, a zero eigenvalue is an adequate indicator that an error exists in the model. If any negative eigenvalues remain after the "cutoff" check is performed, execution is terminated. This indicates that the eigenproblem to be solved is non-positive definite and that the mathematical model must be redefined and analyzed again.

The foregoing eigensolution checks are performed so that the user is advised of the quality of the solution. A structural model may, however, have legitimate ill-conditioning, such as a singular stiffness matrix representing a free or partially constrained structure, in addition to illegitimate ill-conditioning. The in-plane translational stiffness at a node

of a model, for example, is generally much larger than the rotational stiffness at the same node. While there is nothing incorrect about retaining both types of kinematic freedoms for dynamic analyses, the rounding errors incurred when performing the required mathematical operations may lead to substantial inaccuracies in the small eigenvalues. Since the smaller eigenvalues are, in general, of primary interest when the stiffness/mass formulation is used, the results of a vibration analysis and the mathematical model details must be examined carefully.

If an eigenproblem describes a model that undergoes rigid-body motion(s), the coupled rigid-body modes as calculated by the eigensolution may be replaced with user-selected uncoupled modes.

Printout of the vibration analysis data may be requested from the VIBRATION Postprocessor as discussed in section 258.2. Plots of these data can also be effected via the EXTRACT and GRAPHICS Postprocessors as discussed in sections 218.0 and 228.0.

258.1 EXECUTE VIBRATION STATEMENT

The following statement initiates execution of the VIBRATION Processor. A reduced stiffness or flexibility matrix and the corresponding reduced mass matrix must have been generated previously. Reduced ATLAS stiffness and flexibility matrices are generated via execution of the MERGE, CHOLESKY and MULTIPLY Processors. An ATLAS mass matrix may be generated directly for the retained kinematic freedoms by executing OPTION 2 or 3 of the MASS Processor (sec. 238.2). It may also be a Guyan-reduced mass matrix generated via execution of the MERGE, CHOLESKY and MULTIPLY Processors. Cataloged control statements for generation of reduced stiffness, flexibility and Guyan-reduced mass matrices are presented in appendix E. Vibration analyses of a highest-level substructure are performed for the "equivalent" SET/STAGE model (see sec. 130.0). The maximum permitted order of the eigenproblem is approximately 400.

EXECUTE VIBRATION (Plist)

List of optional parameters for Plist:

VSET=xx	A vibration set number (integer) in the range 1 to 99 which is used to identify the vibration problem specified by this statement. If multiple statements of this type are required in a single job, the vibration problem associated with a particular VSET number may be redefined. Default: VSET=1
MASS=Name 1	Name of a previously-generated, reduced mass matrix. Default: Error. Execution is terminated.
{ STIF=Name2 FLEX=Name3 }	Name of a previously-generated stiffness (STIF) User Matrix or flexibility (FLEX) User Matrix (see sec. 200.0). Only one of these parameters may be used. Default: Error. Execution is terminated.

The eigenproblem to be solved is one of the following matrix equations:

$$\begin{aligned}(\text{Name 2}-\text{FREQSxx}*\text{Name 1})*\text{MODESxx}&=0 \\(\text{Name 3}*\text{Name 1}-\text{FREQSxx})*\text{MODESxx}&=0\end{aligned}$$

The calculated eigenvalue matrix and eigenvector matrix are identified by FREQSxx and MODESxx, respectively, where "xx" is the vibration set number, right-adjusted and zero-filled. The corresponding generalized mass and generalized stiffness matrices are automatically calculated, saved and identified as GMASSxx and GSTIFxx, respectively. These four matrices are managed via the User-Matrix Name Catalog (see sec. 200.0).

The following SET and STAGE parameters are required if any one of the parameters SUBSETS, URBM, PRBM or TRBM, as defined below, is used. It is always a good practice, however, to specify the SET and STAGE associated with the eigenproblem so that node/freedom identifiers associated with the modal components can be displayed by the VIBRATION Postprocessor (sec. 258.2).

SET=Se The data set number (integer) identifying the stiffness/mass model associated with the eigenproblem.
Default: SET=1

STAGE=St The boundary condition stage number (integer) associated with set "Se" for which a vibration analysis is to be performed.
Default: STAGE=1

⟨QR
STURM⟩

Key-word denoting whether the QR method or the Sturm-sequence (STURM) method is to be used to extract the eigenvalues. The QR method is more efficient than the STURM method. The STURM method is, however, more stable and therefore allows small eigenvalues to be calculated with greater accuracy.
Default: QR

NFREQS=nf Number (integer) of eigenvalues (frequencies) to be saved. It should be noted that if the QR method is used, all the eigenvalues associated with the eigenproblem are calculated automatically. Only "nf" eigenvalues are calculated, however, if the STURM method is used. The eigenvalues corresponding with the lowest "nf" vibration frequencies are saved.
Default: The order of the eigenproblem.

NMODES=nm Integer number ($0 < nm \leq nf$) of eigenvectors (modes) to be calculated and saved.
Default: NMODES=nf

The following parameter allows selected modal displacements to be extracted from the calculated eigenvector matrix "MODESxx." These displacements are associated with a subset of the degrees-of-freedom used to define the eigenproblem. Subsets of the retained nodes must have been defined via the SUBSET-DEFINITION Preprocessor (sec. 156.0). Generation of subset mode-matrices is required only for interpolation of vibration mode-shapes from the structural freedoms to the aerodynamic control points established by the unsteady aerodynamics processors of ATLAS (see sec. 232.0). For these purposes, the subset mode-matrix must correspond with kinematic freedoms that have the same analysis frame.

SUBSETS=CATlist ATLAS list of node subsets for which subset mode-matrices are to be generated. The Asterisk Name Option (see sec. 200.0) may be used to identify subsets. Non-retained nodes in the specified subsets are ignored.
Default: No subset mode-matrices are generated.

Example: SUBSETS=N2 TO N7, SUBSETS=(N9, N2*, N50), N100 TO N104, SUBSETS=N200

If the structural model is not constrained from assuming rigid-body (RB) motion, 1 to 6 RB coupled modes will be included in the mode matrix. These RB modes may be replaced by a set of user-selected, optionally normalized, uncoupled RB modes as described by the following parameters. A replacement mode is either a unit translation vector or a unit rotation vector. If the model is partially constrained, it is the responsibility of the user to insure that the RB-mode replacements are compatible with the structural supports. If any of the following parameters is used, the MASS Processor (sec. 238.2) should be executed to calculate the total-weight matrix prior to execution of the VIBRATION Processor.

The user-selected RB modes are associated with either the GLOBAL frame, the principal axes of the model or a specified rectangular triad. These reference frames are identified by the key-words URBM, PRBM and TRBM, respectively.

URBM -- A reference frame whose origin is at the center of gravity (cg) of the model as calculated by the MASS Processor and which has the same orientation as the GLOBAL triad. The weight matrix for the model (see the discussion at end of this section) will contain nonzero products of inertia if the axes of this reference frame are not coincident with the principal axes of the model.

PRBM -- A reference frame whose origin is at the cg of the model as calculated by the MASS Processor and which is oriented such that its axes coincide with the principal axes of the model. Generally, dynamic coupling will not exist for this case.

TRBM -- A reference frame whose origin and orientation are defined via the X, Y, Z, RnX, RnY and RnZ parameters discussed below. The resulting weight matrix for the model will, in general, contain nonzero static moments and products of inertia.

$\begin{Bmatrix} \text{URBM} \\ \text{PRBM} \\ \text{TRBM} \end{Bmatrix} = \text{xyzxyz}$	<p>The key-word specified by this parameter denotes which type of reference frame is to be used to reference (uncouple) the overall RB modes. The code <u>xyzxyz</u> is a 1 to 6 digit integer denoting which RB modes are inherent in the model. Each digit in this code is set to 0 or 1 denoting whether or not Y-axis translation, Y-axis translation, Z-axis translation, X-axis rotation, Y-axis rotation and Z-axis rotation, respectively, are present. A one indicates that the corresponding RB motion is present, whereas a zero indicates that it is not present. Left-most zeros of this code need not be input. One of these parameters must be selected if uncoupled RB-modes are to be generated. Only one of these parameters, however, may be specified in a single statement. Default: The RB modes as calculated by the VIBRATION Processor</p>
---	---

are components of the output
"MODESxx" matrix.

Example: This parameter might be input as
TRBM=10101 which indicates that Y-
translation, X-rotation and Z-rotation
are inherent to the model and that these
RB modes are to be uncoupled relative
to a user-selected reference frame.

The following two sets of parameters define the origin
and orientation of a user-selected rectangular reference frame
relative to which the RB modes are to be uncoupled. These
parameters are applicable only if the parameter TRBM=xyzxyz
is input.

$\langle \begin{array}{l} X=a \\ Y=b \\ Z=c \end{array} \rangle$

The GLOBAL X, Y and Z coordinates of
the origin of the user-selected, RB-
mode reference frame.
Default: Values are zero.

$\langle \begin{array}{l} RnX=a \\ RnY=b \\ RnZ=c \end{array} \rangle$

Rotations (degrees) used to orient the
user-selected, RB-mode reference frame
relative to the GLOBAL triad. One to
six of these three types of parameters
may be used in a single statement.
The sequence in which the rotations
are to be effected is denoted by the
integer assigned to "n" ($1 \leq n \leq 6$)
within the words RnX, RnY and RnZ.
The sequence defined by the "n" numbers
must begin with one and must be
continuous up to a maximum of 6 (e.g.,
R1Y=b1, R2X=a1, R3Y=b2, R4Y=b3, R5Z=c1).
The i-th sequential rotation is effected
relative to the X, Y or Z axis as
positioned after the (i-1) rotation.
Default: Values are zero.

Default: If neither of the last two sets of
parameters is specified when the TRBM
parameter is selected, the GLOBAL triad
is used as the RB-mode reference frame.

$$\text{HALF} = \begin{pmatrix} \text{XYx} \\ \text{YZx} \\ \text{XZx} \end{pmatrix}$$

This parameter denotes that the total-weight matrix (see discussion below) associated with the cg of the mass model is to be referenced to a GLOBAL plane of model symmetry. The characters XY (or YX), YZ (or ZY) or XZ (or ZX) denote the plane of symmetry. The letter "x" denotes a user node number (1 to 5 digits) which defines the location of the plane of symmetry. Orthographic projection of the cg of the half model onto the specified GLOBAL plane defines the cg of the symmetrical, total model. The weight matrix is referenced to this total-model cg point prior to performing the RB-mode replacement(s) (URBM, PRBM or TRBM parameter). Only one of these parameters may be specified in a single statement.

Default: The weight matrix terms are not modified to reflect a half-model of a symmetrical structure.

UNITR

Key-word denoting that the RB-replacement modes (URBM, PRBM or TRBM) are not to be normalized.

Default: Each RB-replacement vector is normalized to 1.0 based on its largest component.

RBGM

Key-word denoting that the partition of the generalized mass matrix which is associated strictly with the RB-replacement modes (URBM, TRBM or PRBM) is to be calculated as $R^T * J * P$ where, R is the matrix of RB-modes and J is the total-mass matrix.

Default: The generalized mass matrix is generated by using the reduced mass matrix identified by "Name1" and the calculated mode-shape matrix identified by MODESxx.

CONMASS=Num

The number (integer) of the concentrated mass subset defined via the mass data (sec. 138.0) which is to be associated with a Guyan-reduced mass matrix "Name1." This parameter is applicable only if URBM, PRBM or TRBM is specified. Default: No concentrated mass subset is associated with the mass matrix "Name1."

The total-weight matrix as calculated by the VIBRATION Processor, if one of the parameters URBM, PRBM or TRBM is used, represents the total-weight properties of the model relative to the selected reference frame. Its general form is

$$\begin{array}{ccc|ccc} W & 0 & 0 & 0 & W*z & -W*y \\ 0 & W & 0 & -W*z & 0 & W*x \\ 0 & 0 & W & W*y & -W*x & 0 \\ \hline 0 & -W*z & W*y & I_{xx} & I_{xy} & I_{xz} \\ W*z & 0 & -W*x & I_{xy} & I_{yy} & I_{yz} \\ -W*y & W*x & 0 & I_{xz} & I_{yz} & I_{zz} \end{array}$$

where W is the weight of the mass model and I_{ijk} denotes the weight products of inertia relative to the selected reference frame axes. x, y and z are mass-model cg coordinates relative to a TREM triad. The off-diagonal partition elements (W*z, etc.) introduced by a TRBM coordinate transformation are referenced as "static moments." These terms are zero in the total-weight matrix calculated by the MASS Processor (sec. 238.2). Masses are calculated from weights by using the mass factor specified in the MASS input data (see FACTOR data subset of sec. 138.0). If that factor is not specified, weights are divided by the default value of 386.04 to convert them to masses.

258.2 PRINT OUTPUT STATEMENT

The following statement is used to request printout of the data calculated by the VIBRATION Processor.

PRINT OUTPUT (VIBRATION, Plist)

List of optional parameters for Plist:

VSET=Num The vibration set number (integer) for which calculated data are to be printed.
Default: VSET=1

NO {Data} Key-word that suppresses printing of selected vibration data. One or more of the following key-words may be used in a single statement:

NOFREQS -- Frequencies
NOMODES -- Modes
NOGSTIF -- Generalized stiffness
NOGMASS -- Generalized mass

Default: All the vibration data are printed.

SUBSETS= $\left\{ \begin{array}{l} (Nxxx, Nyyy, \dots) \\ Nxxx \text{ TO } Nyyy \end{array} \right\}$

This parameter defines the node subsets for which subset mode-matrices, as generated by the VIBRATION Processor, are to be printed. Any combination of these options may be used within a single statement. The Asterisk Name Option (see sec. 200.0) may be used to identify subsets.

Default: No previously generated, subset mode-matrices are printed.

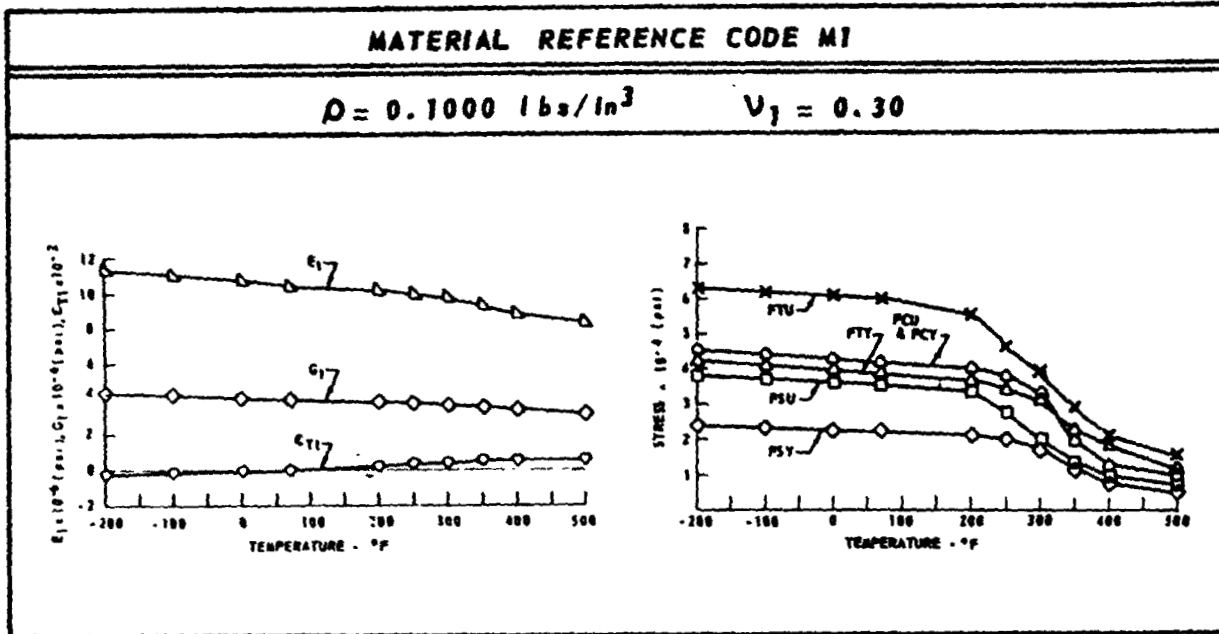
APPENDIX A

ATLAS STANDARD MATERIAL PROPERTY DATA

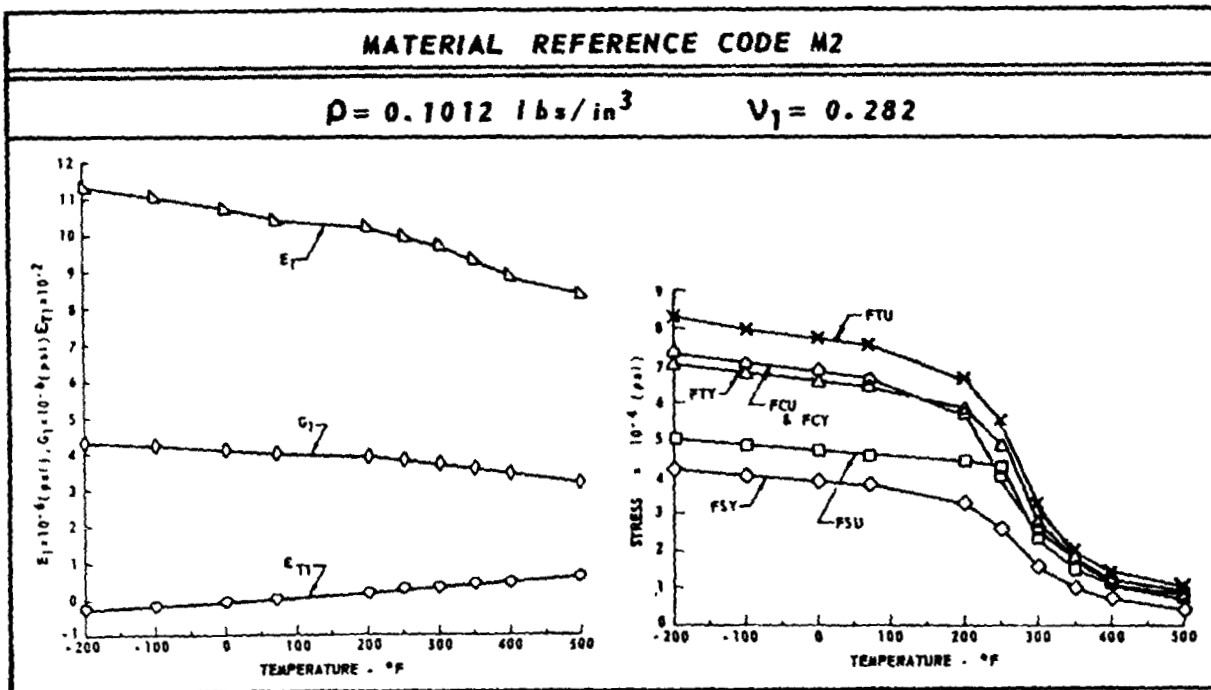
The data presented in this section illustrate the discrete property values for each of the linearly elastic, isotropic materials in the ATLAS standard-material catalog. Each standard material is identified by a material reference code (see sec. 140.0). The single set of nominal property data and forms selected for each alloy are representative of that material (ref. 140-1). The notation used herein is as follows:

E_1	=	Young's modulus (lbs/in. ²)
G_1	=	Shear modulus (lbs/in. ²)
ρ	=	Material density at 70°F (lbs/in. ³)
ν_1	=	Poisson's ratio
ϵ_{T1}	=	Linear thermal strain relative to 70°F
FTU	=	Ultimate tension stress (lbs/in. ²)
FCU	=	Ultimate compression stress (lbs/in. ²)
FSU	=	Ultimate shear stress (lbs/in. ²)
FTY	=	Yield tension stress (lbs/in. ²)
FCY	=	Yield compression stress (lbs/in. ²)
FSY	=	Yield shear stress (lbs/in. ²)

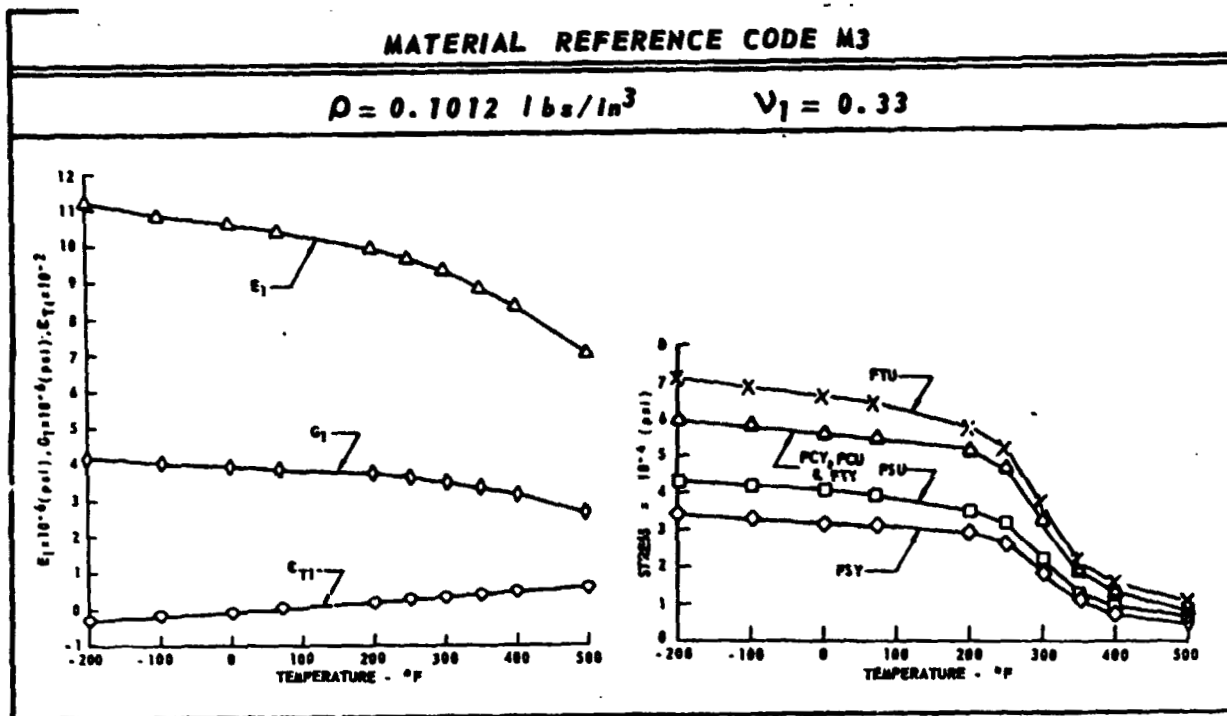
The stored values of E_1 , G_1 and ν_1 satisfy the expression $E_1 = 2G_1(1 + \nu_1)$ for isotropic materials.



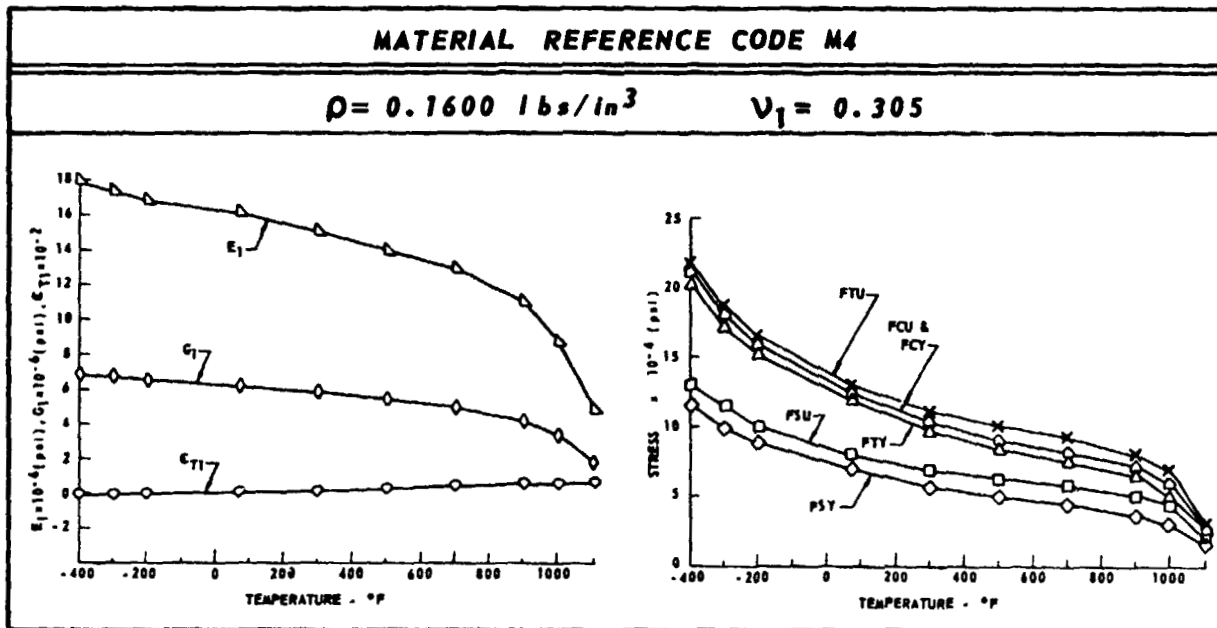
**2024-T351 ALUMINUM
CLAD PLATE (0.50 - 1.0 in.)**



**7075-T6 ALUMINUM
CLAD PLATE & SHEET (0.188 - 0.499 in.)**



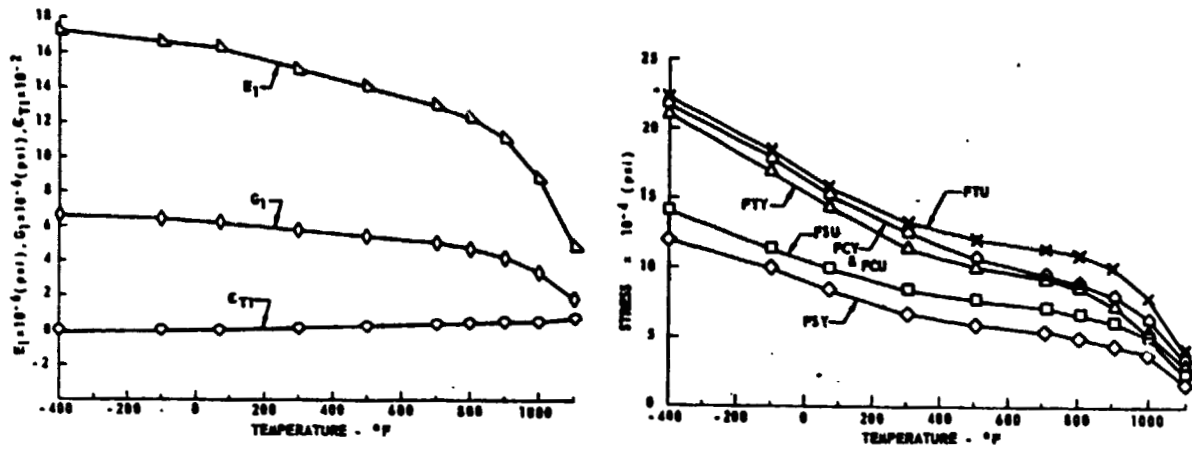
**7075-T73 ALUMINUM
FORGING ($\leq 3.0 \text{ in.}$)**



**6AL-4V ANNEALED TITANIUM
BAR & FORGING**

MATERIAL REFERENCE CODE M5

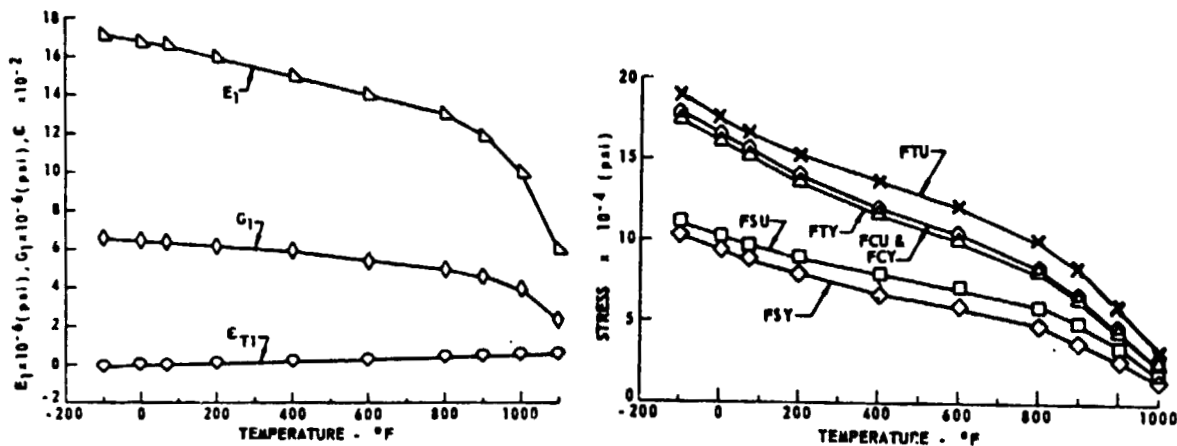
$\rho = 0.1600 \text{ lbs/in}^3$ $\nu_1 = 0.305$



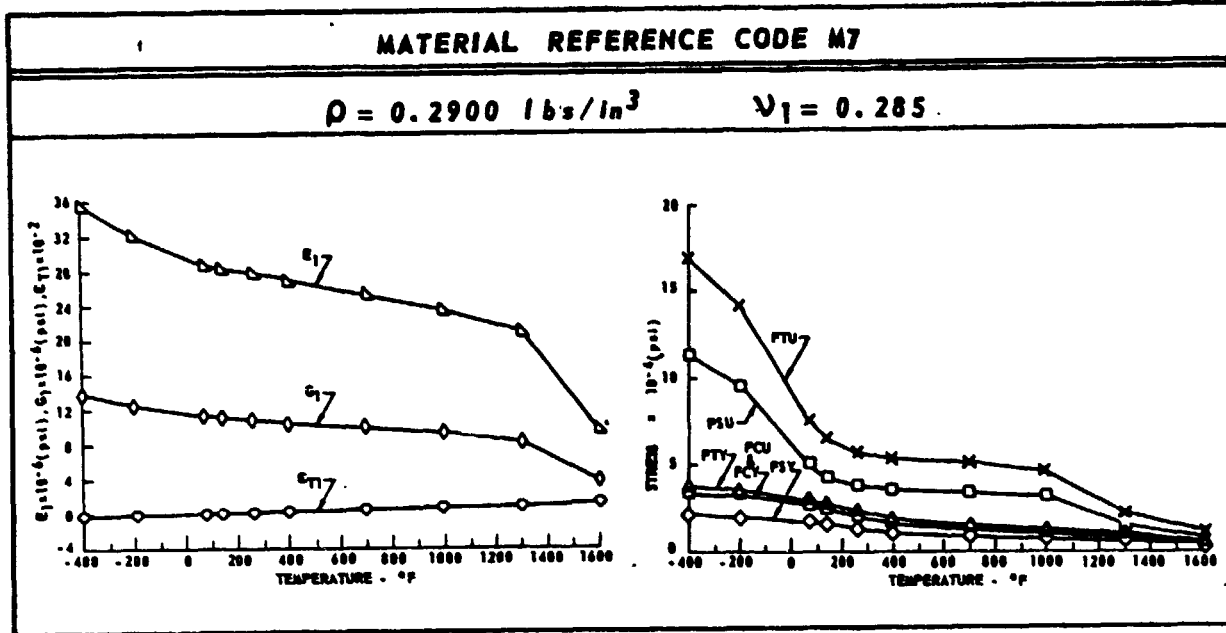
6AL-4V STA 1000 TITANIUM,
SHEET & PLATE ($\leq 0.750 \text{ in.}$)

MATERIAL REFERENCE CODE M6

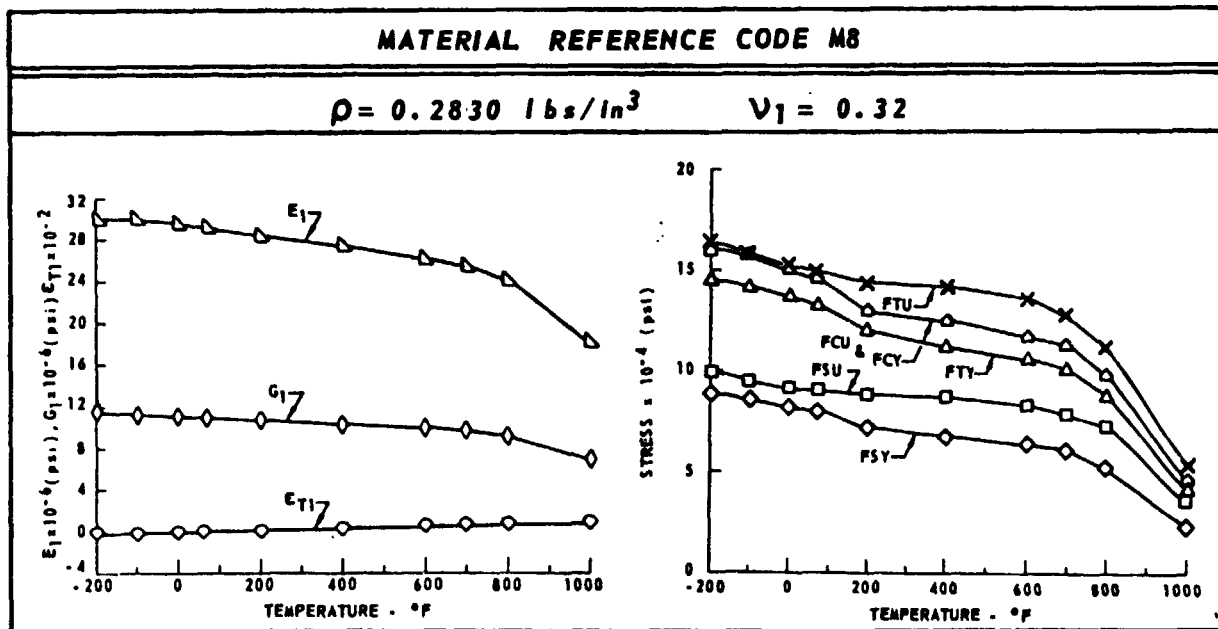
$\rho = 0.1640 \text{ lbs/in}^3$ $\nu_1 = 0.31$



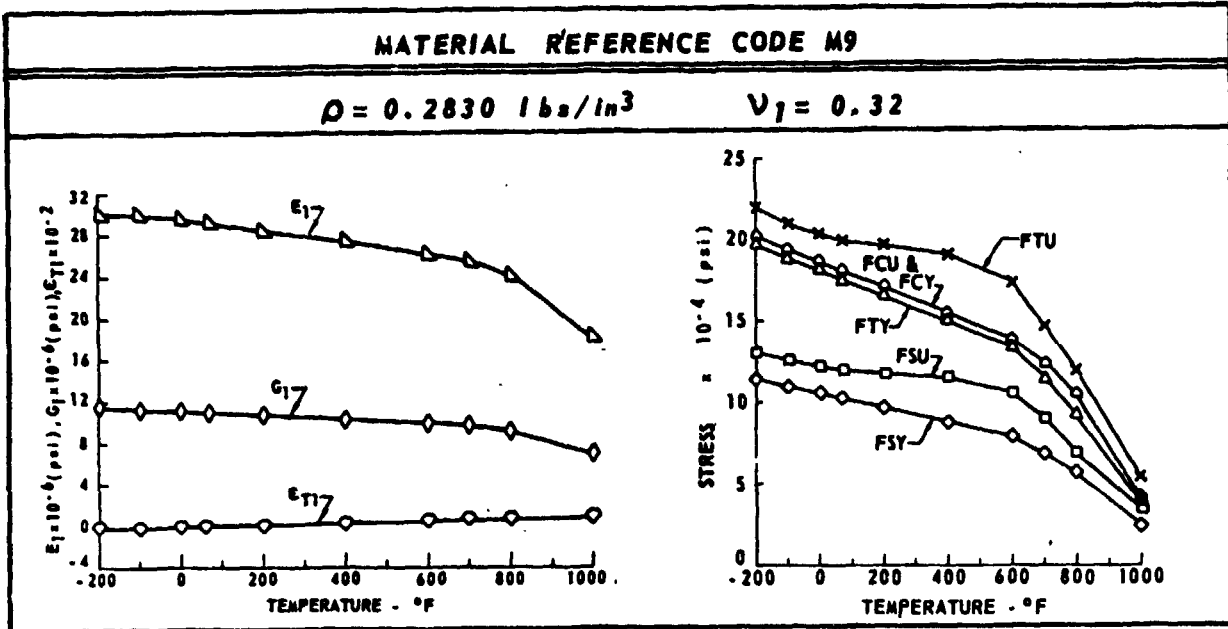
6AL-4V-2SN STA 1100 TITANIUM
FORGING ($\leq 2.0 \text{ in.}$)



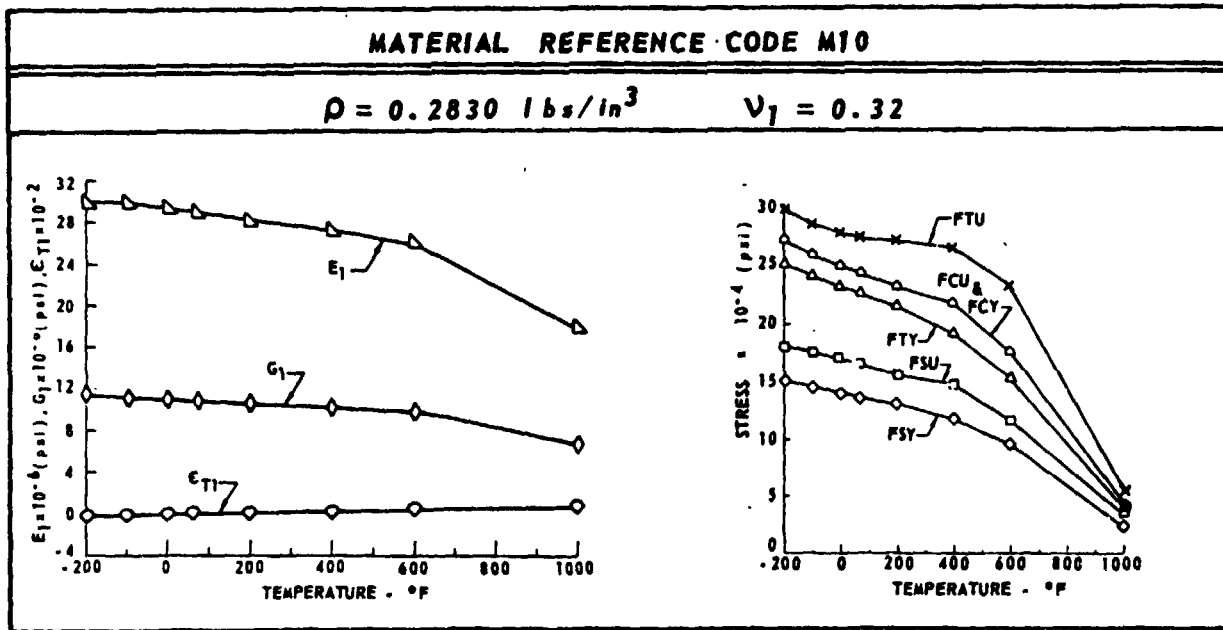
**AISI 321 ANNEALED STAINLESS STEEL
SHEET AND BAR**



**AISI 4130, 4135, OR 4140 STEEL (150-170 ksi H. T.)
ALL WROUGHT PRODUCTS**



AISI 4330M STEEL (200 ksi H.T.)
ALL WROUGHT PRODUCTS



AISI 4340M STEEL (275 KSI H.T.)
BAR & FORGING

APPENDIX B

ATLAS STIFFNESS FINITE-ELEMENT CATALOG

The following items are presented for each of the ATLAS stiffness finite elements:

- a) The key-word or integer-equivalent required for input identification;
- b) A general description of the element including a sketch;
- c) Description of nodes used to define the element;
- d) Definition of the element reference frame used to reference directions of input element loads and output stresses;
- e) Definition of the element section-properties and their alphanumeric identifiers;
- f) Material properties used for stiffness, stress and loads computations;
- g) Nodal input variations (combinations) and illegal conditions;
- h) Section-property input variations (combinations) and illegal conditions;
- i) Description of element loading capabilities;
- j) Description of element thermal loading capabilities;
- k) Definition of element stresses in force (F) and length (L) units.

Herein, lower case x , y and z denote right-handed, orthogonal, Cartesian element-reference-frame axes, whereas X - Y - Z denotes the nodal-data GLOBAL reference frame (sec. 146.0). Principal material axes are denoted by a 1-2-3 orthogonal triad.

Input variations of element section-properties are illustrated via tables of expansion keys. An expansion key is the column of integers associated with a particular number of input property values for the element type. A nonzero integer component of a key denotes which one of the ordered property values defined by an input data record is assigned to the

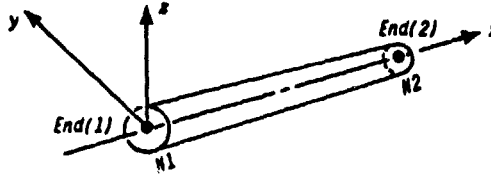
corresponding property shown in the table. A zero integer denotes that the property value is to be set to zero. An asterisk denotes that a property value is to be automatically assigned the corresponding value (specified or default) of the last defined element of the same type. Descriptions of the element-definition input data records are presented in section 152.0.

The element loading and element thermal loading capabilities described herein are the most general input variations provided. Other input variations are described in section 134.0.

STIFFNESS ROD ELEMENT

IDENTIFICATION: MOD or 1

GENERAL DESCRIPTION: Straight element with a linear area variation and a constant axial load. Uniform stress distribution on cross section. The geometric stiffness is based on a linear variation of displacements.



NODES:

N1 = Structural node located at centroid of cross section at end(1).

N2 = Structural node located at centroid of cross section at end(2).

ELEMENT REFERENCE FRAME: If x is parallel to Z, y is parallel to X. Otherwise, the x-y plane is parallel to Z such that the dot product $y \cdot Z > 0$.

SECTION PROPERTIES:

A(1) = Cross section area at end(1).

A(2) = Cross section area at end(2).

MATERIAL PROPERTIES:

- E1 for stiffness
- ET1 for thermal strain

NODAL INPUT: N1, N2

Illegal Condition: Element length less than 10^{-6}

SECTION PROPERTY INPUT:

PROPERTY LABEL	EXPANSION KEYS	
	NUMBER OF INPUT VALUES	
	1	2
A(1)	1	1
A(2)	1	2

Illegal Condition: A(1) or A(2) < 0.

ELEMENT LOADING: Linearly varying distributed load along length (F/L).

ELEMENT THERMAL LOADING: Linearly varying Δt along length.

STRESS OUTPUT:

P = Axial force (F)

P/A(1) = Axial stress at N1 (F/L^2)

P/A(2) = Axial stress at N2 (F/L^2)



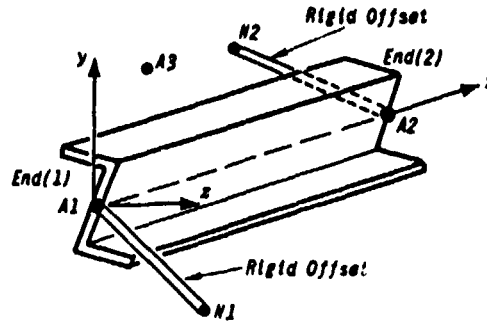
STIFFNESS BEAM ELEMENT

IDENTIFICATION:

BEAM OR 2

GENERAL DESCRIPTION:

Straight element based on Navier's theory of bending and St. Venant's theory of torsion (no warping). Cross section properties vary linearly. The shear center and the intersection point of the principal axes in bending coincide. The elastic axis can be offset from the structural nodes. The geometric stiffness is based on Navier's theory of bending.



NODES:

N1 = Structural node rigidly connected to end(1)
 N2 = Structural node rigidly connected to end(2)
 A1 = Auxiliary node defining elastic axis location at end(1)
 A2 = Auxiliary node defining elastic axis location at end(2)
 A3 = Auxiliary node defining the x-y principal plane

ELEMENT REFERENCE FRAME:

If A3 is specified, z is defined by the cross product of x and A1-A3. If A3 is not specified, z is defined by the cross product of x and GLOBAL Z. Note that if x is parallel to Z, A3 must be specified. The y and z-axes are principal axes in bending.

SECTION PROPERTIES:

A(1) = Cross section area at end(1)
 A-VY(1) = Effective shear area in y-direction at end(1)
 A-VZ(1) = Effective shear area in z-direction at end(1)
 J(1) = St Venant torsion constant at end(1)
 IY(1) = Moment of inertia about y-axis at end(1)
 IZ(1) = Moment of inertia about z-axis at end(1)
 A(2)
 A-VY(2)
 A-VZ(2) } Properties at end(2) analogous to those at end(1);
 J(2)
 IY(2)
 IZ(2)
 CONSTR = A twelve digit number (decimal) defining hinges, sockets and/or slides at A1 and/or A2 relative to the element reference frame. Each digit is a zero or a one corresponding to a particular element kinematic freedom. A one denotes that the corresponding force or moment in the element should be zero. The twelve digits of this number in a left-to-right order are associated with the element nodal freedoms Tz2, Ty2, Tx2, Tz1, Ty1, Tx1, Rz2, Ry2, Rx2, Rz1, Ry1 and Rx1, respectively. Thus, the value 100010.0 causes hinges to be formed relative to the Rz2 and Ry1 freedoms (i.e., the beam-end bending moments Mz(2) and My(1) in the stress output will be zero. A ball-and-socket joint at end(1) of a beam is formed by the value 111.0 for CONSTR.

MATERIAL PROPERTIES:

- E1 for axial deformation and bending
- G12 for transverse shear and torsion
- ϵ_{T1} for thermal strain in BEAM
- ϵ_{T3} for thermal strain in rigid offsets

NODAL INPUT:

- N1, N2, A1, A2, A3 (A1 may equal N1 and/or A2 may equal N2)
- N1, N2, A1, A2 (default element orientation)
- N1, N2, A3 (A1=N1, A2=N2)
- N1, N2 (A1=N1, A2=N2, default element orientation)

- Illegal Conditions:**
- Element length less than 10^{-6}
 - Distance from A1 to A3 less than 10^{-6}
 - $\sin^2 \alpha < 10^{-10}$ where α = angle between x and z or if A3 is defined, it is the angle between x and the line from A1 to A3

**SECTION PROPERTY
INPUT:**

PROPERTY LABEL	EXPANSION KEYS							
	NUMBER OF INPUT VALUES							
	1	2	3	4	6	7	12	13
A(1)	1	1	1	1	1	1	1	1
A-VY(1)	0	0	0	0	2	2	2	2
A-VZ(1)	0	0	0	0	3	3	3	3
J(1)	0	0	0	0	4	4	4	4
IY(1)	0	0	2	2	5	5	5	5
IZ(1)	0	2	3	3	6	6	6	6
A(2)	1	1	1	1	1	1	1	1
A-VY(2)	0	0	0	0	2	2	8	8
A-VZ(2)	0	0	0	0	3	3	9	9
J(2)	0	0	0	0	4	4	10	10
IY(2)	0	0	2	2	5	5	11	11
IZ(2)	0	2	3	3	6	6	12	12
CONSTR	0	0	0	4	0	7	0	13

A zero property value causes the corresponding stiffness/displacement term to be ignored.

- Illegal Conditions:**
- Any property value less than zero
 - Properties 1 through 6 or 7 through 12 all zero
 - Any of the first 12 properties non-zero at one end and zero at the other end
 - Effective shear area in the y-direction (z-direction) non-zero and IZ (IY) zero
 - CONSTR contains other digits than 0 or 1
 - Combinations of kinematic freedoms in CONSTR which allow rigid-body motion of the element (e.g., hinges for R_{x1} and R_{x2})

ELEMENT LOADING:

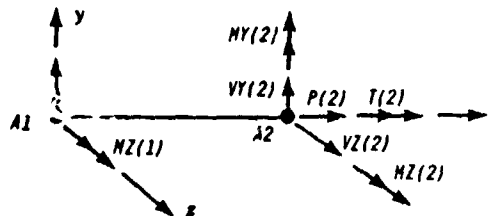
Linearly varying distributed load along length (F/L)

ELEMENT THERMAL LOADING:

Linearly varying Δt along length of BEAM. Rigid offset Δt 's vary linearly along offsets.

STRESS OUTPUT:

- P(2) = Axial force at end(2); (F)
- VY(2) = Shear force in y-direction at end(2); (F)
- VZ(2) = Shear force in z-direction at end(2); (F)
- T(2) = Torque at end(2); (FL)
- MY(1) = Bending moment about y-axis at end(1); (FL)
- MY(2) = Bending moment about y-axis at end(2); (FL)
- MZ(1) = Bending moment about z-axis at end(1); (FL)
- MZ(2) = Bending moment about z-axis at end(2); (FL)

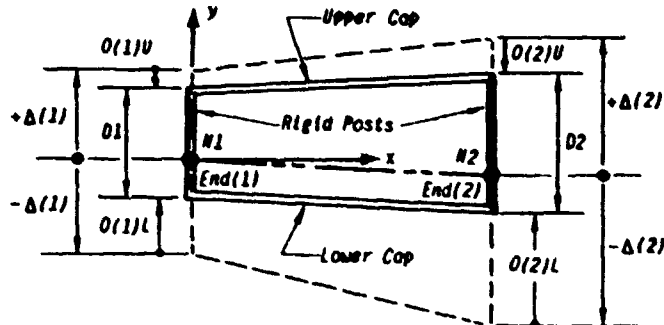


STIFFNESS SPAR ELEMENT

IDENTIFICATION: SPAR or 3

GENERAL DESCRIPTION: An element comprised of a quadrilateral shear web of uniform thickness, two caps with linear area variation and two rigid posts. Each cap is comprised of two components: a directly-defined flange area and a lumped web area. A uniform shear flow between caps and web is assumed.

Mid-surface nodes are required. Addition of the respective input Δ coordinates to the input nodal coordinates defines the upper part of the element, whereas subtraction defines the lower part (see sec. 146.0). Element depths $D1$ and $D2$ are defined by the Δ coordinates and the offset distances as shown below. When stiffnesses and loads are calculated, it is assumed that the rigid post at end (2) is parallel to the post at end (1).



NODES: N1 = Structural mid-surface node at end(1).
N2 = Structural mid-surface node at end(2).

ELEMENT REFERENCE FRAME: The y-axis is parallel to the rigid posts. Upper part of element has positive y-coordinates. The x-axis is in the plane defined by the cross product of y and N1-N2.

SECTION PROPERTIES:

- T-WEB = Web thickness
- FAREA1U = Upper flange area at end(1)
- FAREA1L = Lower flange area at end(1)
- FAREA2U = Upper flange area at end(2)
- FAREA2L = Lower flange area at end(2)
- O(1)U = Upper offset at end(1) - positive toward N1
- O(1)L = Lower offset at end(1) - positive toward N1
- O(2)U = Upper offset at end(2) - positive toward N2
- O(2)L = Lower offset at end(2) - positive toward N2
- A-LMP1U = Upper lumping factor at end(1)
- A-LMP1L = Lower lumping factor at end(1)
- A-LMP2U = Upper lumping factor at end(2)
- A-LMP2L = Lower lumping factor at end(2)

A lumping factor denotes how much of the web area is effective in carrying bending moments. For example, the area to be lumped at the N1 upper flange is calculated by $[(T-WEB) \cdot D1 \cdot (A-LMP1U)]$. If no lumping factors are specified for any defined SPAR element, they are automatically set to 1/6 by the stiffness data pre-processor. However, if they are specified for one element, these values are the default values for subsequently-defined SPARS. It should be noted that when stiffness element data are printed, the lumped areas (not the factors) are always displayed. Lumping factors (specified or default values) are used to generate SPAR element stiffness matrices only if the "LUMP" parameter is specified in the EXECUTE STIFFNESS statement (see sec. 252.2).

STIFFNESS ELEMENTS

MATERIAL PROPERTIES:

- E1 for flanges
- E2 for lumped web areas
- G12 for web
- ET1 for thermal strain in flanges
- ET2 for thermal strain in lumped areas
- ET3 for thermal strain in rigid posts

NORMAL INPUT:

N1, N2

- Illegal Conditions:
- Element length is less than 10^{-6}
 - N1 or N2 is not a mid-surface node
 - $\Delta(1)$ is zero
 - The depth at N1 or N2 is zero when a web of nonzero thickness is specified

SECTION PROPERTY INPUT:

PROPERTY LABEL	EXPANSION KEYS										
	NUMBER OF INPUT VALUES										
	1	2	3	5	6	7	9	10	11	13	
T-WEB	1	1	1	1	1	1	1	1	1	1	
FAREA1U	0	2	2	2	2	2	2	2	2	2	
FAREA1L	0	2	2	2	2	2	3	3	3	3	
FAREA2U	0	2	2	2	2	2	4	4	4	4	
FAREA2L	0	2	2	2	2	2	3	3	3	3	
O(1)U	0	0	3	4	4	4	6	6	6	6	
O(1)L	0	0	3	5	5	5	7	7	7	7	
O(2)U	0	0	3	4	4	4	8	8	8	8	
O(2)L	0	0	3	5	5	5	9	9	9	9	
A-LMP1U	•	•	•	•	•	6	•	10	10	10	
A-LMP1L	•	•	•	•	•	6	•	10	11	11	
A-LMP2U	•	•	•	•	•	6	•	10	10	12	
A-LMP2L	•	•	•	•	•	6	•	10	11	13	

* See discussion under "SECTION PROPERTIES"

- Illegal Conditions:
- Web thickness, any flange area or any lumping factor is less than zero.
 - One or both caps missing when web thickness is nonzero.
 - $D1 \leq 0$ or $D2 < 0$
 - All flange areas and web thickness zero
 - Zero flange area at one end when web thickness is zero

ELEMENT LOADING:

Linearly varying distributed load along length (F/L)

ELEMENT THERMAL LOADING:

Linearly varying Δt along each cap. Rigid post Δt 's vary linearly between upper and lower cap-end valves.

STRESS OUTPUT:

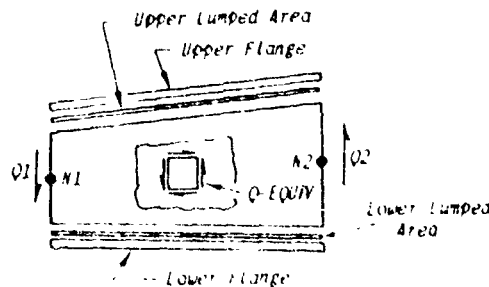
- P-CAPU = Average load in upper cap (F)
- SIGMA-U = Average stress in upper flange (F/L²)
- P-LMPU = Average load in upper lumped area (F)
- P-CAPL = Average load in lower cap (F)
- SIGMA-L = Average stress in lower flange (F/L²)
- P-LMPL = Average load in lower lumped area (F)
- Q-EQUIV = Equivalent shear flow (F/L)
- TAU-MAX = Maximum shear stress (F/L²)

The equivalent shear flow Q-EQUIV is defined as

$$Q-EQUIV = D2 \cdot Q2 / D1 + D1 \cdot Q1 / D2$$

which represents, in a certain sense, the average shear flow in the web.

Positive values of cap load, flange stress and lumped-area load indicate tension.

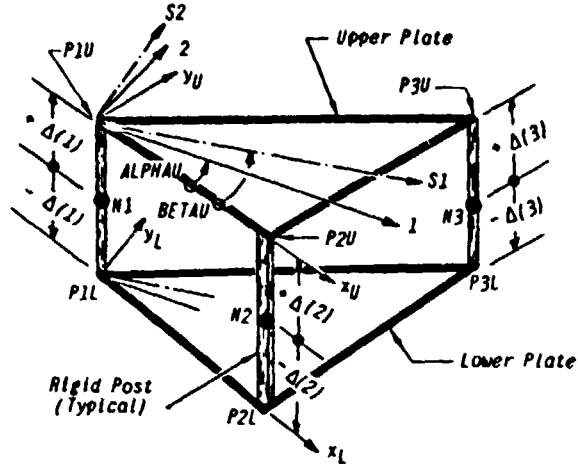


STIFFNESS COVER ELEMENT

IDENTIFICATION: COVER or 4

GENERAL DESCRIPTION:

Two triangular or quadrilateral ATLAS PLATE elements separated by rigid posts. One of the plates may have zero properties. Mid-surface nodes are required. Addition of the respective input Δ coordinates to the input nodal coordinates defines the upper plate corners, whereas subtraction defines the lower plate corners (see sec. 146.0). The directions of the rigid posts need not be parallel.



NODES:

N1, N2, N3 (and N4 for a quadrilateral COVER) are structural mid-surface nodes defining the element corners. Nodes defining a quadrilateral must be input sequentially in either the clockwise or counterclockwise direction.

ELEMENT REFERENCE FRAME:

One local frame, $x_U-y_U-z_U$, is defined for the upper plate and another frame, $x_L-y_L-z_L$, for the lower plate. Each local frame is defined relative to the plate corners, P1U, P2U, etc., in the same manner as the ATLAS PLATE element reference frame is defined.

SECTION PROPERTIES:

Upper Plate
 T(0)U = Plate thickness
 T(1)U = Smearred uniaxial thickness in 1-direction
 T(2)U = Smearred uniaxial thickness in 2-direction
 ALPHAU = Angle between x_U and material axis 1 (degrees)
 BETAU = Angle between x_U and stiffener direction S1 (degrees)

Lower Plate
 T(0)L
 T(1)L
 T(2)L
 ALPHAL
 BETAL
 } Definitions analogous to upper plate

MATERIAL PROPERTIES:

- E1, E2, V12 and G12 for basic plates
- E3 for stiffeners
- ϵT_1 and ϵT_2 for thermal strains in basic plate
- ϵT_3 for thermal strain in stiffeners
- Average of ϵT_1 and ϵT_2 for thermal strain in rigid posts.

NODAL INPUT:

- N1, N2, N3 (triangular COVER element)
- N1, N2, N3, N4 (quadrilateral COVER element)

- Illegal Conditions:**
- N1, N2, N3 or N4 is not a mid-surface node
 - The length of a plate edge is less than 10^{-6}
 - The upper or lower plate has a reentrant angle

**SECTION PROPERTY
INPUT:**

PROPERTY LABEL	EXPANSION KEYS							
	NUMBER OF INPUT VALUES							
	1	2	3	4	5	6	8	10
T(0)U	1	1	1	1	1	1	1	1
T(1)U	0	0	2	2	2	2	2	2
T(2)U	0	0	3	0	3	3	3	3
ALPHAU	0	0	0	0	4	0	0	4
BETAU	0	0	0	0	5	0	4	5
T(0)L	1	2	1	3	1	4	5	6
T(1)L	0	0	2	4	2	5	6	7
T(2)L	0	0	3	0	3	6	7	8
ALPHAL	0	0	0	0	4	0	0	8
BETAL	0	0	0	0	5	0	8	10

- Illegal Conditions:**
- Any thickness less than zero.
 - All properties equal to zero.

ELEMENT LOADING:

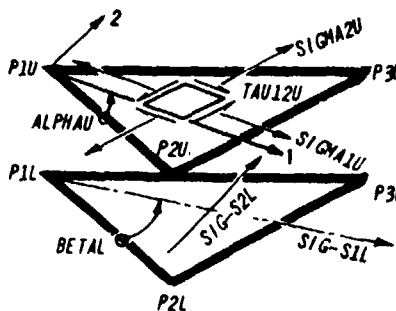
Linearly varying pressure loading may be applied to one or both of the plates (F/L^2)

ELEMENT THERMAL LOADING:

Linearly varying Δt may be applied to the plate components. Rigid post Δt 's vary linearly between upper and lower plate-corner values.

STRESS OUTPUT:

SIGMA1U	} Stress in upper plate (F/L^2)
SIGMA2U	
TAU12U	
SIG-S1U	} Axial stress in upper smeared stiffening material (F/L^2)
SIG-S2U	
SIGMA1L	
SIGMA2L	} Stress in lower plate (F/L^2)
TAU12L	
SIG-S1L	
SIG-S2L	} Axial stress in lower smeared stiffening material (F/L^2)

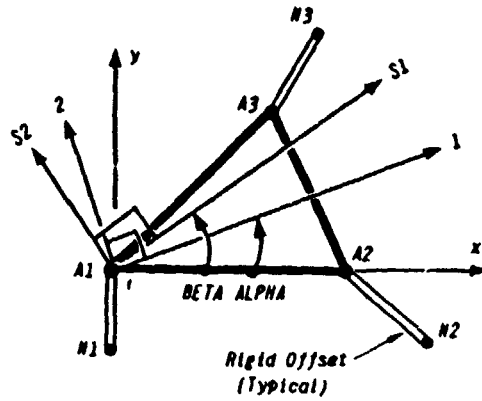


Note that the basic plate stresses are calculated relative to the material axes 1-2 as defined by ALPHAU and ALPHAL.

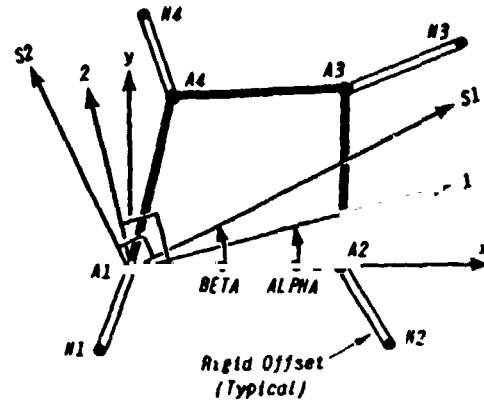
STIFFNESS PLATE ELEMENT

IDENTIFICATION: PLATE OR 5

GENERAL DESCRIPTION: Triangular or quadrilateral membrane element with orthotropic material capability and smeared uniaxial stiffening. The basic element is a constant-strain triangle. Quadrilateral PLATE stiffness is generated from four component triangles joined at a fifth internal node. If warped, the quadrilateral plate is equilibrated by transverse forces. The element may be offset from its structural nodes. The geometric stiffness is based on a linear variation of the membrane displacements.



Triangular PLATE



Quadrilateral PLATE

NODES: N1, N2, N3 (and N4 for a quadrilateral PLATE) are structural nodes rigidly attached to the element. A1, A2, A3 (and A4) are auxiliary nodes defining the location of the membrane. Nodes defining a quadrilateral plate N1-N4 (and A1-A4) must be input sequentially in either the clockwise or counterclockwise direction.

ELEMENT REFERENCE FRAME: The x-y plane is parallel to the plane defined by the mid-points of the plate edges. The origin is at A1 with the axes oriented such that the y-coordinate of A2 is zero and the y-coordinate of A3 is greater than zero.

SECTION PROPERTIES: T(0) = Plate thickness
 TS(1) = Smeared uniaxial thickness in S1 stiffener direction
 TS(2) = Smeared uniaxial thickness in S2 stiffener direction
 ALPHA = Angle between x and material axis 1 (degrees)
 BETA = Angle between x and stiffener direction S1 (degrees)

MATERIAL PROPERTIES:

- E1, E2, V12 and G12 for basic plate
- E3 for stiffeners
- ET1 and ET2 for thermal strains in basic plate
- ET3 for thermal strain in stiffeners
- Average of ET1 and ET2 for thermal strain in rigid offsets

NODAL INPUT:

- N1, N2, N3, N4, A1, A2, A3, A4 } { Ai may equal Ni and
- N1, N2, N3, A1, A2, A3 } { N1 may equal Ni
- N1, N2, N3, N4 } (A1 = N1)
- N1, N2, N3

Illegal Conditions:

- The length of an edge is less than 10^{-6}
- A quadrilateral plate has a reentrant angle

**SECTION PROPERTY
INPUT:**

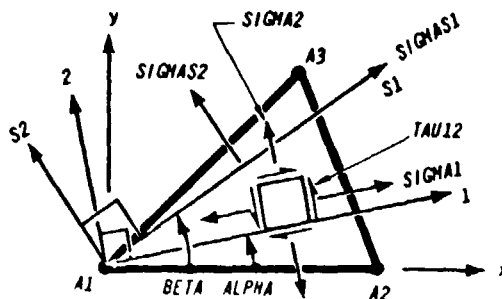
PROPERTY LABEL	EXPANSION KEYS				
	NUMBER OF INPUT VALUES				
	1	2	3	4	5
T(0)	1	1	1	1	1
TS(1)	0	0	2	2	2
TS(2)	0	0	3	2	3
ALPHA	0	2	0	3	4
BETA	0	0	0	4	5

Illegal Conditions: ● Any thickness less than zero
 ● All properties equal to zero

ELEMENT LOADING: Linearly varying pressure (F/L^2)

ELEMENT THERMAL LOADING: linearly varying Δt on surface. Rigid offset Δt 's vary linearly along offsets.

STRESS OUTPUT: SIGMA1 }
 SIGMA2 } Plate stresses (F/L^2)
 TAU12 }
 SIGMAS1 } Stiffener axial stresses (F/L^2)
 SIGMAS2 }



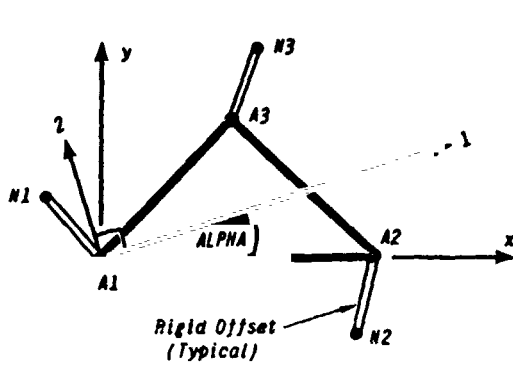
STIFFNESS GPLATE ELEMENT

IDENTIFICATION:

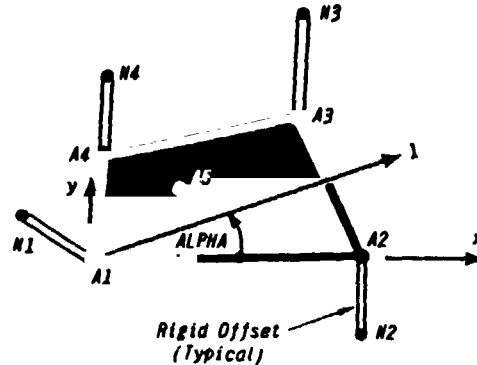
GPLATE or 6

GENERAL DESCRIPTION:

Triangular or quadrilateral plate element with uncoupled membrane and bending stiffnesses and orthotropic material capability. Triangular element assumes constant membrane strain and linear curvature. Quadrilateral element is composed of 4 linear membrane strain and linear curvature bending triangles. The element has no in-plane bending stiffness. It may be offset from its structural nodes. A quadrilateral element may be warped. The geometric stiffness for this element is identical to the one used for the ATLAS PLATE element.



Triangular GPLATE



Quadrilateral GPLATE

NODES:

N1, N2, N3 (and N4 for a quadrilateral GPLATE) are structural nodes rigidly attached to the element. A1, A2, A3 (A4) are auxiliary nodes defining the location of the membrane and neutral surface in bending. Nodes defining a quadrilateral plate N1-N4 (and A1-A4) must be input sequentially in either the clockwise or counterclockwise direction. A5, an optional auxiliary node, defines the point within a quadrilateral where the component triangles are joined.

ELEMENT REFERENCE FRAME:

The x-y plane is parallel to the plane defined by the mid-points of the plate edges. The origin is at A1 with the axes oriented such that the y-coordinate of A2 is zero and the y-coordinate of A3 is greater than zero.

SECTION PROPERTIES:

T-MEMB1 = Membrane thickness at A1
 T-MEMB2 = Membrane thickness at A2
 T-MEMB3 = Membrane thickness at A3
 T-MEMB4 = Membrane thickness at A4 for quadrilateral
 T-MEMB5 = Membrane thickness at A5 for quadrilateral

T-BEND1 = Bending thickness at A1
 T-BEND2 = Bending thickness at A2
 T-BEND3 = Bending thickness at A3
 T-BEND4 = Bending thickness at A4 for quadrilateral
 T-BEND5 = Bending thickness at A5 for quadrilateral

ALPHA = Angle between x and material axis 1 (degrees)

MATERIAL PROPERTIES:

- E1, E2, V12 and G12 for membrane and bending stiffness
- ET1 and ET2 for thermal strains in plate
- ET3 for thermal strain in rigid offsets

STIFFNESS ELEMENTS

NODAL INPUT:

Triangular GPLATE

- N1, N2, N3, A1, A2, A3 (A1 may equal N1 and N1 may equal Nj)
- N1, N2, N3, (A1 = N1)

Quadrilateral GPLATE

- N1, N2, N3, N4, A1, A2, A3, A4, A5 } { A1 may equal N1 and
- N1, N2, N3, N4, A1, A2, A3, A4 } { N1 may equal Nj
- (A5 located at average of corner coordinates)
- N1, N2, N3, N4, A5 (A1 = N1)
- N1, N2, N3, N4 (A1 = N1; A5 located at average of corner coordinates)

- Illegal Conditions:
- The length of an edge is less than 10^{-8}
 - A quadrilateral plate has a reentrant angle
 - A5 (if specified) lying outside the plate edges

SECTION PROPERTY
INPUT:

PROPERTY LABEL	EXPANSION KEYS										
	NUMBER OF INPUT VALUES										
	1	2	3	4	5	6	7	8	9	10	11
T-MEMB1	1	1	1	1	1	1	1	1	1	1	1
T-MEMB2	1	1	1	2	2	2	2	2	2	2	2
T-MEMB3	1	1	1	3	3	3	3	3	3	3	3
T-MEMB4	1	1	1	4	4	0	0	4	4	4	4
T-MEMB5	1	1	1	0	0	0	0	0	0	5	5
T-BEND1	1	2	2	0	0	4	4	5	5	6	6
T-BEND2	1	2	2	0	0	5	5	6	6	7	7
T-BEND3	1	2	2	0	0	6	6	7	7	8	8
T-BEND4	1	2	2	0	0	0	0	8	8	9	9
T-BEND5	1	2	2	0	0	0	0	0	0	10	10
ALPHA	0	0	3	0	5	0	7	0	9	0	11

- Illegal Conditions:
- Any thickness less than zero
 - All properties equal to zero

ELEMENT LOADING:

Linearly varying pressure (F/L^2)

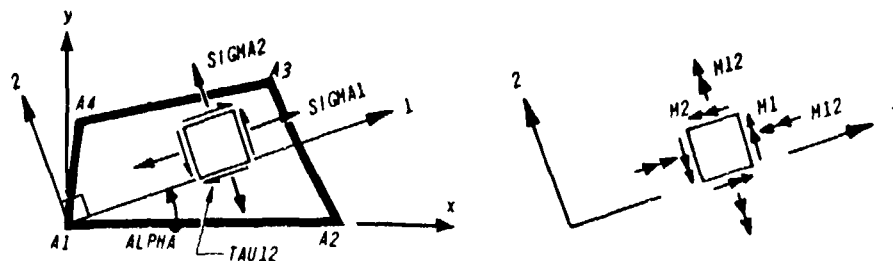
ELEMENT THERMAL LOADING:

Linearly varying Δt on surface. Rigid offset Δt 's vary linearly along offsets.

STRESS OUTPUT:

- | | | |
|--------|---|-------------------------------|
| SIGMA1 | } | Membrane stresses (F/L^2) |
| SIGMA2 | | |
| TAU12 | | |
| M1 | } | Bending moments (F) |
| M2 | | |
| M12 | | |

NOTE: Stresses and moments are averages of corner node values.



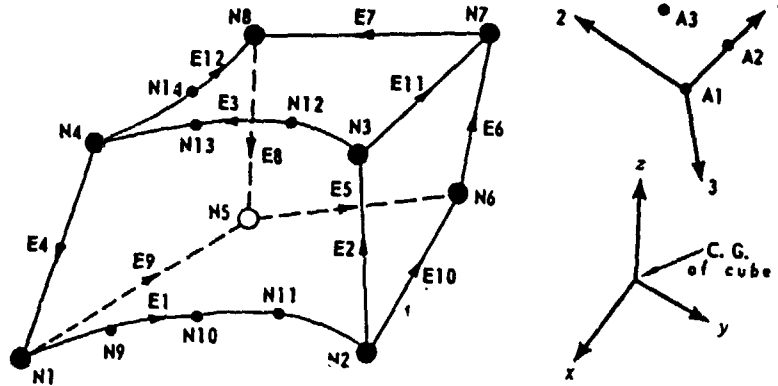
STIFFNESS BRICK ELEMENT

IDENTIFICATION:

BRICK or 8

GENERAL DESCRIPTION:

Isoparametric solid element with three degrees of freedom per node and an orthotropic material capability. The basic BRICK is defined by 8 corner nodes. These nodes define 12 edges (E1-E12) and directions thereof as shown in the figure. Along each edge, 1, 2, or 3 intermediate nodes may be specified. The geometry of these edges is defined by a second-, third- and fourth-, order polynomial, respectively. The geometric stiffness is based on the isoparametric representation of the displacements.



NODES:

N1-N8 = Structural nodes at corners of element
 N9-Nm = Structural nodes on edges of element where $m \leq 44$.
 A1-A3 = Auxiliary nodes defining principal directions of material orthotropy (optional).

The six faces of a BRICK are defined by their corner nodes as follows:

FACE	CORNER NODES
1	N1, N2, N3, N4
2	N5, N6, N7, N8
3	N2, N3, N6, N7
4	N1, N4, N5, N8
5	N3, N4, N7, N8
6	N1, N2, N5, N6

If the contour N1, N2, N3 and N4 is clockwise when viewed in the negative x-direction, face numbers 5 and 6 as shown in the table are interchanged.

ELEMENT REFERENCE FRAME:

Inherent in the analysis of the element behavior is a mapping of the geometry into a cube wherein the planes $x = \pm 1$, $y = \pm 1$ and $z = \pm 1$ define the element faces. The origin of the element reference frame is at the C.G. of the cube. The positive x-axis intersects face 1, the positive y-axis intersects face 3, and the z-axis is formed by the right-hand rule.

SECTION PROPERTIES:

There are no section properties associated with this element.

STIFFNESS ELEMENTS

MATERIAL PROPERTIES:

- The material may be orthotropic as defined by the properties $E_1, E_2, E_3, \nu_{12}, \nu_{31}, \nu_{23}, G_{12}, G_{13}$ and G_{23} . The material axes 1-2-3 are oriented by the auxiliary nodes A1-A3.
- $\epsilon_{T1}, \epsilon_{T2}$ and ϵ_{T3} are used for thermal strains

NODAL INPUT:

- $N_1, N_2, \dots, N_8, N_{E1}^p, N_{E2}^p, \dots, N_{E12}^p, A_1, A_2, A_3$, where N_1, N_2, \dots, N_8 are unique corner nodes which must always be input. N_1, \dots, N_4 and N_5, \dots, N_8 define two opposite sides of the BRICK. N_1, \dots, N_4 may be input clockwise or counter-clockwise, however, N_5, \dots, N_8 must be ordered so that each of the node pairs $(N_1, N_5), (N_2, N_6), (N_3, N_7)$ and (N_4, N_8) lies on a separate edge of the solid (see figure).
- $N_{E1}^p, N_{E2}^p, \dots, N_{E12}^p$: Intermediate edge nodes input in 12 groups of p nodes, where p is the maximum number of intermediate nodes on any one of the edges (i.e., $p = 0, 1, 2, 3$). The groups are ordered according to edge number and the items in each group are ordered according to the edge direction. If each edge does not have the same number of intermediate nodes, zeros must be input in the appropriate locations. For example, in the foregoing figure, $p = 3$ and the nodal input would be as follows:
 $N_1, N_2, \dots, N_8, N_9, N_{10}, N_{11}, 0, 0, 0, N_{12}, N_{13}, 0, 0, 0, 0, \dots, 0, 0, 0, N_{14}, 0, 0$
 - A_1, A_2, A_3 : Auxiliary nodes for defining material directions. Any one of these may be a zero indicating the GLOBAL origin. If not specified, the material axes 1-2-3 coincide with the GLOBAL X-Y-Z axes, respectively.

- Illegal Conditions:
- Any of the first 8 nodes equal to zero
 - Any edge having zero length
 - Improperly ordered intermediate edge nodes

SECTION PROPERTY INPUT:

None

ELEMENT LOADING:

Distributed loads may be applied to any face with a different value specified at each node on the loaded face.

ELEMENT THERMAL LOADING:

A Δt may be applied at each node.

STRESS OUTPUT:

Standard

The following components relative to the material reference frame at the centroid of the BRICK

SIGMA1	} Axial stresses (F/L ²)
SIGMA2	
SIGMA3	
TAU12	} Shear stresses (F/L ²)
TAU13	
TAU23	

Optional

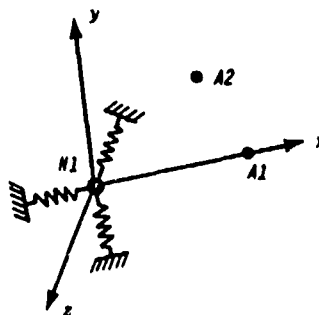
The same six stress components relative to the material reference frame at each node of the BRICK (see BIGBRICK parameter for the EXECUTE STIFFNESS statement, section 252.2).

STIFFNESS SCALAR ELEMENT

IDENTIFICATION: SCALAR or 9

GENERAL DESCRIPTION:

Spring element with three translational and three rotational stiffnesses at a point. The element restrains motion relative to ground.



NODES:

N1 = Structural node restrained by the SCALAR element.
A1, A2 = Auxiliary nodes defining directions of input stiffnesses.

ELEMENT REFERENCE FRAME:

Origin at N1 with the x-axis from N1 to A1. If A2 is specified, z is the vector product of x and N1-A2. If A2 is not specified, only a translation and a rotation stiffness may be defined relative to the x-axis. If A1 and A2 are not defined, the element reference frame coincides with the analysis frame of N1.

SECTION PROPERTIES:

K(T1) = Translational stiffness in x direction
K(T2) = Translational stiffness in y direction
K(T3) = Translational stiffness in z direction
K(R1) = Rotational stiffness about x axis
K(R2) = Rotational stiffness about y axis
K(R3) = Rotational stiffness about z axis

MATERIAL PROPERTIES:

None

NODAL INPUT:

- N1, A1, A2
- N1, A1 (stiffnesses relative to x-axis only)
- N1 (default element orientation)

SECTION PROPERTY INPUT:

PROPERTY LABEL	EXPANSION KEYS			
	NUMBER OF INPUT VALUES			
	1	2	3	6
K(T1)	1	1	1	1
K(T2)	0	0	2	2
K(T3)	0	0	3	3
K(R1)	0	2	0	4
K(R2)	0	0	0	5
K(R3)	0	0	0	6

**STIFFNESS
ELEMENTS**

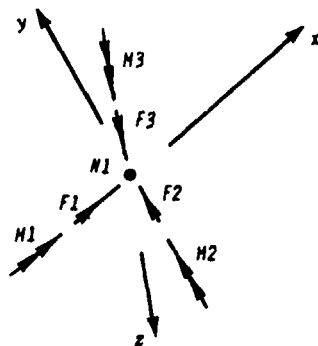
ELEMENT LOADING None

ELEMENT THERMAL LOADING: None

STRESS OUTPUT: **Forces and moments acting on the SCALAR from ground as measured relative to the element reference frame**

F1 }
F2 } Nodal forces (F)
F3 }

M1 }
M2 } Nodal moments (FL)
M3 }



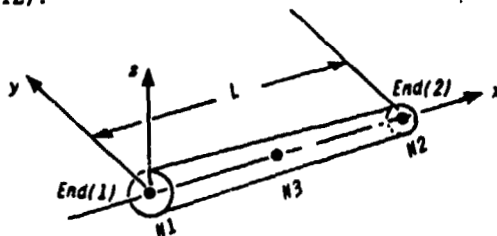
STIFFNESS SROD ELEMENT

IDENTIFICATION:

SROD or 10

GENERAL DESCRIPTION:

Straight element with a linear area variation and a linearly varying axial load. Uniform stress distribution on cross section with shear transfer at N3 to a compatible element (SROD or eight-node SPLATE).



NODES:

N1, N2 = Structural nodes located at centroid of cross section at end(1) and end(2), respectively.
 N3 = Structural shear node at which the total shear force is transferred to a compatible element or ground. The analysis frame of a shear node must be the GLOBAL triad.

ELEMENT REFERENCE FRAME:

If x is parallel to Z, y is parallel to X. Otherwise, the x-y plane is parallel to Z such that the dot product $y \cdot Z > 0$.

SECTION PROPERTIES:

A(1) = Cross section area at end(1)
 A(2) = Cross section area at end(2)

MATERIAL PROPERTIES:

- E1 for stiffness
- $\epsilon T1$ for thermal strain

NODAL INPUT:

N1, N2, N3

Illegal Condition: Element length less than 10^{-6}

SECTION PROPERTY INPUT:

PROPERTY LABEL	EXPANSION KEYS	
	NUMBER OF INPUT VALUES	
	1	2
A(1)	1	1
A(2)	1	2

Illegal Conditions: • One or both areas equal to zero
 • $A(1)$ or $A(2) < 0$.

ELEMENT LOADING:

Linearly varying distributed load along length (F/L).

ELEMENT THERMAL LOADING:

Linearly varying Δt along length.

STRESS OUTPUT:

P1 = Axial force at N1 (F)
 P2 = Axial force at N2 (F)
 P21/L = Shear flow along rod (F/L)
 (P/A)1 = Stress at N1 (F/L²)
 (P/A)2 = Stress at N2 (F/L²)



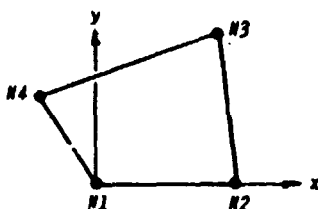
STIFFNESS SPLATE ELEMENT

IDENTIFICATION:

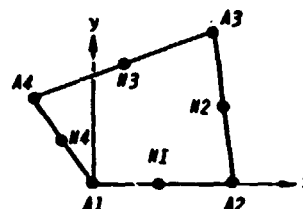
SPLATE or 11

GENERAL DESCRIPTION:

Constant thickness, four or eight-node quadrilateral shear panel. With eight nodes, shear transfer is at nodes N1-N4. With four nodes, shear is lumped at corner nodes. Equilibrated by transverse forces when warped. Eight-node SPLATE is compatible with the SROD element.



4-node SPLATE



8-node SPLATE

NODES:

N1-N4 = Structural nodes at which shear forces are transferred. These are shear nodes for an 8-node SPLATE. The analysis frame of a shear node must be the GLOBAL triad.
A1-A4 = Auxiliary nodes defining the corners of an 8-node SPLATE.

Nodes may be input in either the clockwise or counterclockwise direction.

ELEMENT REFERENCE FRAME:

The x-y plane is parallel to the plane defined by the mid-points of the plate edges. The origin is at N1(A1) with the axes oriented such that the y-coordinate of N2(A2) is zero and the y-coordinate of N3(A3) is greater than zero.

SECTION PROPERTIES:

T = Plate thickness

MATERIAL PROPERTIES:

G,2

NODAL INPUT:

- N1, N2, N3, N4 (4-node SPLATE)
- A1, A2, A3, A4, N1, N2, N3, N4 (8-node SPLATE)

- Illegal Conditions:
- The length of an edge is less than 10^{-6}
 - There is a reentrant angle

SECTION PROPERTY INPUT:

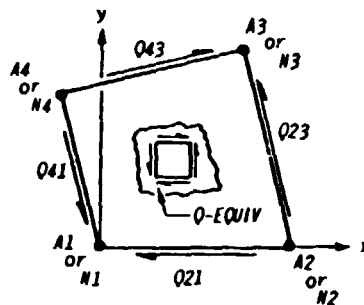
T

Illegal Condition: $T \leq 0$.

ELEMENT LOADING: Linearly varying distributed load (F/L^2)

ELEMENT THERMAL LOADING: None

STRESS OUTPUT: Q-EQUIV }
Q21 } Shear flows (F/L)
Q23 }
Q43 }
Q41 }
TAU-MAX = Maximum shear stress (F/L^2)
W-AVG = Average out-of-plane nodal load for warped plates (F).



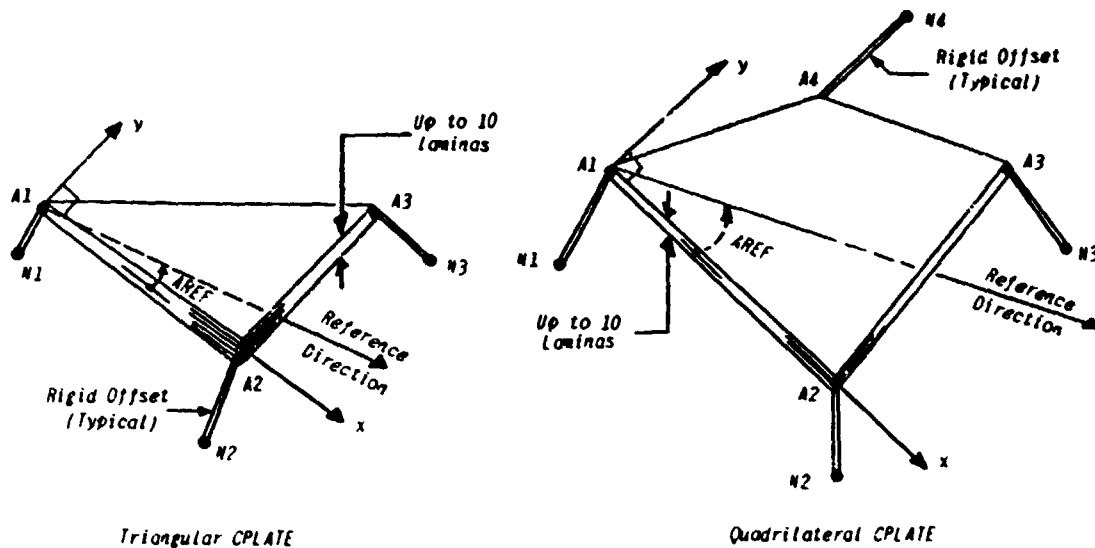
STIFFNESS CPLATE ELEMENT

IDENTIFICATION:

CPLATE or 12

GENERAL DESCRIPTION:

Triangular or quadrilateral laminated membrane plate element comprised of up to 10 orthotropic laminas. The triangular CPLATE is a constant strain element. The quadrilateral CPLATE stiffness is generated from the stiffnesses of four component triangles joined at an internal node. If warped, the quadrilateral plate is equilibrated by transverse forces. The element may be offset from its structural nodes.



NODES:

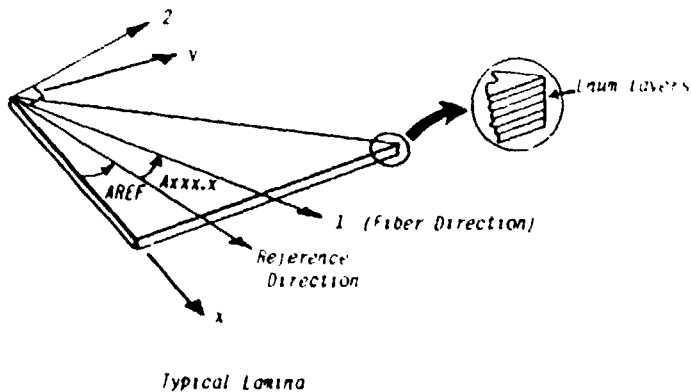
N1, N2, N3 (and N4 for a quadrilateral CPLATE) are structural nodes rigidly attached to the element. A1, A2, A3 (A4) are auxiliary nodes defining the location of the membrane. Nodes defining a quadrilateral plate N1-N4 (A1-A4) must be input sequentially in either the clockwise or counterclockwise direction.

ELEMENT REFERENCE FRAME:

The x-y plane is parallel to the plane defined by the mid-points of the plate edges. The origin is at A1 with the axes oriented such that the y-coordinate of A2 is zero and the y-coordinate of A3 is greater than zero.

SECTION PROPERTIES:

The CPLATE is comprised of one to ten laminas each of which is defined by four properties.



AREF = Angle defining the reference direction for the element (degrees)

The next four properties are specified for each lamina.

Axxx.x = Fiber angle (alphanumeric of the form Axxx.x). The numeric portion defines the fiber direction relative to the reference direction. $-360^{\circ} \leq xxx.x \leq 360^{\circ}$ with accuracy of $\pm 0.1^{\circ}$. This must be the first property input for each lamina.

Txxxx = Temperature deviation (alphanumeric of the form Txxxx). The numeric portion defines the lamina-temperature difference from the element temperature specified by "Tcode" in the element-definition record (sec. 152.0). $-2000^{\circ} \leq xxxxx \leq 4095^{\circ}$ with accuracy of $\pm 1^{\circ}$. Default: Temperature deviation of the previously defined lamina or 0° .

Lnum = Number of layers (alphanumeric of the form Lxxxx). The numeric portion defines the number of layers of composite material comprising the lamina. $1 \leq xxxxx \leq 4095$. Default: Number of layers of the previously defined lamina or 100.

Ccode = Composite material code (alphanumeric of the form Cxx). The numeric portion identifies the composite material of the lamina. $1 \leq xx \leq 31$. Default: Composite material code of the previously defined lamina or the composite material identified by "Mcode" in the element-definition record (sec. 152.0).

MATERIAL PROPERTIES:

- E1, E2, V12, G12 and t for stiffness of laminae
- ET1 and ET2 for thermal strains in laminae

NODAL INPUT:

- N1, N2, N3, N4, A1, A2, A3, A4 } { Ai may equal Ni and
- N1, N2, N3, A1, A2, A3 } { Ni may equal Nj
- N1, N2, N3, N4 } (Ai = Ni)
- N1, N2, N3 } (Ai = Ni)

- Illegal Conditions:
- The length of an edge is less than 10^{-6}
 - A quadrilateral plate has a reentrant angle

SECTION PROPERTY INPUT:

Properties are input in the order

AREF	}	Lamina 1
Axxx.x		
Txxxx		
Lnum		
Ccode	}	Lamina 2
Axxx.x		
Txxxx		
Lnum		
Ccode	}	Last Lamina
Axxx.x		
Txxxx		
Lnum		
Ccode		

Properties Txxxx, Lnum and Ccode may be defaulted for any lamina.

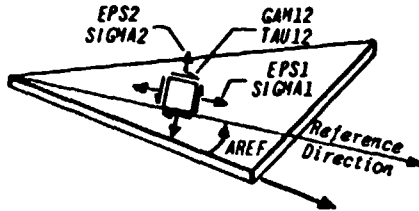
- Illegal Conditions:
- Fiber angle less than -360° or greater than 360°
 - Temperature deviation less than -2000° or greater than 4095°
 - Number of layers in a lamina less than 1 or greater than 4095

STIFFNESS ELEMENTS

ELEMENT LOADING: Linearly varying pressure (F/L^2)
ELEMENT THERMAL LOADING: Linearly varying Δt in each lamina. Rigid offset Δt 's vary linearly along offsets.

STRESS OUTPUT: Standard

EPS1	}	Total element strain (L/L)
EPS2		
GAM12		
SIGMA1	}	Total element stress (F/L^2)
SIGMA2		
TAU12		



Optional

Stress and/or strain components in each lamina measured relative to the fiber direction of the lamina (see sec. 254.0).

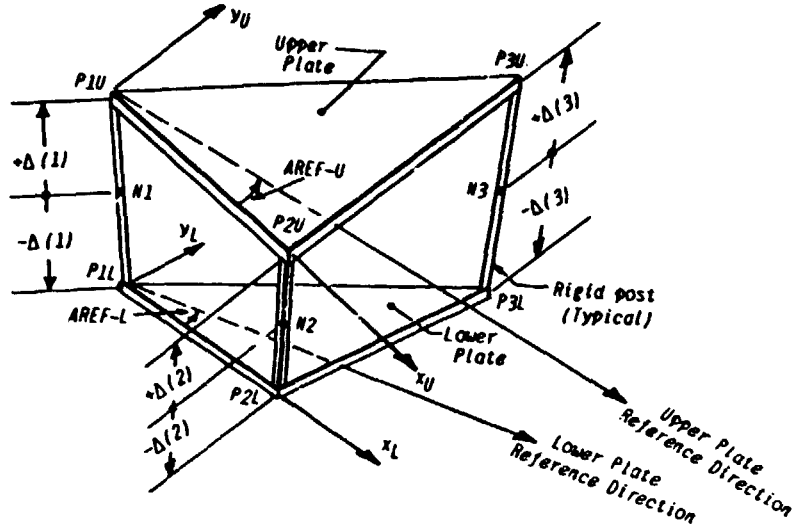
STIFFNESS CCOVER ELEMENT

IDENTIFICATION:

CCOVER or 13

GENERAL DESCRIPTION:

Two triangular or quadrilateral ATLAS CPLATE elements separated by rigid posts. Each plate may be comprised of up to ten orthotropic laminae. One of the plates may have zero properties. Mid-surface nodes are required. Addition of the respective input Δ coordinates to the input nodal coordinates defines the upper plate corners, whereas subtraction defines the lower plate corners (see sec. 146.0). The directions of the rigid posts need not be parallel.



NODES:

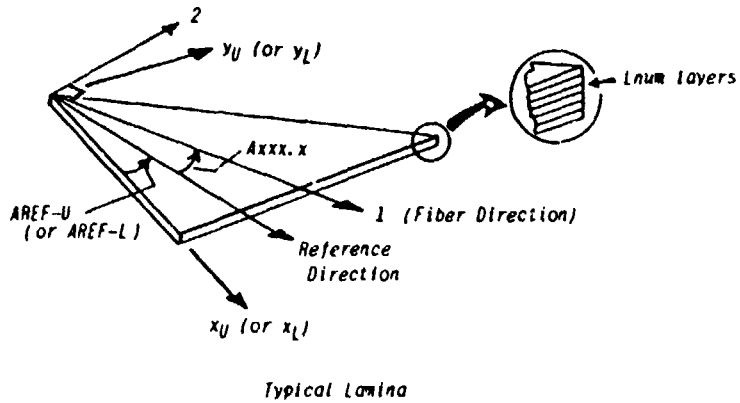
N1, N2, N3 (and N4 for a quadrilateral CCOVER) are structural mid-surface nodes defining the element corners. Nodes defining a quadrilateral must be input sequentially in either the clockwise or counterclockwise direction.

ELEMENT REFERENCE FRAME:

One local frame, $x_u - y_u - z_u$, is defined for the upper plate and another frame, $x_l - y_l - z_l$, for the lower plate. Each local frame is defined relative to the plate corners, P1U, P2U, etc., in the same manner as the ATLAS PLATE element reference frame is defined.

SECTION PROPERTIES:

Each plate is comprised of zero to ten laminae each of which is defined by four properties.



Upper Plate

AREF-U = Angle defining the reference direction for the upper plate (degrees). A value of AREF-U must be specified even if the CCOVER has no upper plate.

The next four properties are specified for each lamina of the upper plate. They are omitted if the CCOVER has no upper plate.

Axxx.x = Fiber angle (alphanumeric of the form Axxx.x). The numeric portion defines the fiber direction relative to the reference direction.
 $-360^\circ \leq xxx.x \leq 360^\circ$ with accuracy of $\pm 0.1^\circ$.
 This must be the first property input for each lamina.

Txxxx = Temperature deviation (alphanumeric of the form Txxxx). The numeric portion defines the lamina-temperature difference from the element temperature specified by "Tcode" in the element-definition record (sec. 152.0).
 $-2000^\circ \leq xxxx \leq 4095^\circ$ with accuracy of $\pm 1^\circ$.
 Default: Temperature deviation of the previously defined lamina or 0° .

Lnum = Number of layers (alphanumeric of the form Lxxxx). The numeric portion defines the number of layers of composite material comprising the lamina.
 $1 \leq xxxx \leq 4095$
 Default: Number of layers of the previously defined lamina or 100

Ccode = Composite material code (alphanumeric of the form Cxx). The numeric portion identifies the composite material of the lamina. $1 \leq xx \leq 31$
 Default: Composite material code of the previously defined lamina or the composite material identified by "Mcode" in the element-definition record (sec. 152.0).

Lower Plate

If the CCOVER has no lower plate, none of the following properties is input:

AREF-L }
 Axxx.x } Definitions analogous to upper plate
 Txxxx }
 Lnum }
 Ccode }

MATERIAL PROPERTIES:

- E1, E2, V12, G12 and t for stiffness of laminas.
- ET1 and ET2 for thermal strains in laminas

NODAL INPUT:

- N1, N2, N3 (triangular CCOVER)
- N1, N2, N3, N4 (quadrilateral CCOVER)

Illegal Conditions: • The length of an edge is less than 10^{-6}
 • A quadrilateral plate has a re-entrant angle

SECTION PROPERTY INPUT:

Properties are input in the order

AREF-U }
 Axxx.x } Lamina 1 of upper plate
 Txxxx }
 Lnum }
 Ccode }
 : }
 Axxx.x }
 Txxxx } Last lamina of upper plate
 Lnum }
 Ccode }


```

AREF-L
Axxx.x
Txxxx } Lamina 1 of lower plate
Lnum
Ccode
...
Axxx.x
Txxxx } Last lamina of lower plate
Lnum
Ccode

```

Properties Txxxx, Lnum and Ccode may be defaulted for any lamina.

- Illegal Conditions:
- Fiber angle less than -360° or greater than 360°
 - Temperature deviation less than -2000° or greater than 4095°
 - Number of layers in a lamina less than 1 or greater than 4095

ELEMENT LOADING:

Linearly varying pressure applied to one or both plates (F/L^2)

ELEMENT THERMAL LOADING:

Linearly varying Δt in each lamina of each plate. Rigid-post thermal strains vary linearly between upper and lower plate-corner values.

STRESS OUTPUT:

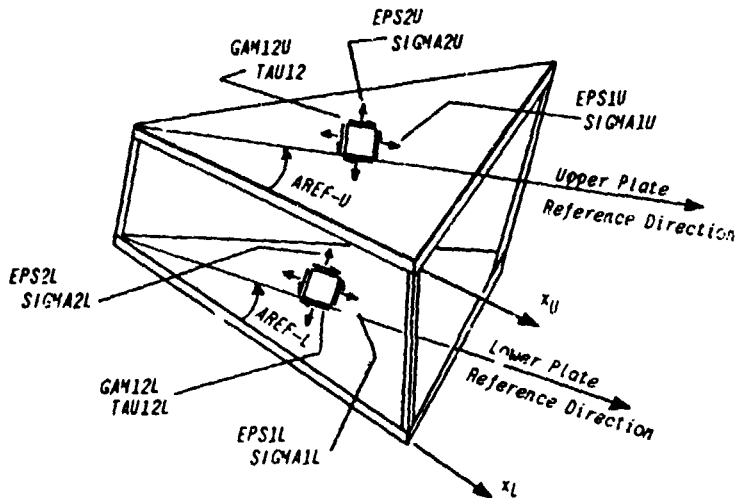
Standard

```

EPS1U }
EPS2U } Total strain in upper plate (L/L)
GAM12U }
SIGMA1U }
SIGMA2U } Total stress in upper plate (F/L2)
TAU12U }

EPS1L }
EPS2L } Total strain in lower plate (L/L)
GAM12L }
SIGMA1L }
SIGMA2L } Total stress in lower plate (F/L2)
TAU12L }

```



Optional

Stress and/or strain components in each lamina measured relative to the fiber direction of the lamina (see sec. 254.0).

APPENDIX C**ATLAS MASS FINITE-ELEMENT CATALOG**

The following items are presented for each of the mass finite elements available for structural and non-structural mass modeling and analysis via the ATLAS system.

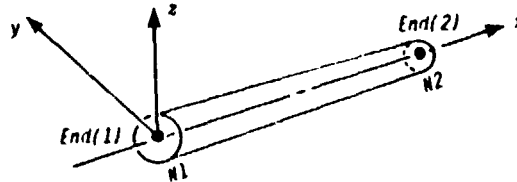
- a) The key-word or integer-equivalent required for input identification;
- b) A general description of the element including a sketch;
- c) Description of the nodes used to define the element;
- d) Definition of the element section-properties and their alphanumeric identifiers;
- e) Nodal input variations (combinations) and illegal conditions;
- f) Section and element mass-property input variations (combinations) and illegal conditions.

Input variations of the element mass-data properties are illustrated via tables of expansion keys. An expansion key is the column of integers associated with a particular number of input property values for the element type. A nonzero integer component of a key denotes which one of the ordered property values specified by an input data record is assigned to the corresponding property shown in the table. A zero integer denotes that the property value is to be set to zero.

MASS ROD ELEMENT

IDENTIFICATION: ROD or I

- GENERAL DESCRIPTION:**
- Thin
 - Linear area distribution
 - $I_{yy}(\text{local}) = I_{zz}(\text{local})$
 - $I_{xx}(\text{local}) = 0$



NODES:

N1 = Node located on centroid of cross section at end(1).
 N2 = Node located on centroid of cross section at end(2).

ELEMENT REFERENCE FRAME: If x is parallel to Z, y is parallel to X. Otherwise, the x-y plane is parallel to Z such that the dot product $y \cdot Z > 0$.

SECTION PROPERTIES:

A(1) = Cross-section area at end(1).
 A(2) = Cross-section area at end(2).
 NODE = A node in a plane that is perpendicular to the element. The intersection of this plane and the x-axis defines the element CG location.
 X = x-coordinate of CG on the element x-axis.

NODAL INPUT: N1, N2
 Illegal Condition: Element length less than 10^{-10}

SECTION PROPERTY INPUT:

F1- DENSITY PROPERTIES	EXPANSION KEYS	
	NO. INPUT VALUES	
	2	3
DENSITY	1	1
A1	2	2
A2	2	3

F2- WEIGHT PROPERTIES	EXPANSION KEYS		
	NO. INPUT VALUES		
	1	2	3
WEIGHT	1	1	1
A1	*	2	2
A2	*	2	3

F3- C. G. PROPERTIES	EXPANSION KEYS	
	NO. INPUT VALUES	
	2	
WEIGHT	1	1
NODE	2	2
X	2	2

*Default area is 1.0

Illegal Conditions:

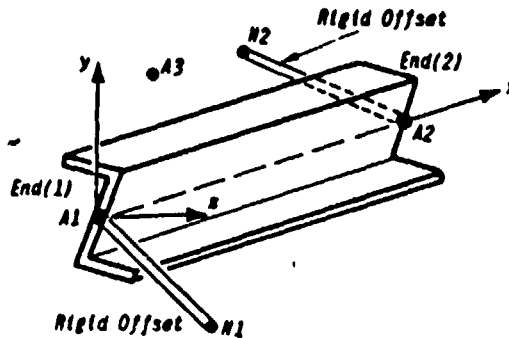
- Any property value less than zero
- Areas equal to zero when the F2-weight option is used
- CG location beyond the ends of the element

MASS BEAM ELEMENT

IDENTIFICATION: BEAM or 2

GENERAL DESCRIPTION:

- Linear cross-sectional property distribution
- Rigid offsets



NODES:

- N1 = Structural node rigidly connected to end(1)
- N2 = Structural node rigidly connected to end(2)
- A1 = Node defining the elastic axis location at end(1)
- A2 = Node defining the elastic axis location at end(2)
- A3 = Node defining the x-y principal plane

ELEMENT REFERENCE FRAME: If A3 is specified, z is defined by the cross product of x and A1-A3. If A3 is not specified, z is defined by the cross product of x and Z. Note that if x is parallel to Z, A3 must be specified.

SECTION PROPERTIES:

- A1 = Cross-section area at end(1)
- J1 = Area inertia about the element x-axis at end(1)
- IY1 = Area inertia about the element y-axis at end(1)
- IZ1 = Area inertia about the element z-axis at end(1)
- A2 } Properties at end(2) analogous to those at end(1)
- J2 }
- IY2 }
- IZ2 }

NODE = A node in a plane that is perpendicular to the element x-axis. The intersection of this plane and the x-axis defines the element CG location.

X = x-coordinate of CG on the element axis

NODAL INPUT:

- N1, N2, A1, A2, A3 (A1 may equal N1 and/or A2 may equal N2)
- N1, N2, A1, A2 (default element orientation)
- N1, N2, A3 (A1 = N1, A2 = N2)
- N1, N2 (A1 = N1, A2 = N2, default element orientation)

Illegal Condition: Element length less than 10^{-10}

**SECTION PROPERTY
INPUT:**

F1- DENSITY PROPERTIES	EXPANSION KEYS			
	NO. INPUT VALUES			
	2	3	5	9
DENSITY	1	1	1	1
A1	2	2	2	2
J1	**	**	3	3
IY1	**	**	4	4
IZ1	**	**	5	5
A2	2	3	2	6
J2	**	**	3	7
IY2	**	**	4	8
IZ2	**	**	5	9

F2- WEIGHT PROPERTIES	EXPANSION KEYS				
	NO. INPUT VALUES				
	1	2	3	5	9
WEIGHT	1	1	1	1	1
A1	*	2	2	2	2
J1	**	**	**	3	3
IY1	**	**	**	4	4
IZ1	**	**	**	5	5
A2	*	2	3	2	6
J2	**	**	**	3	7
IY2	**	**	**	4	8
IZ2	**	**	**	5	9

F3- C. G. PROPERTIES	EXPANSION KEYS	
	NO. INPUT VALUES	
	2	
WEIGHT	1	1
NODE	2	
X		2

* Default area is 1.0

** Computed value is based on a circular cross section

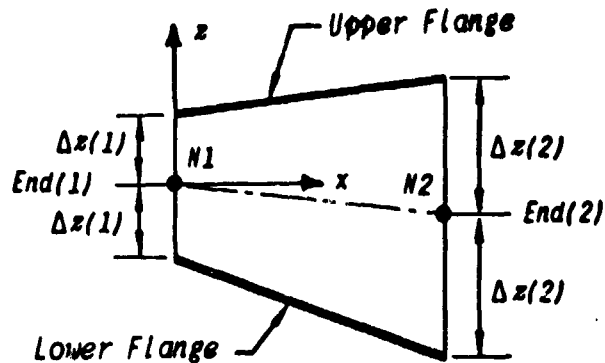
Illegal Conditions:

- Any property value less than zero
- Area equal to zero when the F2-weight option is used
- CG location beyond the ends of the element

MASS SPAR ELEMENT

IDENTIFICATION: SPAR or 3

- GENERAL DESCRIPTION:**
- Linear depth variation
 - Constant thickness web
 - Linear flange area variation
 - Mid-surface nodes required



NODES:

$N1$ = Mid-surface node at end(1)
 $N2$ = Mid-surface node at end(2)

SECTION PROPERTIES:

t -web = Web thickness
 $A1-U$ = Upper flange area at $N1$
 $A1-L$ = Lower flange area at $N1$
 $A2-U$ = Upper flange area at $N2$
 $A2-L$ = Lower flange area at $N2$

NODAL INPUT: $N1, N2$ (Mid-surface nodes)

Illegal Conditions :

- Element length less than 10^{-10}
- SPAR depth less than 10^{-8}
- Web is warped
- $N1$ or $N2$ is not a mid-surface node

**SECTION PROPERTY
INPUT:**

F1- DENSITY PROPERTIES	EXPANSION KEYS		
	NO. INPUT VALUES		
	3	4	6
t -web	1	1	1
DENSITY	2	2	2
$A1-U$	3	3	3
$A1-L$	3	4	4
$A2-U$	3	3	5
$A2-L$	3	4	6

NOTE: Input options F2 and F3 are not available for this element.

Illegal Condition :
Any property value less than zero.

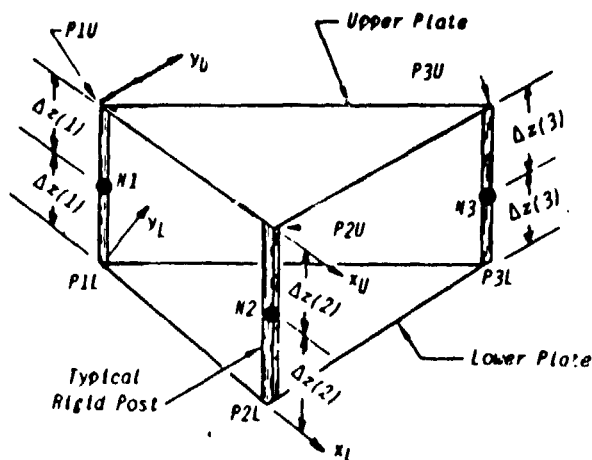
MASS COVER ELEMENT

IDENTIFICATION:

COVER or 4

GENERAL DESCRIPTION:

- Two PLATE elements with 3 to 9 edges separated by rigid posts
- Mid-surface nodes required



NODES:

N1, N2, N3 (and N4, N5, . . . , N9 for general COVERS) are mid-surface nodes defining the element corners. Nodes defining a general COVER must be input sequentially in either a clockwise or counterclockwise direction.

ELEMENT REFERENCE FRAME:

One local frame $x_U-y_U-z_U$ is defined for the upper plate and one frame $x_L-y_L-z_L$ for the lower plate. Each frame is defined relative to the plate corners P_{iU} , P_{2U} , . . . , as illustrated in the figure, in the same manner as the ATLAS PLATE element reference frame is defined.

SECTION PROPERTIES:

t-U = Upper plate thickness
 t-L = Lower plate thickness

NODE-U = A node on a line that is perpendicular to the upper plate. The intersection of this line and the upper plate defines the upper-plate CG location.

X-U = x-coordinate of the upper-plate CG relative to the $x_U-y_U-z_U$ reference frame

Y-U = y-coordinate of the upper-plate CG relative to the $x_U-y_U-z_U$ reference frame

NODE-L }
 Y-L } Properties for the lower plate analogous to those
 Y-L } for the upper plate

**MASS
ELEMENTS**

NODAL INPUT:

$N_1, N_2, N_3, \dots, N_m$ (Mid-surface nodes where $m \leq 9$)

COVER elements must be triangular or quadrilateral when **OPTION=4** of the MASS Processor is used.

Illegal Conditions :

- N_1, N_2, \dots, N_9 is not a mid-surface node
- Cover depth less than 10^{-8}
- Reentrant angle
- Warped upper or lower plate component
- Length of any edge less than 10^{-10}

**SECTION PROPERTY
INPUT:**

F1- DENSITY PROPERTIES	EXPANSION KEYS		
	NO. INPUT VALUES		
	2	3	4
DENSITY-U	1	1	1
DENSITY-L	1	1	2
t-U	2	2	3
t-L	2	3	4

F2- WEIGHT PROPERTIES	EXPANSION KEYS			
	NO. INPUT VALUES			
	1	2	3	4
WEIGHT-U	1	1	1	1
WEIGHT-L	1	2	2	2
t-U	*	*	3	3
t-L	*	*	3	4

F3- C. G. PROPERTIES	EXPANSION KEYS			
	NO. INPUT VALUES			
	4	5	6	
WEIGHT-U	1	1	1	1
WEIGHT-L	2	2	2	2
NODE-U	3	3		
X-U			3	3
Y-U			4	4
NODE-L	4		5	
X-L		4		5
Y-L		5		6

* Default thickness is 1.0

Illegal Conditions :

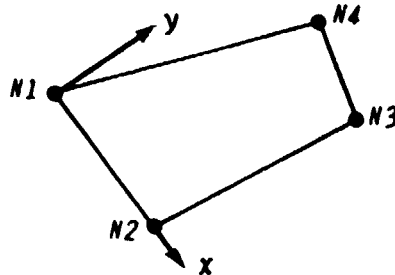
- Any property value less than zero
- Thickness equal to zero when the F2-weight option is used
- CG location outside a plate boundary

C-8

MASS PLATE ELEMENT

IDENTIFICATION: PLATE or 5

GENERAL DESCRIPTION: Three to nine edges



NODES: N1, N2, N3, N4 (and N5, . . . , N9) are nodes defining the element corners. Nodes must be input sequentially in either a clockwise or counterclockwise direction.

ELEMENT REFERENCE FRAME: The x-y plane is parallel to the element. The origin is at N1 and the axes are oriented such that the y-coordinate of N2 is zero and the y-coordinate of N3 is greater than zero.

SECTION PROPERTIES:

- t = thickness
- NODE = A node on a line that is perpendicular to the plate. The intersection of this line and the plate defines the element CG location.
- X = x-coordinate of CG relative to the element reference frame
- Y = y-coordinate of CG relative to the element reference frame

NODAL INPUT: N1, N2, N3, . . . , Nm where $m \leq 9$
PLATE elements must be triangular or quadrilateral when OPTION = 4 of the MASS Processor is used.

Illegal Conditions:

- Length of an edge less than 10^{-10}
- Reentrant angle
- Warped element

**MASS
ELEMENTS**

**SECTION PROPERTY
INPUT:**

F1-DENSITY	EXPANSION KEY	
	NO. INPUT VALUES	
	2	
DENSITY	1	
t	2	

F2-WEIGHT PROPERTIES	EXPANSION KEYS	
	NO. INPUT VALUES	
	1	2
WEIGHT	1	1
t	*	2

F3-C.G. PROPERTIES	EXPANSION KEYS	
	NO. INPUT VALUES	
	2	3
WEIGHT	1	1
NODE	2	.
X	.	2
Y	.	3

* Default thickness is 1.0

Legal Conditions :

- Any property value less than zero
- Thickness equal to zero when the F2-weight option is used
- CG location outside the plate boundary

MASS SCALAR ELEMENT

IDENTIFICATION: SCALAR or 9

GENERAL DESCRIPTION: Six degree-of-freedom concentrated mass element

NODES: N1 is a node that defines the CG of the element

ELEMENT PROPERTIES: WEIGHT
I = XXX The local coordinate system relative to which the element inertia values are defined.
 (Default = GLOBAL)

I = -- The characters *I* =
 XXX -- The name of a local coordinate system defined by the Nodal data

IXX	}	
IYY	}	
IZZ	}	
IXY	}	
IXZ	}	
IYZ	}	

 The weight moments of inertia of the element about a local x-y-z right-handed Cartesian system with its origin at N1 and an orientation defined by *I* = XXX. The products of inertia, IXY, IXZ, IYZ, are defined as follows:

$$I_{ij} = -\rho \int x_i x_j dv.$$

NODAL INPUT: N1

ELEMENT PROPERTY INPUT:

F2- WEIGHT PROPERTIES	EXPANSION KEYS						
	NO. INPUT VALUES						
	1	2	3	4	5	6	7
WEIGHT	1	1	1	1	1	1	1
<i>I</i> = XXX	-	-	-	-	-	-	-
IXX	0	2	2	2	2	2	2
IYY	0	0	3	3	3	3	3
IZZ	0	0	0	4	4	4	4
IXY	0	0	0	0	5	5	5
IXZ	0	0	0	0	0	6	6
IYZ	0	0	0	0	0	0	7

NOTE: Input options F1 and F3 are not available for this element

Illegal Conditions:

- Weight less than zero
- Local coordinate system for inertias is not rectangular.

APPENDIX D

SUBSTRUCTURE ANALYSIS VIA ATLAS

Substructure analyses via the ATLAS System are most conveniently performed by use of the catalogued Control Program statements presented in appendix E. The matrix operations performed via these control procedures are referenced within this appendix which describes the method of solution of the substructure equations.

The static-equilibrium matrix equation associated with a typical substructure can be written as:

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix} * \begin{bmatrix} D_{11} \\ D_{21} \\ D_{31} \end{bmatrix} = \begin{bmatrix} L_{11} \\ L_{21} \\ L_{31} \end{bmatrix} + \begin{bmatrix} 0 \\ R_{21} \\ R_{31} \end{bmatrix} \quad (1)$$

(ref. SS-MERGE (STIFFNESS) and SS-MERGE (LOADS)), where the following nomenclature has been adopted:

- Row 1 = The FREE nodal freedoms--those which are not to be interacted.
- Row 2 = The RETAINED nodal freedoms--those which are retained for generation of a reduced stiffness matrix. Freedoms which are interacted are included in this group.
- Row 3 = The SUPPORTED nodal freedoms--those which have user-specified (zero or nonzero) displacements.
- K_{ij} = The i-j stiffness matrix partition.
- D_{ij} = The i-j displacement matrix. D₃₁ is known, whereas D₁₁ and D₂₁ are unknown.
- L_{ij} = The i-j known applied loads matrix.
- R_{ij} = The i-j unknown reaction loads matrix.

Equation (1) can be expanded as follows:

$$\begin{aligned} K_{11}*D_{11} + K_{12}*D_{21} + K_{13}*D_{31} &= L_{11} & (2) \\ K_{21}*D_{11} + K_{22}*D_{21} + K_{23}*D_{31} &= L_{21} + R_{21} & (3) \\ K_{31}*D_{11} + K_{32}*D_{21} + K_{33}*D_{31} &= L_{31} + R_{31} & (4) \end{aligned}$$

Solution of eq. (2) for D₁₁ yields:

$$D11 = K11^{-1} [L11 - K12*D21 - K13*D31] \quad (5)$$

Substituting this into eq. (3) yields:

$$R21 = [K22 - K21*K11^{-1}*K12]D21 + K21*K11^{-1}[L11 - K13*D31] + K23*D31 - L21 \quad (6)$$

$$\text{or } R21 = KRED*D21 - LRED \quad (7)$$

$$\text{where } KRED = K22 - K21*K11^{-1}*K12 \quad (8)$$

$$\text{and } LRED = L21 - K21*K11^{-1}[L11 - K13*D31] - K23*D31 \quad (9)$$

Decompose K11 such that

$$K11 = L*L^T \quad (10a)$$

$$\text{and } K11^{-1} = (L^T)^{-1}*L^{-1} = (L^{-1})^T *L^{-1} \quad (10b)$$

A forward-substitution pass on K12 (ref. SS-REDU(STIFFNESS) and SS-REDU(LOADS)) yields:

$$A = L^{-1}*K12 \quad (11)$$

such that the reduced stiffness matrix can be expressed as

$$KRED = K22 - A^T *A \quad (12)$$

(ref. SS-REDU (STIFFNESS)).

Defining

$$B = L11 - K13*D31 \quad (13)$$

(ref. SS-REDU (LOADS)) and performing a forward-substitution pass on B yields

$$C = L^{-1}*B \quad (14)$$

such that the reduced loads matrix can be expressed as

$$LRED = (L21 - K23*D31) - A^T *C \quad (15)$$

The following matrix is required for subsequent back-substitution solutions (ref. SS-REDU (LOADS)) to support stress analysis

$$E = K33*D31 - L31 \quad (16)$$

A Guyan-reduced mass matrix associated with a typical substructure for subsequent mass/vibration analyses is based

on eq. (5) with $L_{11} = 0$. Unforced, dynamic equilibrium of the FFEE and RETAINED kinematic freedoms yields the following expression for the reduced mass matrix:

$$MRED = \begin{bmatrix} G \\ I \end{bmatrix}^T * \begin{bmatrix} M11 & M12 \\ M21 & M22 \end{bmatrix} * \begin{bmatrix} G \\ I \end{bmatrix} \quad (17)$$

where $G = - (L^T)^{-1} * A$ (18)

and the following notation has been adopted:

- I = An identity matrix
- M_{ij} = The i-j mass matrix partition (ref. SS-MERGE (MASS)).

Expanding eq. (17) yields

$$MRED = M22 + G^T * M12 + M12^T * G + G^T * M11 * G \quad (19)$$

(ref. SS-REDU (MASS)) for the reduced mass matrix.

Subsequent to calculating the foregoing reduced matrices for all the substructures that interact to form a higher-level substructure, the matrices are merged by the direct stiffness approach (ref. SS-MERGE). These solution steps (eq. 1-19) are repeated for each substructure included in a multilevel substructure analysis. When the highest-level substructure is reached, final matrix reductions are performed.

If a vibration analysis is to be performed on the highest-level substructure, the reduced mass and stiffness matrices are associated with boundary-condition stage number one and a stiffness/mass data set number defined via the interact data (sec. 130.0). These matrices are subsequently input to the VIBRATION Processor (sec. 258.1).

If a stress analysis is being performed, the interaction equilibrium equations for the highest-level substructure are solved using a reduction, solution and back-substitution approach that accounts for any retained freedoms (ref. SS-SSOL). The unknown displacements D₂₁ for the highest-level substructure are calculated as

$$D_{21} = KRED^{-1} * LRED \quad (20)$$

The unknown free displacements are calculated by

$$D11 = (L^T)^{-1} * (C - A * D21) \quad (21)$$

and the unknown reactions R31 are calculated as follows (ref. eq. 4 and 16)

$$R31 = E + K31 * D11 + K32 * D21 \quad (22)$$

Displacement matrices D21 (for the interacted freedoms) for all substructure components forming the highest-level substructure are generated by appropriate partitioning of matrix D11 and D21 (ref. SS-PART).

The load-case dependent matrices C, D31, LRED and E are expanded column-wise such that their column order and dimension are compatible with the associated D21 matrices. This is required since different load cases may be associated with the different substructure components. These expanded matrices are denoted by C^E , $D31^E$, $LRED^E$ and E^E .

The unknown FREE displacements associated with a lower-level substructure can now be calculated by defining

$$F = C^E - A * D21 \quad (23)$$

and performing a backward-substitution pass

$$D11 = (L^T)^{-1} * F \quad (24)$$

(ref. SS-BACK).

By eq. (7) the unknown reactions R21 can be calculated as

$$R21 = KRED * D21 - LRED^E \quad (25)$$

Additionally, by eq. (4) and eq. (16) the unknown reactions R31 are calculated as

$$R31 = E^E + K31 * D11 + K32 * D21 \quad (26)$$

These solution steps are performed for each level of interaction until the lowest-level substructure displacements and reactions are calculated. Subsequent to this, the STRESS Processor (sec. 254.0) may be executed to calculate element stresses.

APPENDIX E**ATLAS STANDARD CATALOGED CONTROL PROCEDURES**

The purpose of the ATLAS, standard, cataloged control procedures is to assist the user in creation of his Control Program for typical structural analyses. Standard control procedures are available for direct reference for analysis of stiffness and mass SET/STAGE structural components and for substructure components. The SET/STAGE procedures, as described in section E.1, assist the user in assemblage and reduction of structural matrices and in performing stress and design analyses. The substructure procedures, as described in section E.2, assist the user in assemblage and reduction of structural matrices and in performing interaction solutions associated with substructure analyses. Use of both types of procedures is illustrated by the sample problems included in appendix F.

As described in section 200.0, a cataloged procedure is included in a Control Program via the ATLAS statement "PERFORM Procedure." The function of each standard procedure and the equivalent sequence of ATLAS statements identified by each procedure are described herein. It should be noted that the standard procedures do not include any preprocessing or postprocessing statements. Thus, preprocessing of input data (FEAD INPUT) and postprocessing activities (PRINT, GRAPHICS, SAVE) must be initiated by user-supplied Control Program statements.

E.1--ATLAS SET/STAGE-STRUCTURE STANDARD CONTROL PROCEDURES

The function of each standard control procedure provided for analyses of SET/STAGE structural models is described in the following table. Each of these procedures is retrieved automatically via the ATLAS statement

PERFORM {Procedure} <(Plist)>

as described in section 200.4.1.

Procedure Name	Function	Plist Option ¹
DESIGN	Perform a fully-stressed design (free and support freedoms)	1, 2, 3, 4, 5, 7
F-REDUCE	Generate reduced stiffness and flexibility matrices	1, 2, 3, 6
K-REDUCE	Generate a reduced stiffness matrix	1, 2, 3, 6
M-REDUCE	Generate a reduced mass matrix	1, 2, 3, 6, 8
REDUCE	Generate reduced stiffness and mass matrices	1, 2, 3, 6, 8
R-STRESS	Perform a stress analysis (free, retained and support freedoms)	1, 2, 3, 4, 5, 7
STRESS	Perform a stress analysis (free and support freedoms)	1, 2, 3, 4, 5, 7

¹ Each integer in this column denotes which one of the following basic ATLAS-System parameters, with the default value shown, is associated with the "Procedure."

- | | | |
|-------------|----------------------|--------------|
| 1) SET=1 | 4) LC=ALL | 7) OPT=(1,2) |
| 2) STAGE=1 | 5) MATERIAL=CONSTANT | 8) OPTION=4 |
| 3) LUMP=0.0 | 6) RUN=STIFFNESS | |

A listing of the SET/STAGE standard-control procedures is included in the following pages. The User-Matrix names assigned by the execution-statement parameters within these procedures may be replaced by user-selected names via the PERFORM statement parameter lists (see sec. 200.4.1).

PROCEDURE DESIGN

C PURPOSE: PERFORM A FULLY STRESSED RESIZE OF A
C STRUCTURAL MODEL BY MODIFYING STIFFNESS ELEMENT *
C PROPERTIES (ACCORDING TO STRENGTH AND BUCKLING *
C CRITERIA) UNTIL EITHER THE CONVERGENCE CRITERIA *
C ARE SATISFIED OR THE FINAL CYCLE IS EXECUTED. *

C METHOD: 1. COMPUTE INITIAL WEIGHT *
C 2. EXECUTE STIFFNESS, STRESS, DESIGN AND MASS *
C PROCESSORS FOR DESIGN CYCLES BEGCYCL THROUGH *
C ENDCYCL. THE INITIAL CYCLE (BEGCYCL) AND *
C THE FINAL CYCLE (ENDCYCL) MUST BE DEFINED VIA *
C FORTRAN STATEMENTS IN THE CONTROL PROGRAM. *
C IF CONVERGENCE TESTING IS DESIRED, VALUES *
C MUST BE SPECIFIED FOR THE CONVERG(I) *
C VARIABLES IN THE CONTROL PROGRAM. *

C CONVERGENCE OPTIONS ARE AS FOLLOWS: *
C A) MAXIMUM ALLOWABLE RATIO BETWEEN NEW AND *
C OLD PROPERTIES FOR SPECIFIED ELEMENT *
C TYPES - - - CONVERG(1) THRU CONVERG(11) *
C FOR RODS THRU SPLATES *
C B) MAXIMUM ABSOLUTE CHANGE IN TOTAL WEIGHT *
C FOR 2 CONSECUTIVE CYCLES - - - CONVERG *
C (14) *
C C) MAXIMUM ABSOLUTE RATIO OF WEIGHT CHANGE *
C TO THE NEW TOTAL WEIGHT FOR 2 CON- *
C SECUTIVE CYCLES - - - CONVERG(15) *

C A PRINT OF THE CONVERGENCE SUMMARY DATA MAY *
C BE OBTAINED BY SPECIFYING A NONZERO VALUE FOR *
C CONVERG(16). CONVERGENCE FOR A SET OR STAGE *
C NUMBERED OTHER THAN ONE WILL REQUIRE ISET *
C AND/OR ISTAGE TO BE INITIALIZED. IF THE USER *
C DESIRES TO PRESERVE SNARK POSITIONS IN THE *
C CONTROL PROGRAM, THE LARGEST POSITION NUMBER *
C MUST BE STORED IN IPOS. *

C 3. GENERATE LOADS IN THE FIRST CYCLE ON.Y. *
C 4. IF BEGCYCL IS NOT EQUAL TO 1, A RESTART IS *
C ASSUMED AND FILE SAVESS4 IS LOADED. *
C 5. RESULTS ARE SAVED ON FILE SAVESS4 AND SHOULD *
C BE PRESERVED FOR RESTART. *

C RESULT: 1. MODIFIED STRUCTURAL PROPERTIES ON DATARNF *
C 2. STRUCTURAL WEIGHT PRINTED FOR EACH CYCLE *
C 3. D11,D31 - NODAL DISPLACEMENT PARTITIONS *
C 4. R31 ----- MATRIX OF REACTIONS FOR SUPPORT *
C FREEDOMS *
C 5. TOTAL STRUCTURAL NODAL DISPLACEMENTS (ON *
C STERNF) *
C 6. ELEMENT STRESSES *

```

C           7. ALL DATA GENERATED BY DESIGN PROCESSOR           *
C           (DESIRNF)                                           *
C                                                                 *
C SCRATCH MATRICES:                                           *
C           FILES CHOLRNF, MERGRNF, MULTRNF, STIFRNF, MASSRNF   *
C           AND STRERNF ARE SUBJECT TO PURGE DURING           *
C           INTERMEDIATE CYCLES.                               *
C                                                                 *
C VARIABLE NAMES USED:                                         *
C           BEGCYCL, CURCYCL, ENDCYCL, CONVERG, ISET, ISTAGE, IPOS *
C                                                                 *
C USER NOTE:                                                  *
C           THE CALL (CATM, KONTROL) CARD MUST BE USED AND THE *
C           FIRST STATEMENT IN THE CONTROL PROGRAM MUST BE:   *
C           BEGIN CONTROL MATRIX PROGRAM                       *
C           ADDITIONALLY, THE FOLLOWING STATEMENTS MUST BE    *
C           INCLUDED AT THE BEGINNING OF A CONTROL PROGRAM    *
C           THAT INCLUDES THIS PROCEDURE:                     *
C           USER COMMON (BEGCYCL, ENDCYCL, CURCYCL, CONVERG, ISET, ISTAGE, *
C           X           IPOS)                                  *
C           INTEGER BEGCYCL, ENDCYCL, CURCYCL                 *
C           DIMENSION CONVERG (17)                             *
C *****
C           IF (BEGCYCL .LT. 1) BEGCYCL=1
C           IF (ENDCYCL .LT. BEGCYCL) ENDCYCL=BEGCYCL
C           CURCYCL = BEGCYCL
C           IF (BEGCYCL .EQ. 1) GO TO 70001
C
C THIS IS A RESTART,
C LOAD RANDOM FILES FROM SAVESS4
C   LOAD FILES (SAVESS4=REWIND)
C   SAVE MATRIX (SAVESS4=REWIND, MERGRNF, L11, L31, D31)
C   GO TO 70002
70001 CONTINUE
C
C BEGCYCL = 1,
C COMPUTE AND PRINT INITIAL WEIGHT
C   EXECUTE MASS (SET=1)
70002 CONTINUE
C   CALL DESCONS
C   IF (CURCYCL .EQ. 0) CALL EXIT
C
C DESIGN CYCLES BEGCYCL THROUGH ENDCYCL
70003 CONTINUE
C   EXECUTE STIFFNESS (SET=1, LUMP=0.0)
C   EXECUTE MERGE (STIFFNESS, SET=1, STAGE=1, K11=11, K13=13, K33=33)
C   IF (CURCYCL .GT. 1) GO TO 70004

```

```

C
C BEGCYCL = 1,
C GENERATE LOADS
  EXECUTE LOADS (SET=1,STAGE=1,LC=ALL,MATERIAL=CONSTANT)
  EXECUTE MERGE (LOADS,SET=1,STAGE=1,L11=11,L31=31)
  EXECUTE MERGE (DISPLACEME,SET=1,STAGE=1,D31=31)
  SAVE MATRIX (SAVSS4=REWIND,MERGRNF,L11,L31,D31)
70004 CONTINUE
  EXECUTE MULTIPLY (TEMP=[ L11-K13*D31 ])
  EXECUTE CHOLESKY (SOLVE,K11,D11,TEMP)
  IF (CURCYCL .EQ. ENDCYCL) GO TO 70005
C
C INTERMEDIATE CYCLE, ASSEMBLE DISPLACEMENTS AND
C COMPUTE STRESSES IN INTERNAL ORDER
  EXECUTE STRESS (SET=1,STAGE=1,INTERNAL,D1=D11,D3=D31,OPT=(1,2))
  GO TO 70006
70005 CONTINUE
C
C LAST CYCLE, ASSEMBLE DISPLACEMENTS AND
C COMPUTE STRESSES IN USER ORDER
  EXECUTE STRESS (SET=1,STAGE=1,D1=D11,D3=D31,OPT=(1,2))
70006 CONTINUE
  EXECUTE DESIGN (SET=1,STAGE=1,CYCLE=CURCYCL,PROCEDURE=1)
  EXECUTE MASS (SET=1)
  CALL DESCONS
  IF (CURCYCL .GE. ENDCYCL) GO TO 70007
  PURGE FILES (MERGRNF,CHOLRNF,MULTRNF,STIFRNF,STRERNF,MASSRNF)
  LOAD FILES (SAVSS4=REWIND)
  CURCYCL = CURCYCL + 1
  GO TO 70003
70007 CONTINUE
C
C LAST CYCLE,
C COMPUTE REACTIONS AND PREPARE SAVSS4
  EXECUTE MULTIPLY (R31=[ -L31+K13 (T) *D11+K33*D31 ])
  SAVE MATRIX (SAVSS4,MULTRNF,R31)
  SAVE MATRIX (SAVSS4,MASSRNF,TOTLWT*)
  SAVE FILES (SAVSS4,DATAFNF,DESIRNF,LOADRNF,STRERNF)

```

PROCEDURE F-REDUCE

```
C*****
C PURPOSE:    GENERATE A REDUCED STIFFNESS MATRIX AND A REDUCED*
C             FLEXIBILITY MATRIX                                *
C
C METHOD:      1. GENERATE ELEMENT STIFFNESS MATRICES          *
C             2. ASSEMBLE THE STRUCTURAL STIFFNESS MATRIX     *
C             3. CALCULATE THE REDUCED STIFFNESS MATRIX       *
C             4. CALCULATE THE REDUCED FLEXIBILITY MATRIX BY  *
C                INVERTING THE REDUCED STIFFNESS MATRIX     *
C
C RESULT:     1. KRED ----- THE REDUCED STIFFNESS MATRIX   *
C             2. FRED ----- THE REDUCED FLEXIBILITY MATRIX *
C
C SCRATCH MATRICES:
C             FK12, DFRED, DKRED, K11, K12, K22                *
C*****
EXECUTE STIFFNESS (RUN=STIFFNESS, SET=1, LUMP=0.0)
EXECUTE MERGE (STIFFNESS, SET=1, STAGE=1, K11=11, K12=12, K22=22)
EXECUTE CHOLESKY (DEFO, K11, FK12, K12)
EXECUTE MULTIPLY (KRED=[ K22-FK12 (T) *FK12 ])
EXECUTE CHOLESKY (DECO, KRED=DKRED)
EXECUTE CHOLESKY (IFOR, DKRED, DFRED)
EXECUTE MULTIPLY (FRED=[ DFRED (T) *DFRED ])
```

PROCEDURE K-REDUCE

```
C*****
C PURPOSE:    GENERATE A REDUCED STIFFNESS MATRIX            *
C
C METHOD:      1. GENERATE ELEMENT STIFFNESS MATRICES          *
C             2. ASSEMBLE THE STRUCTURAL STIFFNESS MATRIX     *
C             3. CALCULATE THE REDUCED STIFFNESS MATRIX       *
C
C RESULT:     1. KRED ----- THE REDUCED STIFFNESS MATRIX   *
C
C SCRATCH MATRICES:
C             FK12, K11, K12, K22                              *
C*****
EXECUTE STIFFNESS (RUN=STIFFNESS, SET=1, LUMP=0.0)
EXECUTE MERGE (STIFFNESS, SET=1, STAGE=1, K11=11, K12=12, K22=22)
EXECUTE CHOLESKY (DEFO, K11, FK12, K12)
EXECUTE MULTIPLY (KRED=[ K22-FK12 (T) *FK12 ])
```

**CONTROL
PROCEDURES**

PROCEDURE M-REDUCE

```

C*****
C  PURPOSE:      GENERATE A GUYAN-REDUCED MASS MATRIX      *
C                                                         *
C  METHOD:       1. GENERATE ELEMENT STIFFNESS AND MASS MATRICES *
C               2. ASSEMBLE REQUIRED PARTITIONS OF STRUCTURAL *
C               STIFFNESS AND MASS MATRICES                 *
C               3. CALCULATE THE REDUCED MASS MATRIX        *
C                                                         *
C  RESULT:      1. MRED ----- THE GUYAN-REDUCED MASS MATRIX *
C                                                         *
C  SCRATCH MATRICES:                                     *
C               DBS,K11,K12,M11,M12,M22,TEMP                *
C*****
EXECUTE STIFFNESS (RUN=STIFFNESS,SET=1,LUMP=0.0)
EXECUTE MASS (SET=1,OPTION=4)
EXECUTE MERGE (STIFFNESS,SET=1,STAGE=1,K11=11,K12=12)
EXECUTE MERGE (MASS,SET=1,STAGE=1,M11=11,M12=12,M22=22)
EXECUTE CHOLESKY (SOLVE,K11,DBS,K12)
EXECUTE MULTIPLY (TEMP=[ M11*DBS ])
EXECUTE MULTIPLY (MRED [ M22-DBS (T) *M12-M12 (T) *DBS+DBS (T) *TEMP ])

```

PROCEDURE REDUCE

```

C*****
C  PURPOSE:      GENERATE A REDUCED STIFFNESS MATRIX AND A GUYAN- *
C               REDUCED MASS MATRIX                          *
C                                                         *
C  METHOD:       1. GENERATE ELEMENT STIFFNESS AND MASS MATRICES *
C               2. ASSEMBLE STRUCTURAL STIFFNESS AND MASS *
C               MATRICES                                     *
C               3. CALCULATE REDUCED STIFFNESS AND MASS MATRICES *
C                                                         *
C  RESULT:      1. KRED ----- THE REDUCED STIFFNESS MATRIX *
C               2. MRED ----- THE GUYAN-REDUCED MASS MATRIX *
C                                                         *
C  SCRATCH MATRICES:                                     *
C               DBS,DK11,FK12,K11,K12,K22,M11,M12,M22,TEMP *
C*****
EXECUTE STIFFNESS (RUN=STIFFNESS,SET=1,LUMP=0.0)
EXECUTE MASS (SET=1,OPTION=4)
EXECUTE MERGE (STIFFNESS,SET=1,STAGE=1,K11=11,K12=12,K22=22)
EXECUTE MERGE (MASS,SET=1,STAGE=1,M11=11,M12=12,M22=22)
EXECUTE CHOLESKY (DEFO,K11=DK11,FK12,K12)
EXECUTE CHOLESKY (BACK,DK11,DBS,FK12)
EXECUTE MULTIPLY (KRED=[ K22-FK12 (T) *FK12 ])
EXECUTE MULTIPLY (TEMP=[ M11*DBS ])
EXECUTE MULTIPLY (MRED=[ M22-DBS (T) *M12-M12 (T) *DBS+DBS (T) *TEMP ])

```

PROCEDURE R-STRESS

```

C*****
C  PURPOSE:    GENERATE NODAL DISPLACEMENTS, ELEMENT STRESSES  *
C              AND REACTIONS FOR A MODEL WITH FREE, RETAINED  *
C              AND SUPPORTED FREEDOMS.                          *
C
C  METHOD:     1. GENERATE ELEMENT STIFFNESS AND STRESS MATRICES*
C              2. GENERATE LOADS INFORMATION                    *
C              3. ASSEMBLE THE STRUCTURAL STIFFNESS MATRIX     *
C              4. ASSEMBLE STRUCTURAL LOADS AND SPECIFIED     *
C                 SUPPORT DISPLACEMENTS                       *
C              5. CALCULATE DISPLACEMENTS, STRESSES AND REACTIONS*
C
C  RESULT:    1. D11, D21, D31 - NODAL DISPLACEMENT PARTITIONS *
C              2. R31 ----- MATRIX OF REACTIONS FOR SUPPORT *
C                 FREEDOMS                                     *
C              3. TOTAL STRUCTURAL NODAL DISPLACEMENTS (ON   *
C                 STRERNF)                                    *
C              4. ELEMENT STRESSES                             *
C
C  SCRATCH MATRICES:
C              FK12, KRED, K11, K12, K13, K22, K23, K33, L, L11, L21, L31, *
C              TEMP1, TEMP2, TEMP3, TEMP4                       *
C              NOTE THAT KRED IS THE REDUCED STIFFNESS MATRIX  *
C*****
EXECUTE STIFFNESS (SET=1, LUMP=0.0)
EXECUTE LOADS (SET=1, STAGE=1, LC=ALL, MATERIAL=CONSTANT)
EXECUTE MERGE (STIFFNESS, SET=1, STAGE=1, K11=11, K12=12, K13=13,
X          K22=22, K23=23, K33=33)
EXECUTE MERGE (LOADS, SET=1, STAGE=1, L11=11, L21=21, L31=31)
EXECUTE MERGE (DISPLACEMENTS, SET=1, STAGE=1, D31=31)
EXECUTE CHOLESKY (DEFO, K11=L, FK12, K12)
EXECUTE MULTIPLY (KRED=[ K22-FK12 (T) *FK12 ])
EXECUTE MULTIPLY (TEMP1=[ L11-K13*D31 ])
EXECUTE CHOLESKY (FORW, L, TEMP2, TEMP1)
EXECUTE MULTIPLY (TEMP3=[ L21-K23*D31-FK12 (T) *TEMP2 ])
EXECUTE CHOLESKY (SOLVE, KRED, D21, TEMP3)
EXECUTE MULTIPLY (TEMP4=[ TEMP2-FK12*D21 ])
EXECUTE CHOLESKY (BACK, L, D11, TEMP4)
EXECUTE STRESS (SET=1, STAGE=1, D1=D11, D2=D21, D3=D31, OPT=(1, 2))
EXECUTE MULTIPLY (R31=[ -L31+K13 (T) *D11+K23 (T) *D21+K33*D31 ])

```

PROCEDURE STRESS

```

C*****
C  PURPOSE:      GENERATE NODAL DISPLACEMENTS,ELEMENT STRESSES  *
C                AND REACTIONS FOR A MODEL WITH FREE AND        *
C                SUPPORTED FREEDOMS                             *
C                *                                              *
C  METHOD:        1. GENERATE ELEMENT STIFFNESS AND STRESS MATRICES*
C                2. GENERATE LOADS INFORMATION                  *
C                3. ASSEMBLE THE STRUCTURAL STIFFNESS MATRIX    *
C                4. ASSEMBLE STRUCTURAL LOADS AND SPECIFIED    *
C                SUPPORT DISPLACEMENTS                         *
C                5. CALCULATE DISPLACEMENTS,STRESSES AND REACTIONS*
C                *                                              *
C  RESULT:       1. D11,D31 ----- NODAL DISPLACEMENT PARTITIONS *
C                2. R31 ----- MATRIX OF REACTIONS FOR        *
C                SUPPORT FREEDOMS                               *
C                3. TOTAL STRUCTURAL NODAL DISPLACEMENTS (ON  *
C                STRENF)                                        *
C                4. ELEMENT STRESSES                           *
C                *                                              *
C  SCRATCH MATRICES:
C                K11,K13,K33,L11,L31,TEMP                       *
C*****
EXECUTE STIFFNESS (SET=1,LUMP=0.0)
EXECUTE LOADS (SET=1,STAGE=1,LC=ALL,MATERIAL=CONSTANT)
EXECUTE MERGE (STIFFNESS,SET=1,STAGE=1,K11=11,K13=13,K33=33)
EXECUTE MERGE (LOADS,SET=1,STAGE=1,L11=11,L31=31)
EXECUTE MERGE (DISPLACEMENT,SET=1,STAGE=1,D31=31)
EXECUTE MULTIPLY (TEMP=[ L11-K13*D31 ])
EXECUTE CHOLESKY (SOLVE,K11,D11,TEMP)
EXECUTE STRESS (SET=1,STAGE=1,D1=D11,D3=D31)
EXECUTE MULTIPLY (R31=[ -L31+K13(T)*D11+K33*D31 ])

```


E.2--ATLAS SUBSTRUCTURE STANDARD CONTROL PROCEDURES

The function of each standard control procedure provided for analyses of substructure models is described in the following table. Each of these procedures is retrieved automatically by the ATLAS statement

PERFORM {Procedure} (Type, SS=CATlist)
--

as described in section 200.4.2.

Procedure Name	Function
SS-MERG	Assemble specified "Type" of matrices
SS-REDU	Generate reduced matrices of specified "Type"
SS-SSOL	Solve the interaction equations for the highest-level substructure
SS-VSOL	Generate reduced stiffness and mass matrices for the highest-level substructure for subsequent vibration analyses
SS-PART	Partition the displacement matrices and expand load-case dependent matrices for back-substitution calculations
SS-BACK	Solve for displacements and reactions using back-substitution

The "Type" parameter must be input as one or more of the words STIFFNESS, MASS or LOADS to identify the type of matrices that are to be assembled or reduced by the SS-MERG and SS-REDU procedures. The parameter "SS=CATlist," however, must be specified for each of the procedures. The numbers of the substructures on which the selected analysis function is to be performed are identified by "CATlist" (see sec. 200.4.2)

The sequence of control statements that is equivalent to each of the standard-control PERFORM statements for substructure analyses is presented in the following pages. The User-Matrix names assigned by the execution-statement parameters within these procedures are required for substructure analyses. The characters xxx denote the substructure number, right-adjusted and zero-filled, within each of the 7-character matrix names denoted herein.

1) **PERFORM SS-MERGE (STIFFNESS, SS=CATlist)**

This statement is used to assemble the elastic, element stiffness matrices for lowest-level substructures and to assemble reduced stiffness matrices associated with higher-level substructures. The equivalent ATLAS statement

```
EXECUTE MERGE(STIFFNESS,SS=xxx,RETAIN=2,OPTION=2)
```

is generated for each substructure identified by CATlist.

2) **PERFORM SS-MERGE (MASS, SS=CATlist)**

This statement is used to assemble element mass matrices for lowest-level substructures and to assemble reduced mass matrices associated with higher-level substructures. The equivalent ATLAS statement

```
EXECUTE MERGE(MASS,SS=xxx,RETAIN=2,OPTION=2)
```

is generated for each substructure identified by CATlist.

3) **PERFORM SS-MERGE (LOADS, SS=CATlist)**

This statement is used to assemble nodal loads matrices for lowest-level substructures and to assemble reduced loads matrices for higher-level substructures. All non-zero displacements specified via boundary condition data are also assembled. This statement is replaced by the two ATLAS statements

```
EXECUTE MERGE(LOADS,SS=xxx,RETAIN=2,OPTION=2)  
EXECUTE MERGE(DISPLACEME,SS=xxx,RETAIN=2,OPTION=2)
```

for each substructure identified by CATlist.

4) **PERFORM SS-REDUCTION (STIFFNESS, SS=CATlist)**

This statement is used to generate reduced stiffness matrices. It is replaced by the following two equivalent ATLAS statements for each substructure identified by CATlist.

```
EXECUTE CHOLESKY(DEF0,K11Sxxx=LSUBxxx,ASUBxxx,K12Sxxx,  
                OPTION=2)  
EXECUTE MULTIPLY(KREDxxx=[K22Sxxx-ASUBxxx(T)*ASUBxxx],  
                OPTION=2)
```

5) **PERFORM SS-REDUCTION (MASS, SS=CATlist)**

This statement is used to generate a Guyan-reduced mass matrix. It is replaced by the following set of equivalent ATLAS statements for each substructure identified by CATlist.

```
EXECUTE CHOLESKY (DEFO, K11Sxxx=LSUBxxx, ASUBxxx, K12Sxxx,
                 OPTION=2)
EXECUTE CHOLESKY (BACK, LSUBxxx, GSUBxxx, ASUBxxx, OPTION=2)
EXECUTE MULTIPLY (HSUBxxx=[ M11Sxxx*GSUBxxx ], OPTION=2)
EXECUTE MULTIPLY (MREDxxx=[ M22Sxxx-GSUBxxx (T) *M12Sxxx
                             -M12Sxxx (T) *GSUBxxx
                             +GSUBxxx (T) *HSUBxxx ], OPTION=2)
```

6) **PERFORM SS-REDUCTION (LOADS, SS=CATlist)**

This statement is used to generate reduced loads matrices. It is replaced by the following equivalent set of ATLAS statements for each substructure identified by CATlist.

```
EXECUTE CHOLESKY (DEFO, K11Sxxx=LSUBxxx, ASUBxxx, K12Sxxx,
                 OPTION=2)
EXECUTE MULTIPLY (BSUBxxx=[ L11Sxxx-K13Sxxx*D31Sxxx ], OPTION=2)
EXECUTE CHOLESKY (FORWARD, LSUBxxx, CSUBxxx, BSUBxxx, OPTION=2)
EXECUTE MULTIPLY (LREDxxx=[ L21Sxxx-K23Sxxx*D31Sxxx-ASUBxxx (T)
                             *CSUBxxx ], OPTION=2)
EXECUTE MULTIPLY (ESUBxxx=[ K33Sxxx*D31Sxxx-L31Sxxx ], OPTION=2)
```

7) **PERFORM SS-SSOLVE (SS=CATlist)**

This statement is used to solve the interaction equations for the highest-level substructure. The solution data are used for subsequent calculation of displacements, reactions and stresses for the lower-level substructures. This statement is replaced by the following set of ATLAS statements for each substructure identified by CATlist.

```
PERFORM SS-REDU (STIFFNESS, SS=xxx)
PERFORM SS-REDU (LOADS, SS=xxx)
EXECUTE CHOLESKY (SOLVE, KREDxxx, DBS2xxx, LREDxxx, OPTION=2)
RENAME MATRIX (D31Sxxx=DBS3xxx, CSUBxxx=CEXPxxx, LREDxxx=LEXPxxx,
              ESUBxxx=EEXPxxx)
PERFORM SS-BACK (SS=xxx)
```

The PERFORM statements (SS-REDU and SS-BACK) are expanded into the equivalent ATLAS statements as shown by (4), (6) and (10).

8) **PERFORM SS-VSOLVE(SS=CATlist)**

This statement is used to generate the reduced stiffness and reduced mass matrices for the highest-level substructure. The reduced matrices may be used subsequently for vibration analyses. This statement is replaced by the following PERFORM statements for each substructure identified by CATlist.

```
PERFORM SS-REDU(STIFFNESS,SS=xxx)  
PERFORM SS-REDU(MASS,SS=xxx)
```

These two statements are expanded into the equivalent ATLAS statements shown by (4) and (5).

9) **PERFORM SS-PARTITION (SS=CATlist)**

This statement initiates generation of the interaction (boundary) displacement matrices for the substructure components of the substructures identified by CATlist. The loadcase dependent matrices associated with the substructure components are expanded column-wise, as necessary, for compatible back-substitution calculations. This statement is replaced by the following set of equivalent ATLAS statements for each substructure identified by CATlist.

```
EXECUTE MERGE(PARTITION,SS=xxx,OPTION=2)  
EXECUTE MERGE(EXPANSION,SS=xxx,DBS3=D31S,OPTION=2)  
EXECUTE MERGE(EXPANSION,SS=xxx,CEXP=CSUB,OPTION=2)  
EXECUTE MERGE(EXPANSION,SS=xxx,LEXP=LRED,OPTION=2)  
EXECUTE MERGE(EXPANSION,SS=xxx,EEXP=ESUB,OPTION=2)
```

Note that the matrices generated via these statements are associated with the substructure components of the substructures identified by the parameter "SS=CATlist." The generated matrices are identified by the component substructure numbers.

10)

PERFORM SS-BACKSUBSTITUTION (SS=CATlist)

This statement is used to back-substitute the results of an interaction solution to calculate displacements and reactions for substructures. This statement is replaced by the following equivalent set of ATLAS statements for each substructure identified by CATlist.

```
EXECUTE MULTIPLY (FSUBxxx=[ CEXPxxx-ASUBxxx*DBS2xxx ],OPTION=2)
EXECUTE CHOLESKY (BACK,LSUBxxx,DBS1xxx,FSUBxxx,OPTION=2)
EXECUTE MULTIPLY (RBS2xxx=[ KREDxxx*DBS2xxx-LEXPxxx ],OPTION=2)
EXECUTE MULTIPLY (RBS3xxx=[ EEXPxxx+K13Sxxx (T) *DBS1xxx
                          +K23Sxxx (T) *DBS2xxx ],OPTION=2)
```

The displacements calculated via these statements for the lowest-level substructures are subsequently used to perform a stress analysis of each lowest-level substructure (ref. sec. 254.1). The displacement matrices identified by DBS1xxx (as calculated via SS-BACK) and the matrices identified by DBS2xxx and DBS3xxx (as generated via SS-PART) are used by the STRESS Processor. Printout of reaction-forces contained in the matrices identified by RBS2xxx and RBS3xxx may be requested from the REACTION Postprocessor (sec. 248.0).

APPENDIX F**SAMPLE ATLAS DECK SETUPS**

Two job deck-setups are presented in this appendix to illustrate typical applications of the ATLAS System. The decks illustrate the following:

- a) Stress and flutter analyses of a delta wing using a single structural model.
- b) Stress and vibration analyses of a transmission tower using a substructure interaction solution.

Listings of the complete job decks are included herein. The standard ATLAS cataloged control procedures were used in writing both of the Control Programs. The analysis options illustrated by these examples, including the selected print and plot displays, are not to be considered exhaustive or restrictive with regard to the complete ATLAS System capabilities. Additionally, the listed data should be used only for deck setup illustrative purposes.

An extensive set of problems that demonstrate the various analysis and design capabilities of ATLAS are documented in reference F-1. Listings of the input data and results are presented, as well as descriptions of the analyses and comparisons of results to theoretical or experimental data where possible.

APPENDIX F.1 — STRESS AND FLUTTER ANALYSES OF A DELTA WING

```

NACA,T3000,CM130000,POO.
***** ACCOUNT CARD *****
GET(CATM/UN=UATLASU)
CALL(CATM,ATTACH)
CALL(CATM,CONTROL)
ATLAS(PL=100000)
PLOTFIL(GERBER,TAPE99,0)
***** END OF RECORD *****
BEGIN CONTROL PROGRAM DEMO10
PROBLEM ID(DEMO10 - STRESS/FLUTTER ANALYSES OF A DELTA WING)
C
C PURPOSE THE PRINCIPAL ATLAS CAPABILITIES DEMONSTRATED BY
C THIS DECK ARE
C 1. NORMAL MODE ANALYSIS
C 2. PLOT OF VIBRATION MODE SHAPE
C 3. FLUTTER ANALYSIS - DUBLAT AERODYNAMICS
C 4. V-G AND V-F PLOTS
C 5. STRESS ANALYSIS
C 6. STRESS CONTOUR PLOTS
C
C METHOD THE SYSTEM IS EXECUTED TO DEMONSTRATE THE
C PROCEDURE TO OBTAIN STRESSES,DISPLACEMENTS, AND A
C FLUTTER SOLUTION FOR A DELTA WING.
C
C *****
C * THIS DECK DEMONSTRATES AN ATLAS ANALYSIS OF THE 45 DEGREE *
C * DELTA WING DESCRIBED IN NACA TN 3999. *
C * *
C * THE PROBLEM TO BE EXECUTED CONTAINS *
C * *
C * 117 STRUCTURAL NODES *
C * 154 SPAR ELEMENTS *
C * 102 COVER ELEMENTS *
C * *
C *****
C
C READ INPUT (MODE2)
C
C PRINT THE STIFFNESS AND LOADS INPUT DATA. PLOT GEOMETRY.
C
C PRINT INPUT (NODAL,SUBSETS,N1=INPUT,N2)
C PRINT INPUT (STIFFNESS,SUBSETS,E10,E20,E30)
C PRINT INPUT(LOADS)
C EXECUTE EXTRACT(EXNAME=NACA1,LSUB=KGRID,ESUB=E10,NSUB=N1)
C EXECUTE GRAPHICS(GNAME=PLANVIEW,OFFLINE=GERBER,TYPE=(ORTH,
X POINT),LABEL=N,SCALE=.05,VIEW=100,EXNAME=NACA1)
C EXECUTE EXTRACT(EXNAME=NACA2,LSUB=KGRID,ESUB=E25,NSUB=N30)
C EXECUTE GRAPHICS(GNAME=REARSPAR,TYPE=ORTH,LABEL=N+E,SCALE=.05,
X VIEW=100000,EXNAME=NACA2)
C
C GENERATE THE REDUCED STIFFNESS MATRIX.
C
C PERFORM K-REDUCE
C PRINT INPUT(BC)
C

```

**SAMPLE DECK
SETUPS**

```

C      GENERATE AND PRINT THE DIAGONAL MASS MATRIX,PANEL WEIGHT
C      MATRIX,AND WEIGHT STATEMENT.
C
EXECUTE MASS (OPTION=2)
PRINT OUTPUT(MASS,STATEMENT,SUMMARY,MDC=MDC****)

C      GENERATE AND PRINT THE MODES,FREQUENCIES,GENERALIZED MASS,AND
C      GENERALIZED STIFFNESS.  PRODUCE MCDE SHAPE PLOT.
C
EXECUTE VIBRATION(STIF=KRED,MASS=MDCOOIA,NMDES=7,SUBSETS=(N1,
X      N29))
X PRINT OUTPUT(VIBRATION,SUBSETS=(N1,N29))
EXECUTE EXTRACT(EXNAME=SPAR2,LSUB=VMODE,VSET=1,NSUB=N29,MODE=3,
X      BSUB=GN2)
X EXECUTE GRAPHICS(GNAME=MODES,TYPE=ORTH,SCALE=.1,VX=-1.,
X      VECTOR2=VMODE,VSCALE=60.,EXNAME=SPAR2)

C      GENERATE,PRINT,AND PLOT THE FLUTTER DATA.
C
EXECUTE INTERPOLATION(N1=(SURFSPLINE,CMCD),OCF=1000)
EXECUTE DUBLAT(COND=1,MACH=.5,KVAL=(35.0,1.,2,5.0,2.2,1.0),
1      BREF=112.0)
EXECUTE ADDINT(ID=FCHCK,(INT,DUBLAT,IGAIN=9,MACH=.5)
EXECUTE FLUTTER(GAFID=FCHCK)
PRINT OUTPUT(FLUTTER)
EXECUTE EXTRACT(EXNAME=FCASE1,LSUB=VGVF)
EXECUTE GRAPHICS(GNAME=FLUTTER,TYPE=GRAPH,SIZE=(15,15),
X      X=V,Y1=G,Y2=F,XMIN=0.,XMAX=3000.,Y1MIN=-.2,
X      Y2MIN=0.,Y1MAX=.1,Y2MAX=200.,EXNAME=FCASE1)

C      GENERATE AND PRINT THE ELEMENT STRESSES,NODAL DISPLACEMENTS,
C      AND REACTIONS.  PLOT STRESS CONTOURS.
C
PERFORM STRESS(STAGE=2,[K]=[S])
PRINT OUTPUT(STRESSES,STAGE=2)
PRINT OUTPUT(DISPLACE,STAGE=2)
PRINT OUTPUT(REACTIONS,STAGE=2,R3=R31,EGCHK)
EXECUTE EXTRACT(EXNAME=NACA3,LSUB=STRESS,ESLB=E10,NSUB=N25,
X      STAGE=2,BSUB=ON1)
X EXECUTE GRAPHICS(GNAME=CONTOURS,TYPE=CCNTCUR,SIZE=(15,15),
X      SCALAR=COVSIGMA1U,FLEV=-250.,LLEV=0.,
X      INTLEV=25.,EXNAME=NACA3)
EXECUTE GRAPHICS(GNAME=CONTOURS,TYPE=CONTOUR,SIZE=(15,15),
X      SCALAR=COVSIGMA2U,FLEV=-60.,LLEV=30.,
X      INTLEV=10.,EXNAME=NACA3)

C      INDEX THE RANDOM ACCESS FILES.
C
CALL PRNTCAT
INDEX FILES
END
***** END OF RECORD *****

BEGIN NODAL DATA
*/ STRUCTURAL NODES
15 196.0 0.0 0.0 2.72 TO 195 100.0 0.0 0.0 2.72 BY 15
**2 -1 0.0 8.0 0.0 0.0 0 -1 0.0 8.0 0.0 0.0 0 0
1 196.0 112.0 0.0 0.844 TO 193 BY 16
1 TO 13
**11 16 0 15

```



```

*/ AUXILIARY PANEL NODES
310 196. 0. 0. TO 316 100. 0. 0.
320 * 16. *2 326 * 16. 0.
330 * 28. *2 336 112. 28. 0.
340 * 40. *2 346 124. 40. 0.
350 * 40. *2 354 124. 40. 0.
360 * 56. *2 364 140. 56. 0.
370 * 72. *2 374 156. 72. 0.
380 * 88. *2 382 172. 88. 0.
390 * 112. 0.
END NODAL DATA
BEGIN STIFFNESS DATA
BEGIN ELEMENT DATA
*/ RIB ELEMENTS
SPAR M01 15,30 0.0254 .01905 TO 180,195 BY 15,15
*2 14,29 0.0508 .03810 TO 179,194 **
*2 13,28 0.1412 .12355 TO 178,193 **
*2 12,27 0.0508 .03810 TO 162,177 **
**9 0,0 -1,-1 0,0 .0 0 -16,-16 0,0,0
*2 2,17 *2
*/ SPAR ELEMENTS
SPAR 1 2 0.0695 0.201362 TO 3 4
* 4 5 0.0695 0.474579 TO 14 15
* 49 50 0.0701 0.477268 TO 59 60
* 97 98 0.0718 1.498695 TC 104 105
* 145 146 0.0720 0.205189 TO 149 150
* 193 194 0.0706 0.039597 TO 194 195
* 1 17 0.0706 0.039597 TC 177 193 BY 16 16
*/ COVER ELEMENTS
COVER N1001 1 2 17 0.0696 0.0281 0. TO N1166 177 178 193 *
BY N15 16 16 16
COVER N1002 2 3 18 17 0.0696 0.0281 0. TO N1014 14 15 30 29
**11 0 15 16 **3 0. 0. 0. 0 14 15 **3
END ELEMENT DATA
END STIFFNESS DATA
*/
BEGIN BC DATA
STAGE 1
RETAIN TZ FOR 1 TO 195
SUPPORT TX FOR 195
SUPPORT ASYM IN SURFACE 2 THROUGH 15
STAGE 2
SUPPORT ALL FOR 15 TO 195 BY 15
END BC DATA
*/
BEGIN MASS DATA
BEGIN CONDITION DATA
STAGE 1 CONDITION 1
PANEL DATA 1 CONDITION 2
END CONDITION DATA
BEGIN PANEL DATA 1
1 315 316 326 325 TO 6 BY 1 -1 **3
**2 0 10 10 10 10 0 0 0 **5
19 353 354 364 363 TO 22 **
**1 4 10 10 10 10 0 4 0 **5
27 372 374 382 381 TO 29 BY 1 -2 -2 -1 -1
29 380 382 390
END PANEL DATA
BEGIN LABEL DATA
LEVEL1 *# TOTAL WING STRUCTURE #
EK10 *# COVER MATERIAL #
EK30 *# RIB MATERIAL #
LEVEL2 *# SPAR MATERIAL #
EK21 *# SPAR 1 (FS) #
EK22 *# SPAR 2 #
EK23 *# SPAR 3 #
EK24 *# SPAR 4 #
EK25 *# SPAR 5 (RS) #
END LABEL DATA
END MASS DATA

```

**SAMPLE DECK
SETUPS**

```

BEGIN LOADS DATA
STAGE 2
BEGIN ELEMENT LOAD DATA
CASE JUNK1
DIRECTION GLOBAL 0. 0. 1.
1014 TO 1168 BY 14 .2
1013 TO 1167 BY 14 .15
1012 TO 1166 BY 14 .10
**10 -1 0 -15 0 0 -.01
1001 .005
END ELEMENT LOAD DATA
END LOADS DATA
BEGIN SUBSET DEFINITION
SUBSETS OF STIFFNESS SET 1
*/ SUBSETS OF ALL STRUCTURAL NODES AND ELEMENTS
E1 = ALL
N1 = ALL
N2 = 300 TO 400
EXCLUDE N2 FROM N1
*/ SUBSET OF ALL COVER ELEMENTS
E10 = COVERS
*/ SUBSETS OF ELEMENTS FOR THE WING SPARS
N21 = 1 TO 193 BY 16 194 195
E21 = IN N21
E22 = SLAB X 124.0
**3 1 0 0 0 24.0
N25 = NODES IN E25
*/ SUBSET OF ALL WING SPARS
E20 = E21 U E22 U E23 U E24 U E25
*/ SUBSET OF ALL SPAR ELEMENTS
E100 = SPARS
*/ SUBSET OF ALL WING RIBS
E30 = E100
EXCLUDE E20
*/ RETAINED NODES ON SPAR 2
N29 = 49 TO 60
*/ BOUNDARY SUBSET FOR STRESS CONTOUR PLOTS
ON1 = 1 15 195 193
*/ SUBSET FOR MODE SHAPE PLOT
ON2 = 49 **11+1
END SUBSET DEFINITION
BEGIN FLUTTER DATA
CASE 1 *NACA DELTA WING *
ALTITUDE 500.0
END FLUTTER DATA
*/
BEGIN DUBLAT DATA
BEGIN GEOMETRY DATA
LIFTING SURFACE DATA
PANEL WING1 100.0 196.0 100.0 196.0 0. 16.0 0. 0.
CHORD DIV 0. 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
SPAN DIV 0. 1.0
PANEL WING2 100.0 196.0 190.0 196.0 16.0 112.0 0. 0.
CHORD DIV 0. 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
SPAN DIV 0. 0.1 0.2 0.3 0.4 0.5 0.57 0.64 0.7 0.75 0.8 +
0.85 0.88 0.92 0.95 0.98 1.0
END GEOMETRY DATA
BEGIN SUBSET DATA
SUBSETS OF BOXES
SUBSET SS 1 TO 170
END SUBSET DATA
BEGIN MODAL DATA
USE CMOD WITH LIFTING SURFACE SS
END MODAL DATA
END DUBLAT DATA
END PROBLEM DATA
***** END OF FILE *****

```

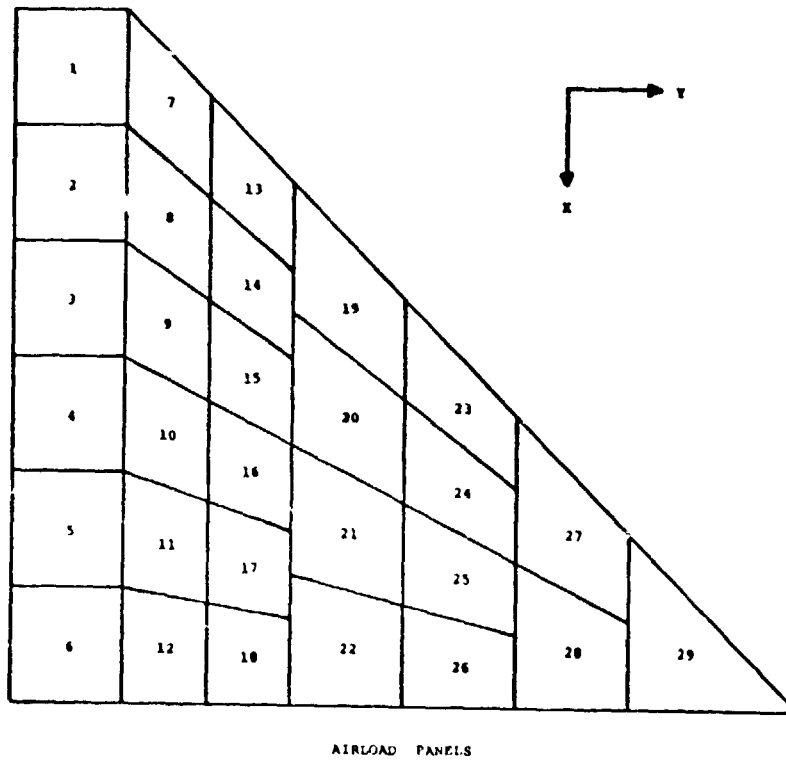
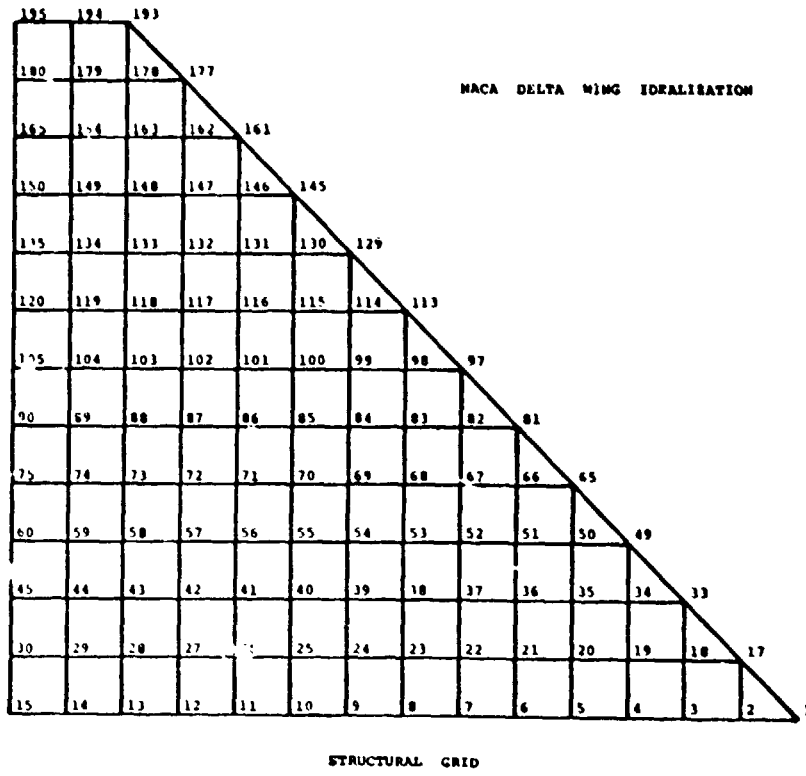


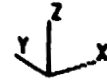
Figure F-1. Delta-Wing Structural and Airload Grids

**SAMPLE DECK
SETUPS**

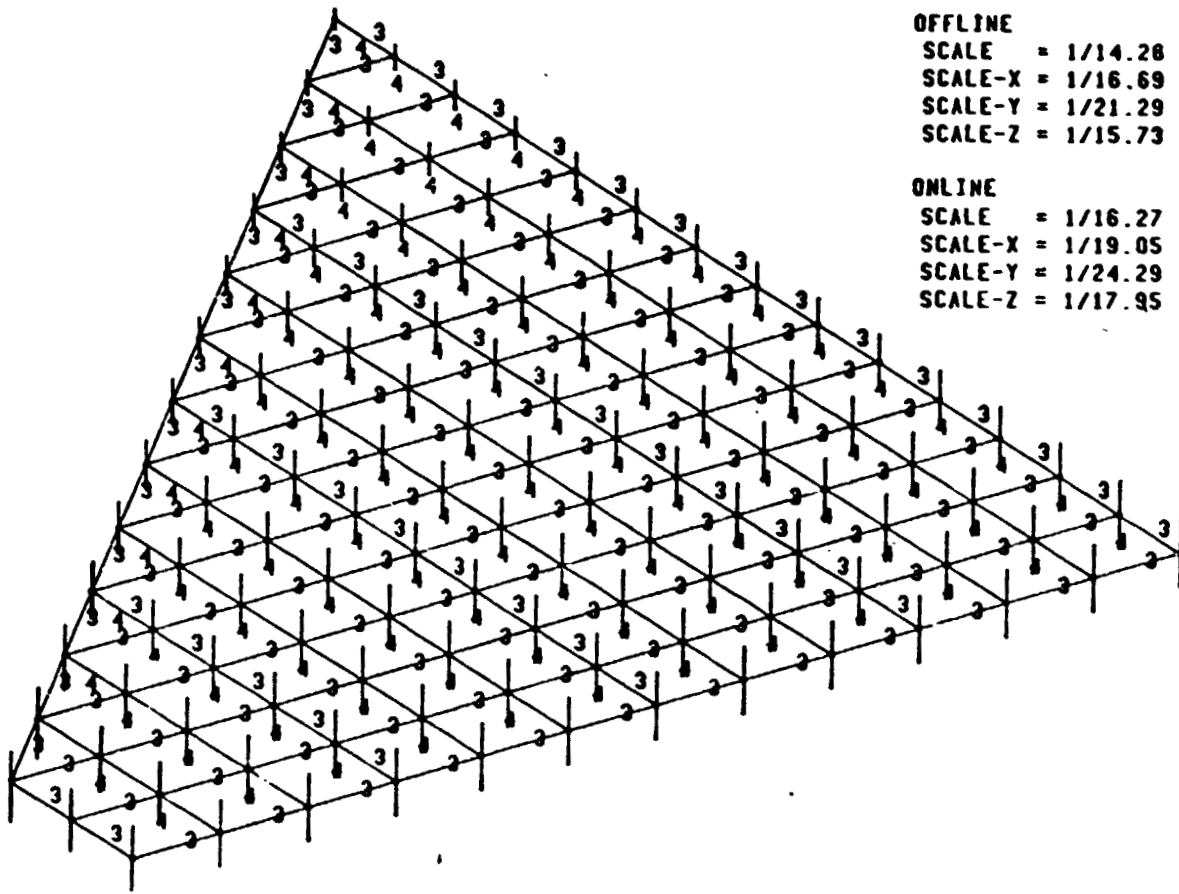
GNAME = GEOM01

PLOT ID = K1E1

GEOMETRY



RZ = 125.
RY = 25.
RX = 0.0



OFFLINE
SCALE = 1/14.28
SCALE-X = 1/16.69
SCALE-Y = 1/21.29
SCALE-Z = 1/15.73

ONLINE
SCALE = 1/16.27
SCALE-X = 1/19.05
SCALE-Y = 1/24.29
SCALE-Z = 1/17.95

EXNAME = K1E1 KSET = 1 NSUB = N111
ESUB = E001 LSUB = KGRID

Figure F-2. Plot of Delta-Wing Element Data

GNAME = VGVFPLOTS

PLOT 10 = V-G, CASE = 12, CO = 1, CS = 8, RS = 3

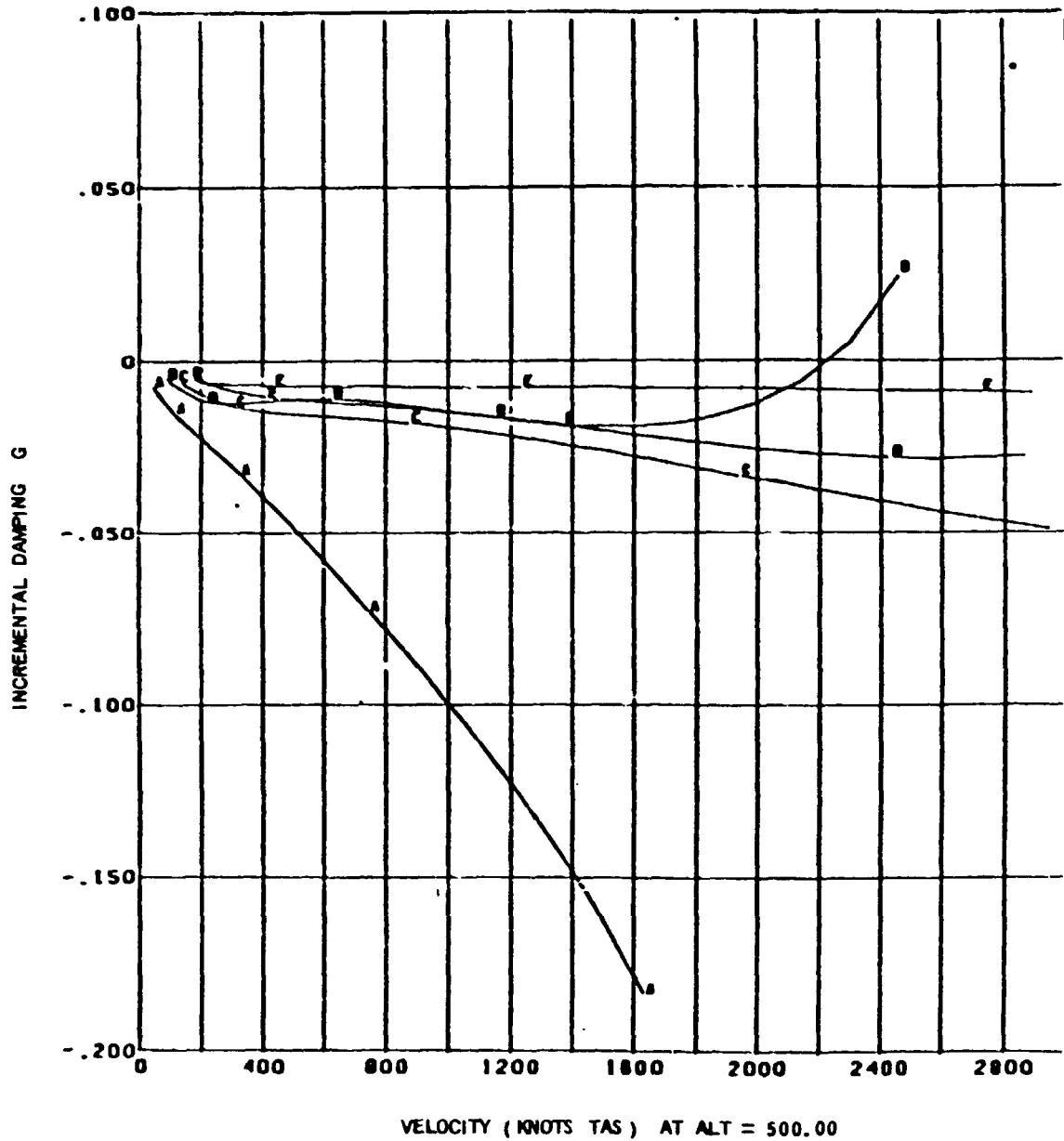


Figure F-3. Flutter V-g Plot for Delta-Wing

GNAME = VGVFPLOTS

PLOT ID = V-F, CASE = 12, CO = 1, CS = 5, RS = 3

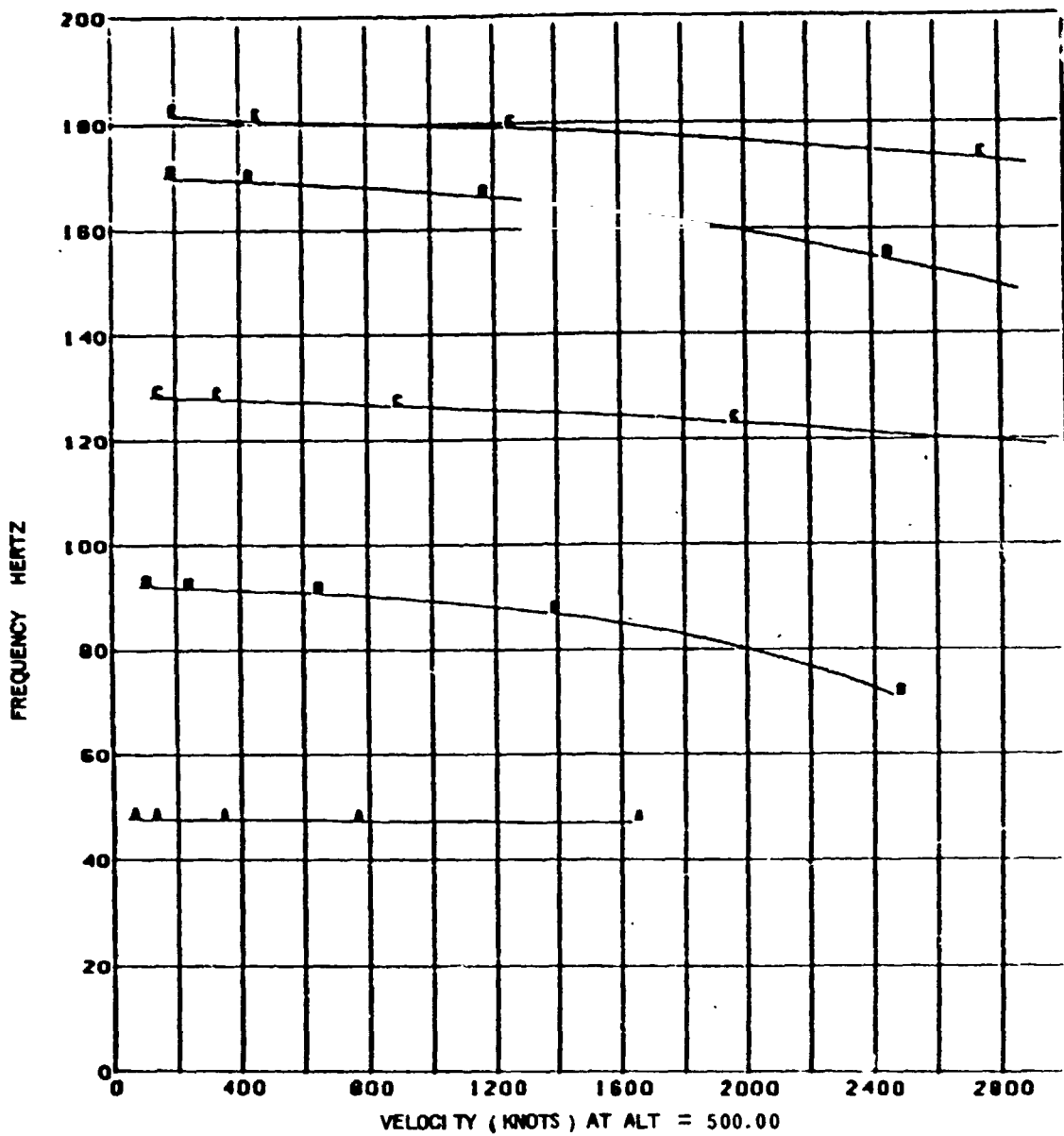


Figure F-4. Flutter V-f Plot for Delta-Wing

**SAMPLE DECK
SETUPS**

```

C      *      7)  PERFORM SS-VSCL ..... COMPUTE REDUCED STIFFNESS AND*
C      *      *      MASS MATRICES FOR SS=5 TO *
C      *      *      PERFORM SLBSEQUENT VIBRATION *
C      *      *      ANALYSIS. *
C      *      *      *
C      *      8)  EXECUTE MASS ..... COMPUTE MASS PROPERTIES FOR *
C      *      *      TOTAL STRUCTURE. (SET=5) *
C      *      *      *
C      *      9)  EXECUTE VIBRATION .. COMPUTE THE FIRST 5 FREQUEN- *
C      *      *      CIES AND MODES. *
C      *      *      *
C      *      10) PERFORM SS-SSOL ..... COMPUTE DISPLACEMENTS FOR *
C      *      *      SS=5. *
C      *      *      *
C      *      11) PERFORM ..... FROM THE DISPLACEMENT MATRI- *
C      *      *      SS-PARTITION CES FOR SS=5, EXTRACT DISPLA- *
C      *      *      CEMENTS AT RETAINED FREEDOMS *
C      *      *      FOR SS=1 TO 4. *
C      *      *      *
C      *      12) PERFORM SS-BACK .... PERFORM BACK SUBSTITUTION TO *
C      *      *      COMPUTE ALL DISPLACEMENTS AND *
C      *      *      REACTIONS FOR SS=1 TO 4. *
C      *      *      *
C      *      13) EXECUTE STRESS, ..... COMPUTE ELEMENTAL STRESSES *
C      *      *      PRINT STRESSES, AND PRINT STRESSES, *
C      *      *      DISPLACEMENTS DISPLACEMENTS AND REACTIONS *
C      *      *      AND REACTIONS FOR SUBSTRUCTURES 1 TO 4 *
C      *      *      *
C      *      14) PRINT INPUT, ..... PRINT NODAL AND STIFFNESS *
C      *      *      NODAL AND INPUT DATA FOR SETS 1 TO 4 *
C      *      *      STIFFNESS *
C      *      *      *
C      *      15) EXECUTE GRAPHICS ... PLOT TOTAL STRUCTURE AND *
C      *      *      COMPONENT SUBSTRUCTURES *
C      *      *      *
C      *      16) PRINT INPUT, ..... PRINT INTERACT DATA FOR *
C      *      *      INTERACT ALL SUBSTRUCTURES. *
C      *      *      *
C      *      17) PRINT OUTPUT, ..... PRINT FREQUENCIES AND MODE *
C      *      *      VIBRATION SHAPES. *
C      *      *      *
C      *      18) ERROR PROCEDURE .... SAVE DATA FILES IF AN ERROR *
C      *      *      IS ENCCOUNTERED DURING *
C      *      *      EXECUTION. *
C      *      *      *

```

```

C      *****
C      USER COMMON (K)
C      ---- 1) ----
C      READ INPUT
C      ---- 2) ----
C      DO 10 K=1,4
C      EXECUTE STIFFNESS (SET=K)
C      EXECUTE MASS (SET=K,OPTION=3,CONDITION=1)
C      10 CONTINUE

```



```

C ----- 3) -----
    EXECUTE LOADS (SS=(1,3))
C ----- 4) -----
    PERFORM SS-MERGE (STIF,LOAD,SS=1 TO 4)
C ----- 5) -----
    PERFORM SS-REDU (STIF,LOAD,SS=1 TO 4)
C ----- 6) -----
    PERFORM SS-MERGE (STIF,LOAD,MASS,SS=5)
C ----- 7) -----
    PERFORM SS-VSOL (SS=5)
C ----- 8) -----
    EXECUTE MASS (SET=5,CONDITION=1)
C ----- 9) -----
    EXECUTE VIBRATION (STIF=KREDO05,MASS=MREDO05,NFREQS=5,SET=5)
C ----- 10) -----
    PERFORM SS-SSOL (SS=5)
C ----- 11) -----
    PERFORM SS-PARTITION (SS=5)
C ----- 12) -----
    PERFORM SS-BACK (SS=1 TO 4)
C ----- 13) -----
    DO 20 K=1,4
        EXECUTE STRESS(SS=K)
        PRINT OUTPUT(STRESS,SS=K)
        PRINT OUTPUT(DISP,SS=K)
        PRINT OUTPUT(REACTIONS,EQCHK,SS=K)
    20 CONTINUE
C ----- 14) -----
    DO 30 K=1,4
        PRINT INPUT (NODAL,SET=K)
        PRINT INPUT (STIFFNESS,SET=K)
    30 CONTINUE
C ----- 15) -----
    EXECUTE EXTRACT(EXNAME=SS1,LSUB=KGRID,KSET=1,ESUB=E1,NSUB=N2)
    EXECUTE EXTRACT(EXNAME=SS2,LSUB=KGRID,KSET=2,ESUB=E1,NSUB=N1)
    EXECUTE EXTRACT(EXNAME=SS3,LSUB=KGRID,KSET=3,ESUB=E1,NSUB=N1)
    EXECUTE EXTRACT(EXNAME=SS4,LSUB=KGRID,KSET=4,ESUB=E1,NSUB=N1)
    EXECUTE GRAPHICS(GNAME=GEOM,OFFLINE=GERBER,EXPLODE,TYPE=ORTH,
X          SIZE=(15,15),EXNAME=(SS1,SS2,SS3,SS4))
    EXECUTE GRAPHICS(GNAME=GEOM,EXPLODE,TYPE=CRTH,SIZE=(15,15),
X          TX=-250.,EXNAME=SS1,TX=0.,TY=100.,EXNAME=SS2,
X          TX=50.,TY=0.,EXNAME=SS3,TX=0.,TY=-250.,
X          EXNAME=SS4)
C ----- 16) -----
    PRINT INPUT (INTERACT,NCDE,SS=1 TO 5)
    PRINT INPUT (INTERACT,CONN,SS=1 TO 4)
    PRINT INPUT (INTERACT,RETA,SS=1 TO 4)
    PRINT INPUT (INTERACT,BC,SS=1 TO 5)
    PRINT INPUT (INTERACT,LOADS,SS=1 TO 5)
C ----- 17) -----
    PRINT OUTPUT (VIBRATION)
    ERROR PROCEDURE
    SAVE FILES
    END CONTROL PROGRAM
          ***** END OF RECORD *****

```

**SAMPLE FBCK
SETUPS**

```

*/ ***** /
BEGIN NODAL DATA /
    1 169.0 148.0 634.5 /
    2 169.0 190.0 634.5 /
    400 112.8 169.0 96.0 /
    1001 100.0 * 408.0 TO 1003 100.0 169.0 600.0 /
    101 100.0 100.0 0.0 TO 106 148.0 * 360.0 /
    OF 8.0,7.0,6.0,5.0,4.0 /
    401 * 238.0 *2 406 148.0 190.0 ** /
    106 TO 116 148.0 148.0 600.0 /
    406 * 416 * 190.0 ** /
REORDER FROM 101 /
SET 2 /
*/ ***** /
    2 169.0 190.0 634.5 /
    300 * 225.2 96.0 /
    301 238.0 *= 0.0 TO 306 190.0 *= 360.0 /
    OF 8.0,7.0,6.0,5.0,4.0 /
    401 100.0 *3 406 148.0 ** /
    306 TO 316 190.0 190.0 600.0 /
    406 * 416 148.0 ** /
REORDER FROM 301 /
SET 3 /
    1 169.0 148.0 634.5 /
    2 * 190.0 * /
    200 225.2 169.0 96.0 /
    2001 238.0 * 408.0 TO 2003 238.0 169.0 600.0 /
    201 * 100.0 0.0 * 206 190.0 148.0 360.0 /
    OF 8.0,7.0,6.0,5.0,4.0 /
    301 * 238.0 *2 306 * 190.0 ** /
    206 TO 216 190.0 148.0 600.0 /
    306 * 316 * 190.0 ** /
REORDER FROM 201 /
SET 4 /
*/ ***** /
    1 169.0 148.0 634.5 /
    100 * 112.8 96.0 /
    101 100.0 *= 0.0 TO 106 148.0 *= 360.0 /
    OF 8.0,7.0,6.0,5.0,4.0 /
    201 238.0 *3 206 190.0 ** /
    106 TO 116 148.0 148.0 600.0 /
    206 * 216 190.0 ** /
REORDER FROM 101 /
END NODAL DATA /
BEGIN STIFFNESS DATA /
BEGIN PROPERTY DATA /
    P1 5.75 0. 0. 39.8 19.9 *= /
    P2 3.75 0. 0. 11.2 5.6 5.6 /
    P3 2.11 0. 0. 3.6 1.8 1.8 2.11 0. 0. 3.6 1.8 1.8 111.0 /
    P4 2.11 2.11 /
    P5 1.15 *= /
    P6 3.25 3.25 /
END PROPERTY DATA /
BEGIN ELEMENT DATA /
    BEAM M8 101 102 400 P1 TO 105 106 400 BY 1 1 0 /
    * 401 402 400 P1 TC 405 406 400 BY 1 1 0 /

```

```

*      106 107 406 P2 TO 115 116 406 ** /
*      406 407 106 P2 TO 415 416 106 ** /
*      416 2 116 P2 /
*      116 1 416 P2 /
*      402 400 403 P3 /
*      102 400 103 P3 /
ROD    401 400 P4 /
*      101 400 * /
*      403 400 * /
*      103 400 * /
*      403 103 P5 TO 416 116 /
*      103 404 *2 115 416 BY 2 2 /
*      404 105 *2 414 115 ** /
*      1 2 P5 /
*      1 416 * /
*      1001 108 P6 TO 1003 116 BY 1 4 /
*      * 408 * * 416 ** /
*      * 109 * /
*      * 409 * /
*      1002 113 * /
*      * 413 * /
*      1003 1 * /
*      * 2 * /
END ELEMENT DATA /
SET 2 /
*/ ***** /
BEGIN PROPERTY DATA /
P3 2.11 0. C. 3.6 1.8 1.8 2.11 0. 0. 3.6 1.8 1.8 111.0 /
P4 2.11 2.11 /
P5 1.15 *= /
END PROPERTY DATA /
BEGIN ELEMENT DATA /
BEAM M8 302 300 303 P3 /
BEAM 402 300 403 P3 /
ROD 301 300 P4 /
ROD 401 300 * /
ROD 303 * * /
ROD 403 * * /
ROD 303 403 P5 TO 316 416 /
ROD 403 304 *2 415 316 BY 2 2 /
ROD 304 405 *2 314 415 ** /
END ELEMENT DATA /
SET 3 /
*/ ***** /
BEGIN PROPERTY DATA /
P1 5.75 0. 0. 39.8 19.9 *= /
P2 3.75 0. 0. 11.2 5.6 5.6 /
P3 2.11 0. 0. 3.6 1.8 1.8 2.11 0. 0. 3.6 1.8 1.8 111.0 /
P4 2.11 2.11 /
P5 1.15 *= /
P6 3.25 3.25 /
END PROPERTY DATA /
BEGIN ELEMENT DATA /
BEAM M8 201 202 200 P1 TO 205 206 200 BY 1 1 0 /
* 301 302 200 P1 TO 305 306 200 BY 1 1 0 /
* 206 207 306 P2 TO 215 216 306 ** /

```

SAMPLE DECK
SETUPS

```

*      306 307 206 P2 TO 315 316 206 ** /
*      216 1 316 P2 /
*      316 2 216 * /
*      202 200 203 P3 /
*      302 200 303 P3 /
ROD    201 200 P4 /
*      301 * * /
*      203 * * /
*      303 * * /
*      203 303 P5 TO 216 316 /
*      303 204 * * 315 216 BY 2 2 /
*      204 305 * * 214 315 ** /
*      216 2 * /
*      2001 208 P6 TO 2003 216 BY 1 4 /
*      * 308 * * 316 ** /
*      * 209 P6 /
*      * 309 * /
*      2002 213 * /
*      * 313 * /
*      2003 1 * /
*      * 2 * /
END ELEMENT DATA /
SET 4 /
*/ ***** /
BEGIN PROPERTY DATA /
P3 2.11 0. 0. 3.6 1.8 1.8 2.11 0. 0. 3.6 1.8 1.8 111.0 /
P4 2.11 2.11 /
P5 1.15 *= /
END PROPERTY DATA /
BEGIN ELEMENT DATA /
BEAM M8 102 100 103 P3 /
* 202 100 203 P3 /
ROD 101 100 P4 /
* 201 100 * /
* 103 100 * /
* 203 100 * /
* 103 203 P5 TO 116 216 /
* 203 104 *2 215 116 BY 2 2 /
* 104 205 *2 114 215 ** /
END ELEMENT DATA /
END STIFFNESS DATA /
*/
BEGIN BC DATA /
SET 1 /
RETAIN TX TY TZ FOR 1001 TO 1003 /
RETAIN TX TY FOR 400 /
SUPPORT ALL FOR 101 401 /
SET 2 /
RETAIN TX TY FOR 300 /
SUPPORT ALL FOR 301 401 /
SET 3 /
RETAIN TX TY TZ FOR 2001 TO 2003 /
RETAIN TX TY FOR 200 /
SUPPORT ALL FOR 201 301 /
SET 4 /
RETAIN TX TY FOR 100 /

```

```

SUPPORT ALL FOR 101 201
END BC DATA
*/
BEGIN MASS DATA
  SET 1
    BEGIN CONDITION DATA
      STAGE 1 CONDITION 1
    END CONDITION DATA
  END MASS DATA
  BEGIN MASS DATA
    SET 2
      BEGIN CONDITION DATA
        STAGE 1 CONDITION 1
      END CONDITION DATA
    END MASS DATA
  BEGIN MASS DATA
    SET 3
      BEGIN CONDITION DATA
        STAGE 1 CONDITION 1
      END CONDITION DATA
    END MASS DATA
  BEGIN MASS DATA
    SET 4
      BEGIN CONDITION DATA
        STAGE 1 CONDITION 1
      END CONDITION DATA
    END MASS DATA
*/
BEGIN LOADS DATA
  SET 1
    BEGIN NODAL LOAD DATA
      CASE 1
        ORDER FX FY FZ
          1001 TO 1003 10000.0 *= 20000.0
      END NODAL LOAD DATA
    SET 3
      BEGIN NODAL LOAD DATA
        CASE 1
          ORDER FX FY FZ
            2001 TO 2003 10000.0 *= 20000.0
        END NODAL LOAD DATA
    END LOADS DATA
*/
BEGIN SUBSET DEFINITION
  SUBSETS OF STIFFNESS SET 1
    E1 = ALL
    E6 = E1
    N1 = 1001 TO 1003
    E5 = OPEN IN N1
    EXCLUDE E5 FROM E6
    N2 = ALL
  SUBSETS OF STIFFNESS SET 2
    E1 = ALL
    N1 = ALL
  SUBSETS OF STIFFNESS SET 3
    E1 = ALL

```

**SAMPLE DECK
SETUPS**

```
      N1 = ALL /
SUBSETS OF STIFFNESS SET 4 /
      E1 = ALL /
      N1 = ALL /
END SUBSET DEFINITION /
BEGIN INTERACT DATA /
  DEFINE SS 1 AS SET 1 STAGE 1 /
  DEFINE SS 2 AS SET 2 STAGE 1 /
  DEFINE SS 3 AS SET 3 STAGE 1 /
  DEFINE SS 4 AS SET 4 STAGE 1 /
  SS 5 /
  INTERACT 1 2 3 4 /
BEGIN BC CHANGES /
  SS 5 /
  REFERENCE SS 1 /
  RETAIN TX TY FOR 102 TO 116 /
  *4 402 TO 416 /
  *4 1 2 /
  REFERENCE SS 3 /
  RETAIN TX TY FOR 202 TO 216 /
  *4 302 TO 316 /
END BC CHANGES /
  DEFINE HIGHEST SS 5 AS SET 5 /
END INTERACT DATA /
END PROBLEM DATA /
```

***** FND OF FILE *****

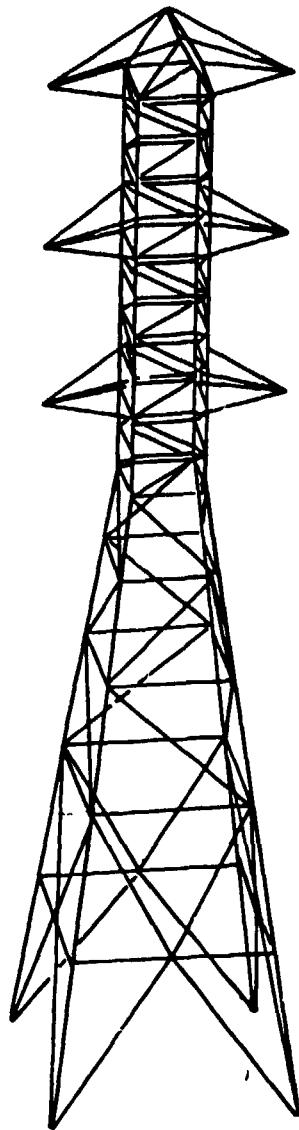


Figure F-5. Transmission Tower Structural Model

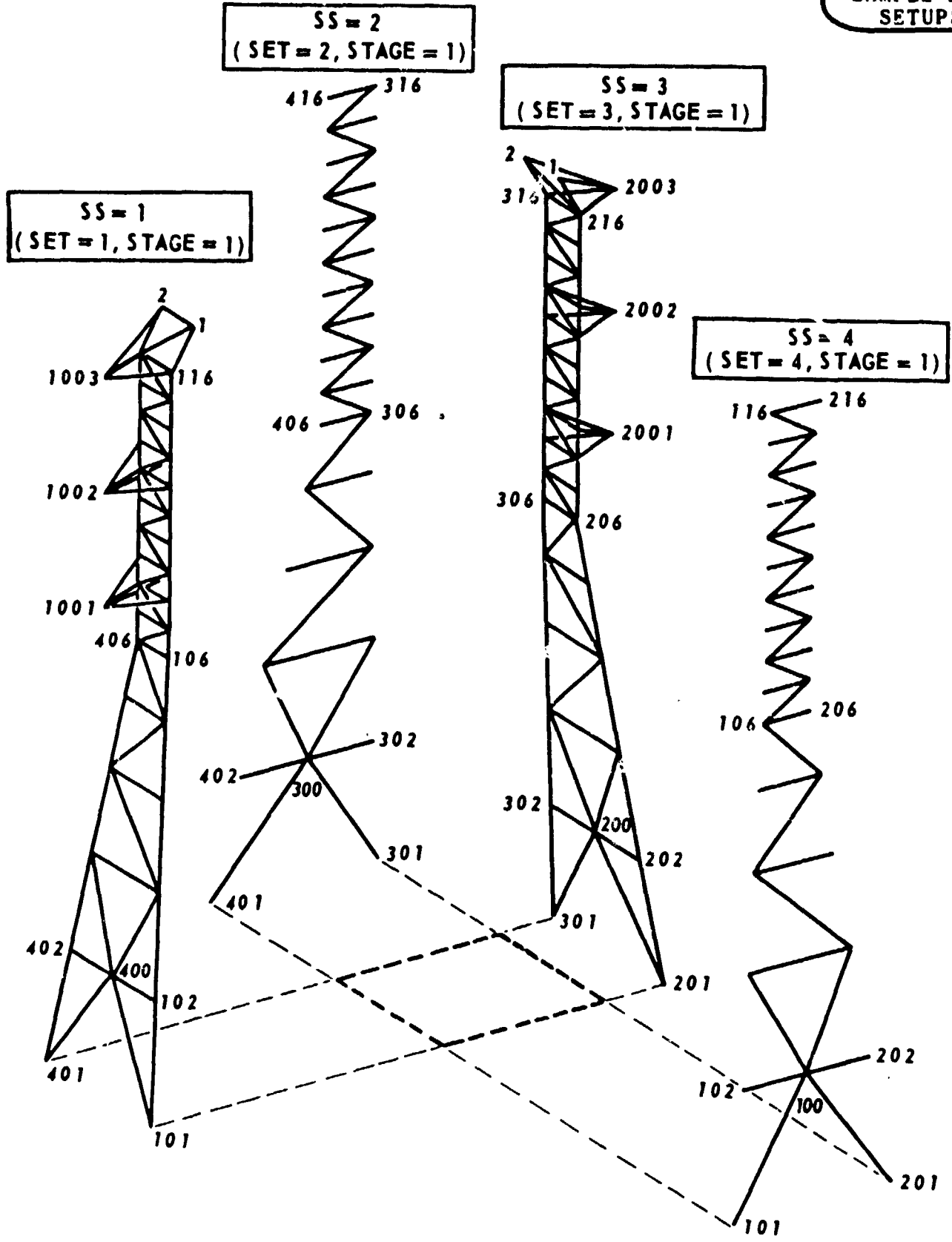
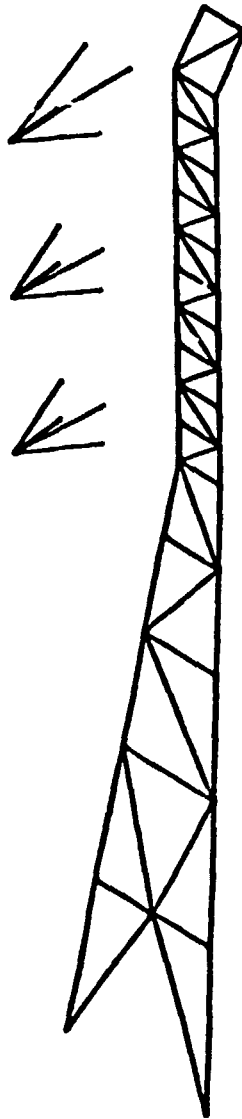


Figure F-6. Substructure Models



EXPLODED VIEW OF SUBSETS
OF STIFFNESS DATA SET 1
NO SCALE

Figure F-7. Exploded View of Substructure 1

APPENDIX G

SPECIAL ATLAS-LIBRARY ROUTINES AND
ATLAS INTERFACES WITH EXTERNAL PROGRAMS

Analyses which are not directly attainable by the ATLAS System Modules may be performed by use of FORTRAN and SNARK statements in an ATLAS Control Program in addition to using any of the ATLAS-System library routines as described in reference 1-1. Some of the library routines provide data management functions that are applicable to modification of ATLAS structural-model input data. Other routines provide automated interfaces of execution and problem-definition data between ATLAS and several computer programs that are external to ATLAS. These interfaces and the capabilities provided for modifying ATLAS structural-model data are described in the following sections.

- G.1 ATLAS STRUCTURAL DATA MODIFICATION ROUTINES-- Capabilities are provided to a) generate nodal and/or stiffness data-record images, one node or element per record, from a preprocessed ATLAS data deck that includes data-generation input records, and b) create two or more ATLAS node/stiffness data sets from a single, parent, node/stiffness data set.
- G.2 NASA-LRC AIRPLANE CONFIGURATION PROGRAM INTERFACE -- Interface of the geometry input data for the NASA-LRC aerodynamic configuration programs (ref. G-1 and G-2) with ATLAS. ATLAS nodal data may be defined by use of the interfaced geometry data.
- G.3 GCS (GEOMETRY CONTROL SYSTEM) PROGRAM INTERFACE -- Interface with ATLAS the surface-definition geometry data saved from a GCS execution (ref. G-3). ATLAS nodal data may be defined by use of the interfaced, GCS-generated geometry data.
- G.4 ATLAS/FLEXSTAB SYSTEM INTERFACES -- Interface of ATLAS-generated, structural and mass data with the FLEXSTAB System (ref. G-4) for performing aeroelastic loads and elastic stability analyses. Additionally, there is an interface of the FLEXSTAB-generated steady-state loads with ATLAS for performing stress analysis and structural-design functions.

**G.5 ATLAS/NASTRAN SYSTEM INTERFACES -- Interface of
ATLAS input data to the NASTRAN System (ref. G-5)
and an interface of NASTRAN input data to ATLAS.**

G.1 ATLAS STRUCTURAL-DATA MODIFICATION ROUTINES

Two ATLAS-library routines identified by the names ATEXP and CUTNE provide data-management functions applicable to modification of ATLAS structural-model input data. These two routines, as described in the next sections, provide the following capabilities:

ATEXP --ATLAS nodal and/or stiffness input data-set records are generated from data matrices on file DATARNF in the ATLAS data base. The generated data records are such that each node and each stiffness element used to define the structural model are output on separate data records.

CUTNE --Two or more ATLAS node/stiffness data sets are created from a single, parent, node/stiffness data set which was input to ATLAS. The parent data set is defined by its corresponding data matrices on file DATARNF in the ATLAS data base. Each new data set is established via node and element subsets of the parent set.

G.1.1 The ATLAS-Library Routine ATEXP

Generation of ATLAS nodal and/or stiffness input data-record images from matrices in the ATLAS data base may be effected by use of the routine ATEXP. This routine uses those matrices in file DATARNF that correspond to a particular node and/or stiffness data set. These matrices are generated by previous execution of the NODAL and STIFFNESS Preprocessors (sec. 146.0 and 152.0). The data-record images generated by ATEXP are output onto one of the ATLAS Savefiles for subsequent use.

The format of each output data-record image is such that each node and each stiffness element used to define the structural model is on a separate record. This capability thus provides a means by which nodal and/or stiffness data-generation input records can be expanded for interrogation and/or modification by the ATLAS Interactive Control Module (sec. 200.5). An option is provided to replace the nodal and/or stiffness data sets within an ATLAS data deck by the corresponding, expanded data sets. All other data sets within a data deck are not modified by ATEXP.

Subroutine ATEXP is called from a Control Program by the following statement.

CALL ATEXP (Se,Elnum,Elements,Nodes,Afile,Bfile,0,0,0,0,0,0)

- Se = The number (integer) of the node and/or stiffness data set for which data are to be processed.

- Elnum = A switch (integer) input as 0 or 1 which denotes the following:
 - 0--Include element-number identifiers in the generated data records.
 - 1--Do not include element numbers in the generated data records.

- Elements }
Nodes } = These parameters denote that the element and/or the nodal data are to be processed. Each of these parameters is input as the integer 0 or 1 as follows:
 - 0--Do not process the corresponding type of data.
 - 1--Process the corresponding type of data.

- Afile = One of the ATLAS Savefiles (SAVESS1 through SAVESS4) containing the record images of a data deck. If "Afile" is non-zero, the resulting "Bfile" will include the total data deck from "Afile." File "Afile" is not changed by ATEXP. If this option is not required, "Afile" should be specified as the integer zero.

Caution: If "Afile" is specified as one of the ATLAS Savefiles, it must be different from "Bfile" and it must be different from the input data file "infile" specified in the ATLAS Job Control Card (sec. 11.2).

- Bfile = One of the ATLAS Savefiles (SAVESS1 through SAVESS4) onto which the expanded data-record images are

written. Complete nodal and/or stiffness input data-sets are generated. Nodes are ordered according to increasing user node-numbers, whereas elements are ordered according to increasing user element-numbers.

The following example Control Program illustrates the statements required to generate an ATLAS data file with expanded nodal and stiffness input data sets.

```

BEGIN CONTROL MATRIX PROGRAM EXPAND
PROBLEM ID (GENERATE NODE AND STIFFNESS ELEMENT RECORDS)
DIMENSION A0(115),A1(115),A2(115),A3(115)
DIMENSION A4(115),FET(50,2)
C PREPROCESS ATLAS DATA DECK.
READ INPUT
C OPEN FILES WHICH ARE USED BY THE ROUTINE ATEXP.
CALL FILEADD(FET,DATARNF,SC00RNF)
CALL FETADD(SC00SSF,A0,115,1,0,IRR)
CALL FETADD(SC01SSF,A1,115,1,0,IRR)
CALL FETADD(SC02SSF,A2,115,1,0,IRR)
CALL FETADD(SC03SSF,A3,115,1,0,IRR)
CALL FETADD(SAVESS2,A4,115,1,0,IRR)
C CALL ROUTINE ATEXP TO GENERATE DATA RECORDS FOR SET 5.
CALL ATEXP(5,0,1,1,0,SAVESS2,0,0,0,0,0,0)
END CONTROL PROGRAM

```

Execution of "READ INPUT" results in data matrices being written onto file DATARNF of the ATLAS data-base. All of the working files required by the subroutine ATEXP must be opened as shown by the calls to FILEADD and FETADD. Only the output file of ATEXP, SAVESS2 for this example, may be different from that shown. The expanded data file, SAVESS2 for this example, may be saved on magnetic tape for subsequent use via the following CDC 6600 Control Cards placed after the ATLAS-execution Control Card.

```

REWIND(SAVESS2)
REQUEST(CARDS,F=I,LB=KL,PO=AW) SAVE
COPYEI(SAVESS2,CARDS)
RETURN(CARDS)

```

G.1.2 The ATLAS-Library Routine CUTNE

Creation of two or more ATLAS node/stiffness data sets from a single, parent, node/stiffness data set may be effected by use of the routine CUTNE. The parent set and the new sets are manipulated/created via data matrices in the ATLAS data base.

The CUTNE routine uses those matrices in file DATARNF which are generated for the parent set by the NODAL and STIFFNESS Preprocessors (sec. 146.0 and 152.0). The new data sets are established via node and element subsets of the parent set as defined by the SUBSET-DEFINITION Preprocessor (sec. 156.0).

The stiffness and corresponding node data sets created by CUTNE may be used as structural models for any ATLAS analyses. This capability provides, for example, a means by which multiple substructures can be "cut" from a single, previously-defined, parent, structural model.

Subroutine CUTNE is called from a Control Program by the following statement.

```
CALL CUTNE (Pset,n,Nsub,Esub,File,Buff,Irr)
```

Pset	=	The number (integer) of a node and stiffness data set from which new data sets are to be created.
n	=	A node/stiffness data-set number (integer) in the range 1 to 36 to be assigned to the newly-created SET. The node/stiffness set numbers used in a single job must be different.
Nsub	=	Node subset number (integer) associated with the parent set "Pset." The nodes in "Nsub" are to be included in the new set "n." Subset Nxxx must include at least those nodes associated with the elements identified by "Esub."
Esub	=	Stiffness-element subset number (integer) associated with the parent set "Pset." The elements in "Esub" are to be included in the new set "n."
File	=	The filename DATARNF
Buff	=	The data-block size of the structural-element matrices in DATARNF. This parameter should be specified as the integer 2500

Irr = This parameter is an error-switch (integer) returned from CUTNE as follows:

- ≤0--Successful execution
- 1--"Pset" data matrices not in "File."
- 2--Set "n" was defined previously.
- 3--"Nsub" was not defined for "pset."
- 4--"ESUB" was not defined for "pset."
- 5--"Nsub" does not include those nodes associated with the elements in "Esub."

If Irr>0, the new data set "n" is not created.

The following example Control Program illustrates the statements required to execute the subroutine CUTNE.

```

BEGIN CONTROL MATRIX PROGRAM CUTTER
PROBLEM ID (CREATE 3 NEW DATA SETS FROM SET 2)
DIMENSION FET(50)
C LOAD ATLAS DATA MATRICES FOR SET 2.
LOAD FILES (DATARNF)
C DEFINE NODE AND ELEMENT SUBSETS OF SET 2.
C THESE SUBSETS ARE USED TO ESTABLISH
C THE NEW NODE/ELEMENT SETS 3, 4 AND 5.
READ INPUT
C OPEN DATARNF FILE FOR USE BY THE ROUTINE CUTNE.
CALL FILEADD(FET,DATARNF)
C CALL ROUTINE CUTNE TO CREATE NEW DATA SETS.
CALL CUTNE (2,3,600,610,DATARNF,2500,IRR)
CALL CUTNE (2,4,601,611,DATARNF,2500,IRR)
CALL CUTNE (2,5,602,612,DATARNF,2500,IRR)
C SAVE GENERATED DATA FOR SUBSEQUENT USE.
SAVE FILES (SAVESS1)
C PRINT DATA FOR THE NEW DATA SETS.
PRINT INPUT (NODAL,SET=3)
PRINT INPUT (NODAL,SET=4)
PRINT INPUT (NODAL,SET=5)
PRINT INPUT (STIFFNESS,SET=3)
PRINT INPUT (STIFFNESS,SET=4)
PRINT INPUT (STIFFNESS,SET=5)
END CONTROL PROGRAM

```

A new node/stiffness data set is created by each call to CUTNE. Input data-record images may be generated for the new data sets, if required, by using the routine ATEXP described in section G.1.1.

G.2 NASA-LRC AIRPLANE CONFIGURATION PROGRAM INTERFACE

Geometry input data for the NASA-LRC airplane configuration plot program (ref. G-1) or the geometry input data for the NASA supersonic-aircraft, aerodynamic design and analysis system (ref. G-2) are converted to an equivalent, ATLAS, input data-deck by the LRCGEOM subroutine in the ATLAS library. The input, geometry data (LRCGEOM data) for each of the NASA programs have similar formats as described in the referenced documents. The generated ATLAS data deck is comprised of data-record images that are used as input to the GEOMETRY Preprocessor (sec. 126.0). ATLAS nodal data (sec. 146.0) may be defined subsequently by use of the interfaced geometry data.

LRCGEOM data define a structural configuration that is comprised of a number of components. Each component is identified as a wing, body (fuselage), pod (nacelle), fin (vertical tail) or a canard (horizontal tail). The correspondence between the LRCGEOM component data and the equivalent ATLAS component-geometry data is shown in table G-1.

Table G-1. Correspondence of LRCGEOM Data to the Converted, Equivalent, ATLAS Geometry Components

LRCGEOM COMPONENT	EQUIVALENT ATLAS GEOMETRY COMPONENT(S) ①		
	LABEL(S)	INPUT-FRAME COORDINATE SYSTEMS	LONGITUDINAL CONTROL CURVES
WING	WINGR WINGL		LC1 and LC2 enriched at XAF locations of LRCGEOM specified airfoil ordinates.
CIRCULAR SECTION BODY (FUSELAGE)	BODYi (i = A, B, C, ...)		LC1 and LC2 enriched at 1.0, .99, .98, .97, .96, .95, .94, .92, .90, .88, .86, .84, .82, .80, .75, .70, .60, .50, .40, .30, .25, .20, .18, .16, .14, .12, .10, .08, .06, .05, .04, .03, .02, .01, .00
ARBITRARY SECTION BODY (FUSELAGE)	BODYi (i = A, B, C, ...)		Curves generated through each point used to define each half section. Right Side: RSA, RSB, ..., RSj Left Side: LSB, LSC, ..., LSj
POD (MACELLE)	PODiR PODiL (i = A, B, C, ...)		LC1 and LC2 enriched at 1.0, .99, .98, .97, .96, .95, .94, .92, .90, .88, .86, .84, .82, .80, .75, .70, .60, .50, .40, .30, .25, .20, .18, .16, .14, .12, .10, .08, .06, .05, .04, .03, .02, .01, .00
FIN (VERTICAL TAIL)	FINi or FINiR FINiL		LC1 and LC2 enriched at XFIN locations of LRCGEOM specified airfoil ordinates.
CANARD (HORIZONTAL TAIL)	CANRDiR CANRDiL (i = A, B, C, ...)		LC1 and LC2 enriched at XCAN locations of LRCGEOM specified airfoil ordinates.

① The LRCGEOM coordinate system is identified by (x, y, z) , whereas the ATLAS input reference frame axes are identified by (x, y, z) . An LRCGEOM component that is not symmetrical relative to the $x-z$ plane is converted to a right (R) and left (L) ATLAS geometry component. The ATLAS components are all "simple" except for a body with arbitrary cross section which is a "detailed" component as defined in Section 126.0. The letter "i" is assigned automatically to the i -th component of a particular type as ordered within LRCGEOM input data. When extracting nodal coordinates via nodal data (Sec. 146.0), the name of the input frame must correspond to the assigned component label (e.g. WINGR).

G.2.1 Job-Deck Structure

Figure G-1 illustrates a typical ATLAS job deck for interfacing LRCGEOM data with ATLAS.

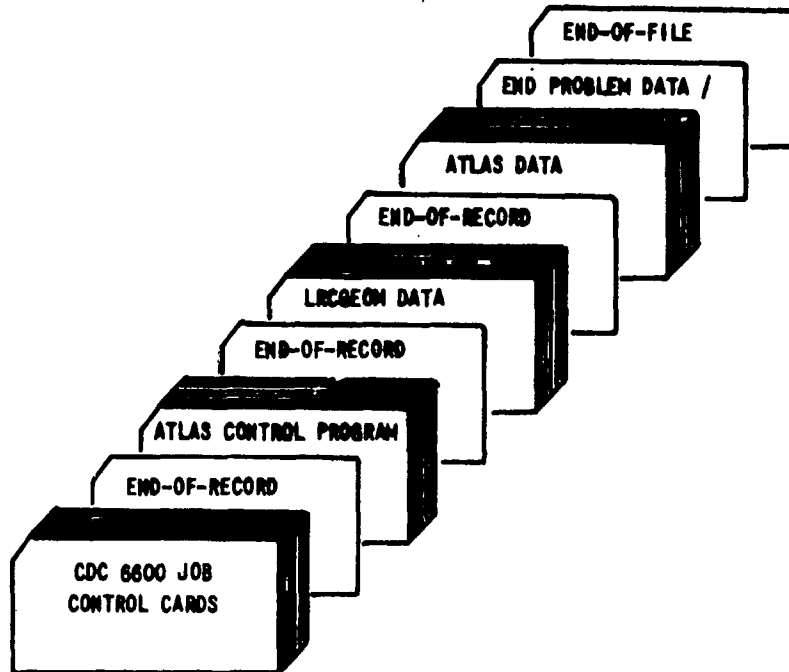


Figure G-1. ATLAS Job Deck with LRCGEOM Data

G.2.2 CDC 6600 Job Control Cards

The standard control cards for execution of ATLAS using the CALL(CATM,KONTROL) statement as described in section 11.0 may be used.

G.2.3 ATLAS Control Program

The following Control Program illustrates the statements required to convert LRCGEOM data to "equivalent" ATLAS component-geometry data and execute the ATLAS System.

```

BEGIN CONTROL MATRIX PROGRAM GEOM
PROBLEM ID (LRCGEOM DATA INTERFACE WITH ATLAS)
C OPEN FILES THAT ARE USED BY THE ROUTINE LRCGEOM.
  DIMENSION A1(115),A2(115)
  CALL FETADD(SAVESS1,A1,115,1,0,IRR)
  CALL FETADD(SAVESS2,A2,115,1,0,IRR)
C READ LRCGEOM DATA FROM FILE SAVESS1, CONVERT TO EQUIVALENT
C   ATLAS GEOMETRY INPUT DATA AND STORE ON FILE SAVESS2.
  CALL LRCGEOM(SAVESC1,SAVESS2)
C PREPROCESS THE CONVERTED DATA BY THE ATLAS GEOMETRY
C   PREPROCESSOR.
  REWIND SAVESS2
  READ INPUT(I=SAVESS2)
C OTHER ATLAS STATEMENTS MAY BE USED HERE AS REQUIRED.
  END CONTROL PROGRAM

```

Subroutine LRCGEOM is called according to the following statement.

```
CALL LRCGEOM (Afile,Bfile)
```

Afile = Name of the ATLAS Savefile (SAVESSF or SAVESS1 through SAVESS4) containing the LRCGEOM data to be converted.

Bfile = Name of the ATLAS Savefile onto which is written the equivalent, ATLAS, geometry-input data deck. These data are preprocessed by ATLAS via the statement "READ INPUT (I=Bfile)."

If the LRCGEC. data are input on cards as shown in figure G-1, "Afile" should be specified as the word 5LINPUT and the corresponding call to FETADD shown in the foregoing Control Program should not be used.

G.2.4 LRCGEOM and ATLAS Input Data

The format of the LRCGEOM input data is described in references G-1 and G-2. No ATLAS input data are required for execution of the subroutine LRCGEOM. ATLAS nodal data (sec. 146.0), however, are used to define nodes for the interfaced geometry components. The nodal input reference frame used for extraction of nodal coordinates must correspond to the coordinate system used to define the LRCGEOM data as shown in table G-1. The ATLAS reference frame must also be identified by the same name (label) that is assigned to the component as shown in table G-1. ATLAS finite elements are defined via the extracted nodes for subsequent analysis/design of the model by the ATLAS System.

G.3 GCS (GEOMETRY CONTROL SYSTEM) PROGRAM INTERFACE

Geometry data generated by execution of the GCS program (ref. G-3) are interfaced with ATLAS via the GCSGEOM subroutine in the ATLAS library. This routine converts the GCS-generated data to equivalent, ATLAS geometry data and stores the converted data in file DATARNF in the ATLAS data base. ATLAS nodal data (sec. 146.0) may be defined directly by use of the interfaced geometry data.

Each geometry component defined via GCS is identified either as a "wing" or a "body." A "wing" is converted to an ATLAS "simple" geometry component, whereas a "body" is converted to an ATLAS "detailed" geometry component as described in section 126.0. There is a one-to-one correspondence between the GCS and ATLAS geometry-component coordinate systems and methods of identifying longitudinal control-curves. Therefore, coordinates for ATLAS nodes may be extracted directly from the interfaced geometry data as described in section 146.1.2.

G.3.1 Job Deck Structure

A typical job deck for interfacing GCS data with ATLAS is illustrated by figure G-1 provided that the LRCGEOM data records shown in that figure are replaced by GCS data records saved from a GCS-program execution.

G.3.2 CDC 6600 Job Control Cards

The standard control cards for execution of ATLAS using the CALL(CATM,KONTROL) statement as described in section 11.0 may be used.

G.3.3 ATLAS Control Program

The following Control Program illustrates the statements required to convert GCS-generated data to "equivalent" ATLAS component-geometry data and execute the ATLAS System.

```
BEGIN CONTROL MATRIX PROGRAM GCSNODE
PROBLEM ID (GCS GEOMETRY DATA INTERFACE WITH ATLAS)
C OPEN FILES THAT ARE USED BY THE ROUTINE GCSGEOM.
  DIMENSION A1(115)
  CALL FETADD(SAVESS1,A1,115,1,0,IRR)
C READ GCS-GENERATED DATA FROM FILE SAVESS1,
C   CONVERT TO EQUIVALENT ATLAS GEOMETRY
C   DATA AND STORE ON FILE DATARNF.
  ID=6HCOMP39
  CALL GCSGEOM(SAVESS1,ID)
C ATLAS NODES MAY NOW BE DEFINED FOR THE GEOMETRY
```

```

C           COMPONENT IDENTIFIED AS COMP39.
          READ INPUT
C           OTHER ATLAS STATEMENTS MAY BE USED HERE AS REQUIRED.
          END CONTROL PROGRAM
  
```

Subroutine GCSGEOM is called according to the following statement.

```
CALL GCSGEOM (Afile,label)
```

Afile	=	Name of the ATLAS Savefile (SAVESSF or SAVESS1 through SAVESS4) containing the GCS-generated data for a geometry component to be converted.
label	=	An alphanumeric word of 1 to 7 characters that identifies the "equivalent" ATLAS geometry component. The selected word must be preceded by the letter "H" and the number of characters in the word.

If the GCS-generated data are supplied on cards as discussed in section G.3.1, "Afile" should be specified as the word 5LINFUT and the DIMENSION statement and call to FETADD shown in the foregoing Control Program should not be used.

G.3.4 GCSGEOM and ATLAS Input Data

The data saved from an execution of the GCS program for definition/generation of a geometry component are described in reference G-3. No ATLAS input data are required for execution of the subroutine GCSGEOM. ATLAS nodal data (sec. 146.0), however, are used to define nodes for the interfaced geometry components. The nodal input reference frame used for extraction of nodal coordinates must correspond to the coordinate system used to define the GCS data and must be identified by the same name that is assigned to the component by "label" in the call to GCSGEOM.

ATLAS finite elements are defined via the extracted nodes for subsequent analysis/design of the model by the ATLAS System.

G.4 ATLAS/FLEXSTAB SYSTEM INTERFACES

The execution and input data required for the interfaces between the ATLAS System and programs within the FLEXSTAB System (ref. G-4) that provide steady-state loads and elastic stability analysis capabilities are described herein. The interfaces from ATLAS to FLEXSTAB and from FLEXSTAB to ATLAS in addition to the use of the ATLAS-library routine VAMAT (ref. G-6) are described in terms of the CDC job-control card, ATLAS System Control-Program and input-deck setup requirements. The following capabilities are provided:

- a) ATLAS to FLEXSTAB: Generate a tape on which ATLAS-generated structural and mass data are written in a format that is compatible with the data-input requirements of the ESIC (External Structural Influence Coefficient) Program of FLEXSTAB. These data are subsequently used by FLEXSTAB in performing aeroelastic loads and elastic stability analyses.
- b) FLEXSTAB to ATLAS: Read a tape on which FLEXSTAB-generated nodal-loads data are written to perform a stress analysis via the ATLAS System. The net loads at retained ATLAS nodes are calculated by the SDSS (Stability Derivatives and Static Stability) Program of FLEXSTAB and written onto the SDSS output tape.
- c) The ATLAS-Library Routine VAMAT: The ATLAS-generated panel-weight matrix associated with the aerodynamic panels defined for FLEXSTAB is used to integrate panel loads by subroutine VAMAT in the ATLAS-System library. This routine generates resultant shear forces, bending moments and torsions relative to user-defined reference axes. These resultant loads are not normally used explicitly in structural design, however, they are used in the selection of critical design-load conditions.

G.4.1 ATLAS to FLEXSTAB Interface

The ATLAS to FLEXSTAB interface produces an input data tape called NASTAP for the ESIC program of FLEXSTAB. NASTAP contains structural geometry and mass data in the first file and structural flexibility matrices associated with symmetric and antisymmetric aeroelastic analyses in subsequent files. Units of the mass data must be lb-sec²/ft. A description of the format of NASTAP may be found in reference G-4. The first file of NASTAP is produced by execution of the MASSFIL subroutine, whereas subsequent files are generated by execution of the FLEXFIL subroutine in the ATLAS-System library (ref.

1-1). Data for only one weight condition may be written onto NASTAP.

G.4.1.1 Job Deck Structure

Figure G-2 illustrates the ATLAS job deck required to generate the ATLAS to FLEXSTAB interface data tape.

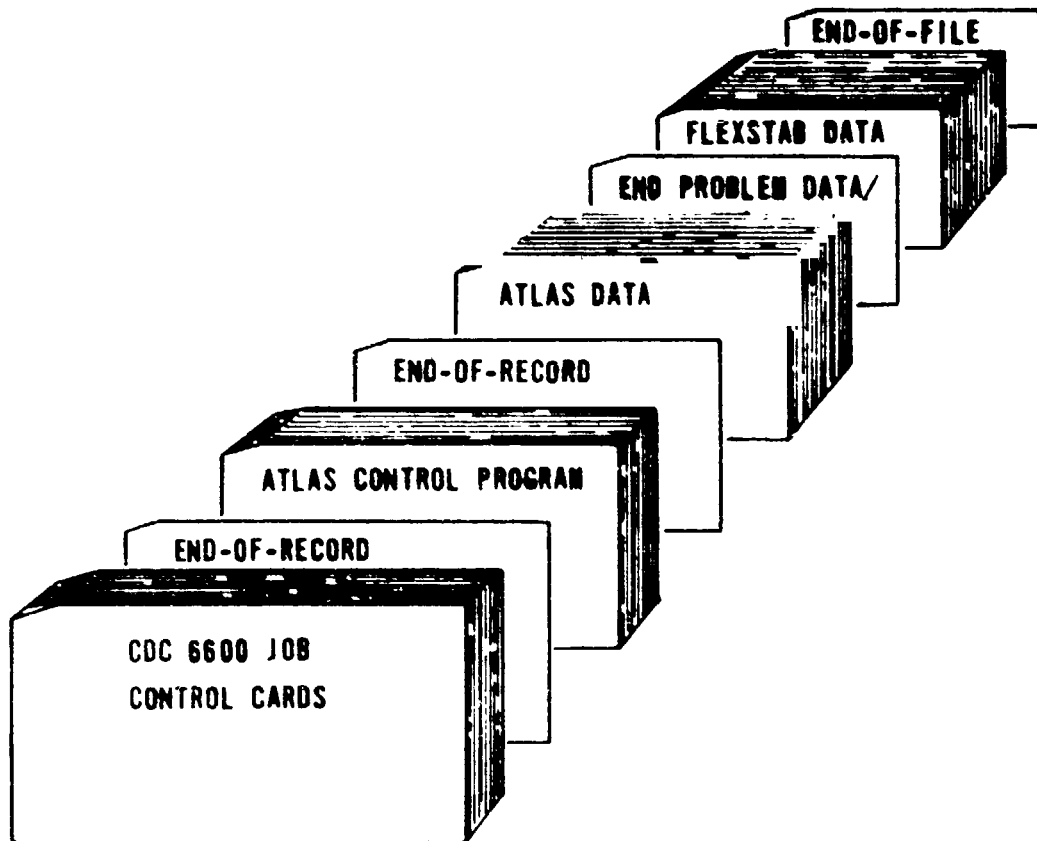


Figure G-2. ATLAS/FLEXSTAB Job Deck

G.4.1.2 CDC 6600 Job Control Cards

The following list shows the control cards required to generate the data tape NASTAP for subsequent aeroelastic analyses via FLEXSTAB.

```
Job Card
Account Card
GET(CATM/UN=UATLASU)
CALL (CATM,ATTACH)
CALL (CATM,KONTROL)
ATLAS.
REWIND (SAVESSF)
REQUEST (NASTAP,F=I, LB=KL,PO=AW) SAVE
COPYEI (SAVESSF,NASTAP)
RETURN (NASTAP)
```

G.4.1.3 ATLAS Control Program

The following Control Program illustrates the statements required to generate NASTAP for both a symmetric and an antisymmetric analysis.

```
BEGIN CONTROL MATRIX PROGRAM FLEX
PROBLEM ID (ATLAS/FLEXSTAB INTERFACE)
DIMENSION FET(50,3),FETS(560)
READ INPUT
EXECUTE MASS (OPTION=2)
C
C STAGE 1 --SYMMETRICAL
C STAGE 2 --ANTISYMMETRICAL
C
PERFORM F-REDUCE (STAGE=1,[ FRED]=[ FLEX1 ])
PERFORM F-REDUCE (STAGE=2,[ FRED]=[ FLEX2 ])
CALL FILEADD (FET,DATARNF,MASSRNF,MULTRNF)
CALL FETADD (SAVESSF,FETS,560,1,0,IRR)
REWIND SAVESSF
NSET=Stiffness/mass data set number
NSTAGE=1
INMASS=Mass matrix name
CALL MASSFIL (INMASS,NSET,NSTAGE)
INSYMM=5LFLEX1
CALL FLEXFIL (INSYMM)
INASYM=5LFLEX2
CALL FLEXFIL (INASYM)
END CONTROL PROGRAM
```

Subroutines MASSFIL and FLEXFIL are called with user-defined parameters as described below. The first file of information on NASTAP is generated by the subroutine MASSFIL, the

second and subsequent data files on NASTAP are generated by FLEXFIL--one file per subroutine call.

CALL MASSFIL (Name,Set,Stage)

Name = Name of the mass matrix associated with the mass condition number as assigned by the mass data (sec. 138.0).

Set = The number (integer) of the stiffness/mass data set.

Stage = Integer identifying a boundary-condition STAGE for symmetrical boundary conditions associated with "Set" (sec. 106.0).

CALL FLEXFIL (Name)

Name = Name previously assigned to the flexibility matrix.

If two flexibility matrices are required (one for a symmetrical analysis and one for an antisymmetrical analysis), two calls to FLEXFIL are required. In the foregoing Control Program, the matrices are named FLEX1 and FLEX2, respectively.

G.4.1.4 FLEXSTAB Aerodynamic Model Data

Structural FLEXSTAB-model data are required and must be stacked immediately following the "END PROBLEM DATA" record of the ATLAS data deck. These data are comprised of the following data blocks:

- a) Slender body data
- b) Vertical and horizontal thin-body data including the corresponding panel-definition data.

Each slender body and thin body, as required for the FLEXSTAB aerodynamic model, must be defined by one of these data blocks. The order in which the bodies are defined must be compatible with the order in which the corresponding kinematic freedoms are RETAINED by the ATLAS boundary-condition (BC) data (sec. 106.0). That is, the order of the retained degrees-of-freedom associated with the input-sequence of body data must coincide with the order in which the same degrees-of-freedom are retained by the BC data.

The analysis frame of all retained nodes must be the GLOBAL reference frame (see sec. 100.0) oriented such that the positive X-axis is in the direction of the free-stream flow and the positive Z-axis is upward. The X-Z plane is assumed to be the plane of model symmetry. All coordinates defined by the following data records must be measured relative to the GLOBAL X-Y-Z triad.

The following data blocks are input, as required, according to the foregoing stacking requirement. Each block starts with a "BEGIN" record and terminates with an "END" record. Detailed descriptions of the following data items are included in the FLEXSTAB System documentation (ref. G-4).

SLENDER BODY DATA

Record 1 Begin Slender-Body Data Block

BEGIN SLENDER BODY DATA

Record 2 Slender Body Data

y z Symm Asym Code Nodes Mass N dof Flist

- y = Y-coordinate defining the origin of the slender-body for the FLEXSTAB model.
- z = Z-coordinate defining the origin of the slender-body local coordinate system used for the FLEXSTAB model.
- Symm = The row number (integer) of the flexibility matrix that corresponds to the structural node associated with the beginning of this body when a symmetrical analysis is to be performed. A zero must be input for this item if a symmetrical analysis is not to be performed.
- Asym = Same as "Symm" but must be specified as a nonzero integer when an antisymmetrical analysis is to be performed.

Both "Symm" and "Asym" may be specified as nonzero. One of these items must be nonzero.

- Code = An integer that denotes the location of the body relative to the X-Z plane of model symmetry. The options are:
- 0--the body is not in the X-Z plane
1--the body is in the X-Z plane
- Nodes = The number (integer) of retained nodes on the body.
- Mass = The row number (integer) of the mass matrix associated with the beginning of the body.
- Ndof = The number (integer) of degrees of freedom retained at each node on the body ($1 \leq \text{Ndof} \leq 3$).
- Flist = A list of 1, 2 or 3 unique letters selected from X, Y and Z that denotes which translational degrees-of-freedom are retained at each node on this body. The order in which these letters are input must be compatible with the order in which the freedoms are retained.

Record 3 End Slender-Body Data Block

END DATA

THIN BODY AND PANEL DEFINITION DATA

The following sequence of records is used to define a vertical thin body or a horizontal thin body.

Record 4 Begin Thin-Body Data Block

BEGIN { VERTICAL } THIN BODY DATA
 { HORIZONTAL }

This record denotes whether the subsequent records define a VERTICAL thin body or a HORIZONTAL thin body.

Record 5 Thin Body Data

x y z dih Symm Asym Code Nodes
Npan Mass Ndof Flist

x,y,z = The X, Y and Z-coordinates defining the origin of the body local coordinate system used for the FLEXSTAB model.

dih = Dihedral angle (radians) of the thin body. A positive dihedral corresponds to a right-hand rotation about the X-axis relative to the X-Y plane.

Symm }
Asym } = These items are as described for
Code } Record 2.
Nodes }

Npan = The number (integer) of structural panels used for FLEXSTAB interpolation purposes. These panels are defined by subsequent Records 6 and 7 for this body.

Mass }
Ndof } = These items are as described for
Flist }

Record 6 Begin Panel Definition Data

BEGIN PANEL DEFINITION DATA

Record 7 defines a panel associated with the thin body defined by Records 4 and 5. This record must be repeated to define "Npan" panels for the body.

Record 7 Panel Definition Data

N1 B1 N2 B2 N3 B3 N4 B4

Ni,Bi = Ni identifies the j-th sequential retained node for the body identified by the integer number "Bi." The "Ni" integer numbers must be in the range 1 to "Nodes" as specified by Record 5. Items N1, N2, N3 and N4 define the 3 or 4 corners of a triangular or quadrilateral structural panel,

respectively, that is used for FLEXSTAB interpolation purposes. N1 through N4 may be input in either a clockwise or a counterclockwise order. N4 and B4 must be input as zeros for a triangular panel.

Record 8 End Thin-Body Data Block

END DATA

Record 9 End Data

END FLEXSTAB DATA

This record indicates that all slender-body and thin-body aerodynamic model data have been input for the SET/STAGE structural model for which aeroelastic loads are to be calculated.

G.4.2 FLEXSTAB to ATLAS Interface

The FLEXSTAB to ATLAS interface consists of processing the FLEXSTAB-generated net-loads data file by subroutine STREFIL in the ATLAS-System library (ref. 1-1). This routine generates structural nodal-loads matrices used for stress analyses via ATLAS.

G.4.2.1 Job Deck Structure

Figure G-3 illustrates the ATLAS job deck required to generate nodal-loads matrices and to perform a stress analysis based on the FLEXSTAB net-loads data.

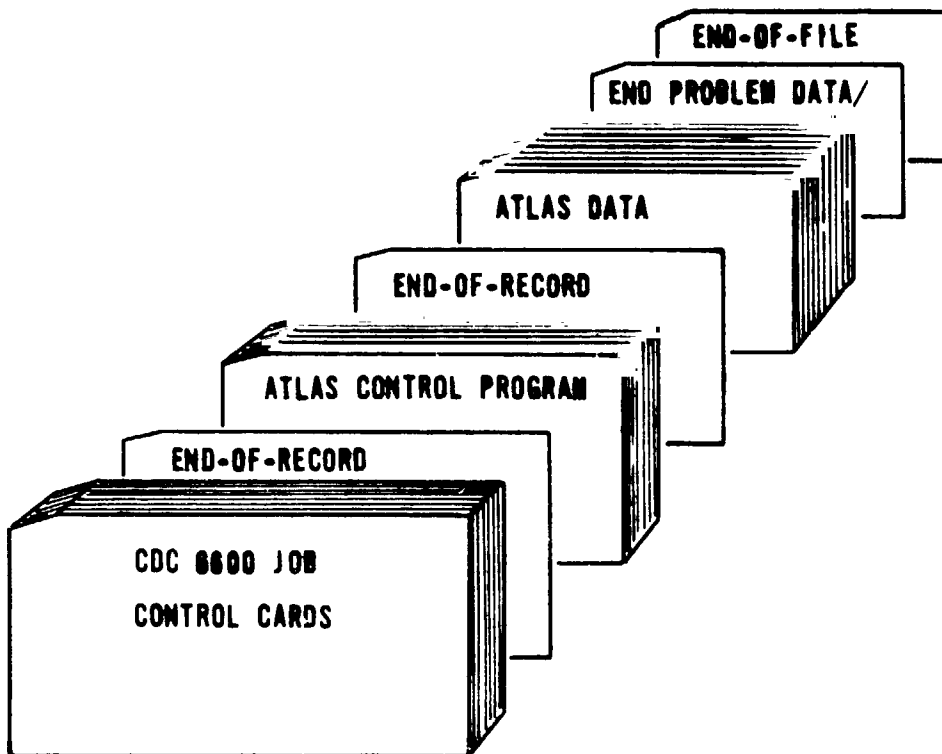


Figure G-3. FLEXSTAB/ATLAS Job Deck

G.4.2.2 CDC 6600 Job Control Cards

The following list shows the control cards required to execute ATLAS using the output tape from FLEXSTAB. This tape is identified as "SDD" in the subsequent cards.

```

Job Card
Account Card
GET (CATM/UN=UATLASU)
CALL (CATM,ATTACH)
CALL (CATM,KONTROL)
REQUEST (SDD,VSN=66xxxx,PO=AR,F=I,LB=KU)
REWIND (SDD)
COPY (SDD,SAVESSF)
RETURN (SDD)
REWIND (SAVESSF)
ATLAS.

```

G.4.2.3 ATLAS Control Program

The following Control Program illustrates the statements required to compute stresses and displacements via ATLAS using the FLEXSTAB nodal loads.

```

BEGIN CONTROL MATRIX PROGRAM FLEX
PROBLEM ID (FLEXSTAB/ATLAS INTERFACE)
DIMENSION FET (50), FETS (560)
CALL FILEADD (FET,DATARNF)
CALL FETADD (SAVESSF,FETS,560,1,0,IRR)
CALL STREFIL
READ INPUT

C
C   STAGE 1 -- SYMMETRIC
C
PERFORM R-STRESS
PRINT OUTPUT (STRE,Plist)
PRINT OUTPUT (DISP,Plist)

C
C   STAGE 2 -- ANTISYMMETRIC
C
PERFORM R-STRESS (STAGE=2)
PRINT OUTPUT (STRE,STAGE=2,Plist)
PRINT OUTPUT (DISP,STAGE=2,Plist)
END CONTROL PROGRAM

```


Within this Control Program, subroutine STREFIL is called as follows:

```
CALL STREFIL
```

G.4.2.4 ATLAS Loads Data

The following ATLAS loads data are required.

```
BEGIN LOADS DATA /  
STAGE 1 / SYMMETRIC CONDITION  
READ NODAL LOADS FROM DATARNF WITH INDEX QSYMM**/  
*/ THE FOLLOWING TWO CARDS ARE REQUIRED IF ANTISYMMETRICAL  
LOADS ARE ALSO AVAILABLE. /  
STAGE 2 / ANTISYMMETRICAL CONDITION  
READ NODAL LOADS FROM DATARNF WITH INDEX QASYM** /  
END LOADS DATA /
```

G.4.3 The ATLAS-Library Routine VAMAT

Subroutine VAMAT allows the shear, moment and torsion along the wing, body and horizontal tail to be calculated for a selected set of panel airloads and panel weights for symmetric conditions only. An additional routine identified by the name VAMSCN is provided to scan the VAMAT-generated load conditions to find the minima and maxima. Further descriptions of VAMAT and VAMSCN are presented in reference G-6.

G.4.3.1 Job Deck Structure

Figure G-4 illustrates the ATLAS job deck required to execute the VAMAT and VAMSCN routines.

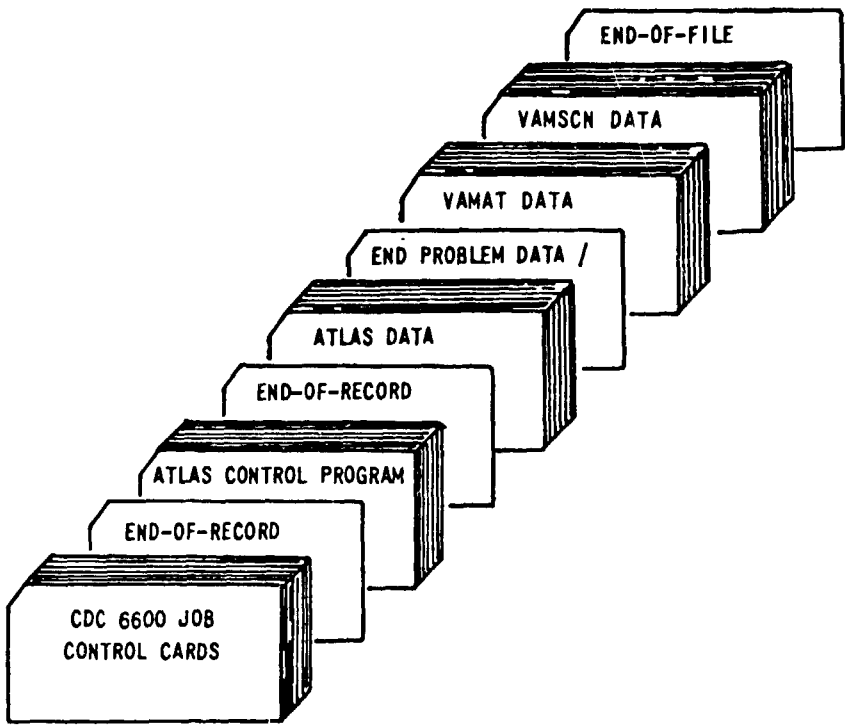


Figure G-4. ATLAS Job Deck with VAMAT and VAMSCN Data

G.4.3.2 CDC 6600 Job Control Cards

The following list shows the control cards required to execute the VAMAT and VAMSCN routines.

Job Card
Account Card
GET (CATM/UN=UATLASU)
CALL (CATM, ATTACH)
CALL (CATM, KONTROL)
(1) REQUEST (SDD, VSN=66xxxx, PO=AR, F=I, LB=KU)
(1) FEWIND (SDD)
(1) COPY (SDD, SAVESSF)
(1) FETURN (SPD)
(2) REQUEST (SCI, VSN=66xxxx, PO=AR, F=I, LB=KU)
(2) REWIND (SCI)
(2) COPY (SCI, SAVESS1)
(2) RETURN (SCI)
ATLAS.
(3) FEQUEST (SCO, FO=AW, F=I, LB=KU) SAVE
(3) FEWIND (SAVESS1)
(3) CCPY (SAVESS1, SCO)
(3) FETURN (SCO)

- Notes:
- (1) Control cards required to mount the FLEXSTAB SDSS data tape for execution of VAMAT. File SAVESSF is rewound by VAMAT. Therefore, no information other than the SDSS data may be on SAVESSF.
 - (2) Control cards required to mount the save tape from a previous VAMAT execution. Positioning of SAVESS1 is left to the user since additional conditions may be generated by additional calls to VAMAT prior to executing VAMSCN. These cards are required if additional conditions are to be generated or if VAMSCN is to be executed without a prior call to VAMAT in the ATLAS Control Program.
 - (3) Control cards required only to save the VAMAT-generated data for execution of VAMSCN via a different job.

G.4.3.3 ATLAS Control Program

The following Control Program illustrates the statements required to execute the VAMAT and VAMSCN routines.

```

BEGIN CONTROL MATRIX PROGRAM LOAD
PROBLEM ID (VAMAT/VAMSCN EXECUTION)
DIMENSION FET(50,2),FET1(560),FET2(560),FET3(560),FET4(560)
(1) READ INPUT
(1) EXECUTE MASS
(1) PRINT OUTPUT(MASS,MDC=MDC****)
CALL FILEADD(FET,DATARNF,MASSRNF)
CALL FETADD(SAVESSF,FET1,560,1,0,IRR)
CALL FETADD(SAVESS1,FET2,560,1,0,IRR)
CALL FETADD(SC00SSF,FET3,560,1,0,IRR)
CALL FETADD(SC01SSF,FET4,560,1,0,IRR)
(2) CALL VAMAT
(3) CALL VAMSCN
END CONTROL PROGRAM

```

Notes: (1) Required only if VAMAT is to be executed.
(2) Required to execute VAMAT
(3) Required to execute VAMSCN

G.4.3.4 ATLAS Input Data

The input data required for execution of VAMAT and VAMSCN are summarized in this section. These data should be stacked immediately after the "END PROBLEM DATA" record of the ATLAS data deck. Detailed descriptions of the following data items are presented in reference G-6.

Record 1 Begin VAMAT Data Block

BEGIN VAMAT DATA

Record 2 FLEXSTAB Component Identification

Pntype Npa(I,1) Npa(I,2) Npa(I,3)

Pntype	=	Alphanumeric name (≤ 10 characters) of the FLEXSTAB component
Npa(I,1)	=	Integer number of panels per component. A zero must be input for this item if the component is on the input data-file but is not used.
Npa(I,2)	=	Integer number of freedoms per component. A zero must be input for this item if Npa(I,1) is zero.
Npa(I,3)	=	The integer 1, 2 or 3 corresponding to the x, y or z freedom number.

if Npa(I,1) is zero, the number of items to be skipped on the input data-file should be specified by Npa(I,3).

This record may be repeated to identify a maximum of 10 different components on the input data-file.

Record 3 Panel-Cut Data

Imax	ivam	nprint
------	------	--------

Imax = Number of panels (integer ≤ 300)
ivam = Number of cuts (integer ≤ 50)
nprint = Switch (integer) for printing the panel area-ratio and panel-cut data.
The options are:

1--Print the data
0--Do not print the data

Record 4 Section Data

Nd	sc	lbcut	nscut	Panel	Set
----	----	-------	-------	-------	-----

Nd = Number of sections (VAMAT components).
An integer ≤ 12

sc = Load factor

lbcut = Last body cut. The integer number of the last cut for which loads are multiplied by two and torsion is set to zero.

nscut = Integer number of cut of special surface (wing fin).

Panel = Integer number of the ATLAS Mass Auxiliary Panel Data subset of panels corresponding to the FLEXSTAB airload panels. If the panel geometry data are to be read from cards, see Record 6, a zero must be input for this iter

Set = The ATLAS Mass data-set number (integer) used in the generation of the panel weights. If the panel

geometry data are to be read from cards, see Record 6, a zero must be input for this item.

Record 5 Wing-Fin Panel Data (optional)

This record should not be input if "nscut = 0" in Record 4.

nps1	nps2	ncs1	ncs2
------	------	------	------

- nps1, nps2 = First and last panel numbers (integers) of special surface.

- ncs1, ncs2 = First and last cut numbers (integers) at which special-surface moments are to be added.

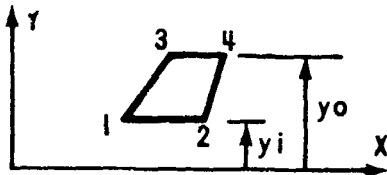
Record 6 Panel Geometry Data (optional)

If the panel geometry data are to be retrieved from the ATLAS data base (Panel#0 and Set#0 in Record 4), this record should not be used.

yi	yo	x(I,1)	x(I,2)	x(I,3)	x(I,4)
----	----	--------	--------	--------	--------

- yi, yo = Y-coordinates of the inboard and outboard panel edges.

- $\left. \begin{matrix} x(I,1), x(I,2) \\ x(I,3), x(I,4) \end{matrix} \right\}$ = X-coordinates of panel corners.



For "special surfaces" (wing-fins), the Y-coordinates are found by adding the Z-coordinates to the Y-coordinates.

This record is repeated for each panel ("lmax" in Record 3) in the same order that the panel airloads are contained in the input data file.

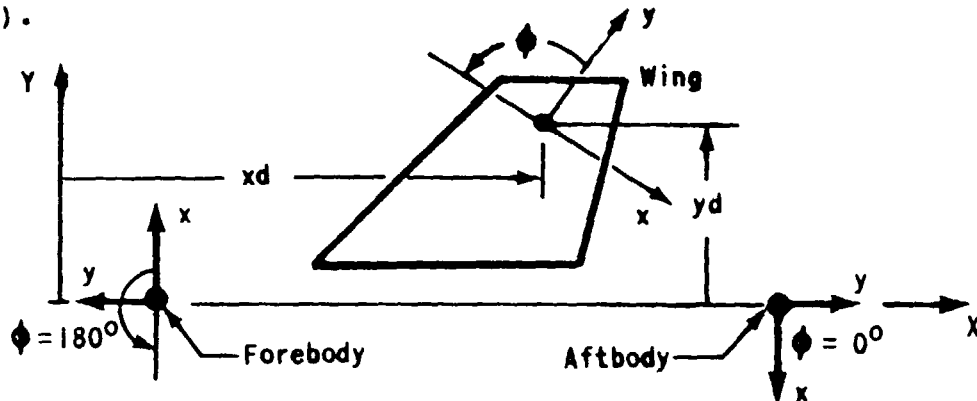
Record 7 Cut-Definition Data

xd	yd	s
----	----	---

- xd, yd = X-coordinate and Y-coordinate of the cut.

s = Orientation of the cut defined as " $-\text{TAN } \phi$ " and illustrated in the following sketch.

This record is repeated for each cut ("ivam" in Record 3).



For Wing: $0^\circ < \phi < 90^\circ$; If $\phi = 90^\circ$ (streamwise cut),
s = -999.

For Forebody: $\phi = 180^\circ$; s = .0001

For Aftbody: $\phi = 0^\circ$; s = 0.0

Record 8 Section Title Identification

***# name #**

name = Alphanumeric label (≤ 80 characters) for the section.

This record is repeated for each section ("Nd" VAMAT components in Record 4).

Record 9 Last Cut Numbers for Sections

nn(1) nn(2) ... nn(Nd)

nn(i) = The last cut number (integer) for the i-th section. If "k" components are skipped on the input data file, the last "k" items on this record must be input as zero.

Record 10 RT-value (panel area ratio) Modifications

mod(1) mod(2) ... mod(ivam)

`mod(i)` = Flag (integer) that identifies one of the following options:

- 0--RT-values are not modified.
- 1--RT-values for a continuous set of panels are to be modified.
- 2--Repeat previous RT-values for a continuous set of panels.
- 3--RT-values of individual panels are to be modified.

Modification data are input by Records 11 and 12.

Record 11 RT-value Modification Data (optional)

This record is input only if "`mod(i)=1`" or "`mod(i)=3`" in Record 10.

Variation 1: "`mod(i)=1`"

`ifirst ilast rtvalu`

`ifirst,ilast` = Integer numbers of first panel and last panel in a series to be modified.

`rtvalu` = RT-value to be given to the sequence of panels.

Variation 2: "`mod(i)=3`"

`nopan`

`nopan` = Number of panels to be modified (integer).

Record 12 RT-value Modification Data (optional)

This record is input only if "`mod(i)=3`" in Record 10.

`np(1) rt(1) np(2) ... np(nopan) rt(nopan)`

`np(i)` = Integer number of i-th panel to be modified

`rt(i)` = RT-value for the i-th panel.

Record 13 Panel Weight Matrix Name

`MDCxxxx`

MDCxxxx = Alphanumeric name (7 characters)
of the ATLAS panel weight matrix.

Record 14 End VAMAT Data Block

END VAMAT DATA

The following input data records are required for execution of VAMSCN. These data, if included in the data deck, should be stacked immediately after the VAMAT data.

Record 1 Begin VAMSCN Data Block

BEGIN VAMSCN DATA

Record 2 Condition and Case Data

Ncond ncyc nsum

Ncond = Integer number of conditions on the input data file.

ncyc = The number of groups of cases (integer ≤ 20) to be scanned (searched) on the first pass through the input data file.

nsum = The number of subsequent passes (integer ≤ 10) to be made through the data collected from the first pass.

Record 3 Subsequent Passes

This record is input only if "nsum" is not zero in Record 2.

ism(1) ism(2) ... ism(nsum)

ism(i) = The integer number of groups of sumcases to be scanned (searched) during the i-th subsequent pass through the data.

Record 4 Cases Scanned

nrec(1) nrec(2) ... nrec(ncyc)

nrec(i) = The integer number of cases to be scanned (searched) for maxima and

minima in the i-th group during the first pass through the input data file.

If a continuous sequence of cases is to be scanned and the "continuous-sequence" option of Record 6 is used, the number 2 shall be input for "nrec(i)." That is, only the first and last cases are counted, although the cases in between are also considered. If "nrec(i)" is input as a negative number, the resulting sumcase is not considered in subsequent passes.

Records 5 and 6 are input in pairs for each group of cases ("ncyc" in Record 2).

Record 5 Sumcase Title

***# title1 #**

title1 = Alphanumeric title (≤ 80 characters) for the sumcase of this group of cases--first pass through the input data.

Record 6 Cases to be Scanned

Mcase(1) Mcase(2) ... Mcase(nrec(i))

Mcase(i) = The i-th sequence number (integer) of a case to be included (scanned) in the group--first pass through the input data. The case sequence numbers within a group should be in the order in which these cases appear in the data file. The file is rewound after a group of cases has been read. Therefore, a particular case may be included in multiple groups.

"Continuous-Sequence Option"--If a continuous sequence of cases is to be scanned, e.g. cases 1,2,3,4,5, the user need only identify the first and last numbers of the sequence as negative numbers (-1 and -5 for this example). If this option is used, "nrec(i)" in Record 4 must be the integer 2.

If "nsum" is zero in Record 2, Records 7-9 should not be input. If "nsum≠0," Records 7, 8 and 9 are repeated "nsum" times.

Record 7 Sumcases for Subsequent Passes

nrec(1) nrec(2) ... nrec(ism(j))

nrec(i) = The integer number of sumcases to be scanned (searched) for maxima and minima in the i-th group during the j-th subsequent pass.

If a continuous sequence of cases is to be scanned and the "continuous-sequence" option of Record 9 is used, the number 2 shall be input for "nrec(i)." That is, only the first and last cases are counted, although the cases in between are also considered. If "nrec(i)" is input as a negative number, the resulting sumcase is not considered in subsequent passes.

Records 8 and 9 are repeated "ism(j)" times for the j-th pass.

Record 8 Sumcase Title

***# title2 #**

title2 = Alphanumeric title (≤ 80 characters) for the sumcase of this group of sumcases--the j-th pass.

Record 9 Sumcases to be Scanned

Mcase(1) Mcase(2) ... Mcase(nrec(i))

Mcase(i) = The i-th sequence number (integer) of a sumcase to be included (scanned) in the group--the j-th pass.

"Continuous-Sequence Option"--If a continuous sequence of sumcases is to be scanned, e.g. sumcases 1,2,3,4,5, the user need only identify the first and last numbers of the sequence as negative numbers (-1 and -5 for this example). If this option is used, "nrec(i)" in Record 7 must be the integer 2.

Record 10 End VAMSCN Data Block

END VAMSCN DATA

G.5 ATLAS/NASTRAN SYSTEM INTERFACES

The execution and input data required for the conversion of problem-data decks between ATLAS and NASTRAN Level 15 (ref. G-5) are described herein. The interfaces from ATLAS to NASTRAN and from NASTRAN to ATLAS are described in terms of the CDC-job-control card, ATLAS System Control-Program and input data-deck setup requirements. Limitations, approximations and details of the data conversions are also presented. Input data for one of the programs that are not directly relatable to input data for the other program are identified.

The ATLAS/NASTRAN System interfaces are described in the following sections.

G.5.1 ATLAS to NASTRAN Interface: Generate a tape on which ATLAS-generated structural and mass problem-data are written in a format that conforms to a NASTRAN bulk data deck.

G.5.2 NASTRAN to ATLAS Interface: Generate a tape on which NASTRAN bulk data are written in a format that conforms to an ATLAS data deck.

G.5.1 ATLAS to NASTRAN Interface

Conversion of ATLAS-generated structural and mass problem-data to compatible NASTRAN bulk-data card images is performed by the ATLAS to NASTRAN interface. The converted data are automatically written onto a file (tape) in an unsorted sequence. The user must add NASTRAN executive-control data and case-control data to the unsorted bulk data to produce a NASTRAN data deck.

ATLAS data are converted to NASTRAN bulk data by execution of the ATNA subroutine in the ATLAS-System library (ref. 1-1). This subroutine, which is called from an ATLAS Control Program, provides options for converting one or more of the following data categories:

- a) NODAL
- b) ELEMENT (Stiffness)
- c) BC and LOADS
- d) MASS

Matrices in the ATLAS data base are used by ATNA to effect the data conversion. These matrices are generated by previous execution of certain ATLAS Preprocessors and Processors. Generally, multiple bulk-data cards are generated for each ATLAS data record. This procedure of data conversion takes advantage

of the ATLAS input-data generation capabilities. The ATLAS program modules that must be executed prior to execution of ATNA are identified in section G.5.1.3.

G.5.1.1 Job Deck Structure

The ATLAS job deck required to convert ATLAS problem-definition data to NASTRAN bulk data is illustrated by figure 11-1 in section 11.0.

G.5.1.2 CDC 6600 Job Control Cards

The following list shows the control cards required to generate a NASTRAN bulk-data tape, NAST, that may be used for subsequent execution of NASTRAN.

```
Job Card
Account Card
GET (CATM/UN=UATLASU)
CALL (CATM,ATTACH)
CALL (CATM,KONTROL)
ATLAS.
REWIND (Savefile)
REQUEST (NAST,F=I, LB=KL,PO=AW) SAVE
COPYEI (Savefile,NAST)
RETURN (NAST)
```

Note: Savefile = One of the ATLAS Savefiles (SAVSSF or SAVESS1 through SAVESS4) on which the NASTRAN bulk-data card images are stored. See the description of subroutine ATNA in section G.5.1.3.

G.5.1.3 ATLAS Control Program

The following example Control Program illustrates the statements required to convert each of the categories of data from ATLAS to NASTRAN bulk-data card images.

```
BEGIN CONTROL MATRIX PROGRAM CONVERT
PROBLEM ID(CONVERSION OF ATLAS DATA TO NASTRAN DATA)
DIMENSION FET(50,5),AA(115),BB(115),CC(115),DD(115),EE(115)
C PREPROCESS THE ATLAS DATA TO BE CONVERTED.
READ INPUT
C EXECUTE THOSE ATLAS PROCESSORS, AS REQUIRED,
C FOR CONVERSION OF BC, LOADS AND MASS
C DATA CATEGORIES. THE STIFFNESS/MASS DATA
C SET NUMBER IS 3 AND THE BC STAGE NUMBER IS
C 2 FOR THIS EXAMPLE.
EXECUTE STIFFNESS (SET=3,RUN=STIFFNESS)
```

```

EXECUTE LOADS (SET=3, STAGE=2)
EXECUTE MERGE (LOADS, SET=3, STAGE=2, L11=11, L21=21, LOAD3=31)
EXECUTE MERGE (DISPLACEME, SET=3, STAGE=2, DISP=31)
EXECUTE MASS (SET=3, OPTION=3)
C OPEN FILES THAT ARE USED BY THE ROUTINE ATNA.
CALL FILEADD (FET, DATARNF, MERGRNF, MASSRNF, STIFRNF, SC0ORNF)
CALL FETADD (SC00SSF, AA, 115, 1, 0, IRR)
CALL FETADD (SC01SSF, BB, 115, 1, 0, IRR)
CALL FETADD (SC02SSF, CC, 115, 1, 0, IRR)
CALL FETADD (SC03SSF, DD, 115, 1, 0, IRR)
CALL FETADD (SAVESSF, EE, 115, 1, 0, IRR)
C IDENTIFY VALUES OF PARAMETERS FOR THE ATNA ROUTINE.
NSET = 3
NCOND = 0
IELEM = 1
INODE = 1
IMASS = 0
NFILE = SAVESSF
NSTAGE= 2
N1 = 3LL11
N2 = 3LL21
N3 = 5LLOAD3
N4 = 4LDISP
IBCLD = 1
C CONVERT ATLAS DATA TO NASTRAN BULK DATA AND
C STORE ON FILE SAVESSF.
CALL ATNA (NSET, NCOND, IELEM, INODE, IMASS, NFILE,
X NSTAGE, N1, N2, N3, N4, IBCLD)
C OTHER ATLAS STATEMENTS MAY BE USED HERE AS REQUIRED.
END CONTROL PROGRAM

```

The "READ INPUT" statement causes execution of the ATLAS Preprocessors for those data included in the ATLAS data deck. The required ATLAS input data sets and those ATLAS Processors which must be executed prior to execution of subroutine ATNA are identified in table G-2. All of the working files required by the routine ATNA must be opened as shown by the calls to FILEADD and FETADD. Only the output file of ATNA may be different from that shown in this example. This file, SAVESSF for this example, contains card images of the NASTRAN bulk data.

Table G-2. ATLAS Input Data and Processor Executions Required for Conversion of Data to NASTRAN Bulk Data

Data Category	ATLAS Input Data	Required ATLAS Processor Execution
NODAL	Nodal	
ELEMENT	Nodal; Stiffness	
BC, LOADS (No Loads or support displacement data)	Nodal; BC; Stiffness	STIFFNESS
BC, LOADS (BC, Loads and/or support displacement data)	Nodal; BC; Stiffness; Loads	STIFFNESS; LOADS; MERGE(LOADS); MERGE(DISPLACEMENT)
MASS	Nodal, BC, Mass	MASS

Subroutine ATNA is called with user-defined parameters as described below.

```
CALL ATNA (Se,Cond,Opt1,Opt2,Opt3,Afile,St,N1,N2,N3,N4,Opt4)
```

Se = The number (integer) of the ATLAS data SET to be processed.

Cond = The number (integer) of the ATLAS mass-distribution condition number to be processed. Input as zero if mass data are not to be processed.

Opt1,Opt2 } = Switches (integers) input as 0 or 1 to identify
Opt3,Opt4 } which data categories are to be processed.
"Opti" corresponds to one of the data categories as shown below.

Opt1 -- ELEMENT
Opt2 -- NODAL
Opt3 -- MASS
Opt4 -- BC,LOADS

Only those data that correspond to a category for which "Opti" is input as 1 are converted

to NASTRAN bulk-data card images.

- Afile = One of the ATLAS Savefiles (SAVESSF or SAVESS1 through SAVESS4). The NASTRAN bulk-data card images are stored on this file by execution of ATNA.
- St = Integer that identifies a boundary-condition STAGE associated with set "Se." Input as zero if only NODAL and/or ELEMENT data are to be processed.
- N1,N2,N3 = Alphanumeric names of the loads User Matrices previously generated for the specified SET and STAGE via execution of the MERGE Processor. These matrices are associated with the FREE, RETAIN and SUPPORT partitions, respectively of the gross stiffness matrix. Each name must be preceded by the letter "L" and the number of characters in the name. Input a zero for those matrices not available. If Loads data are not being converted, a zero should be input for each of these parameters.
- N4 = Alphanumeric name of the support-displacement User Matrix previously generated for the specified SET and STAGE via execution of the MERGE Processor. The name must be preceded by the letter "L" and the number of characters in the name. If support-displacements are not being converted, a zero should be input for this parameter.

G.5.1.4 Description of Data Conversion

The following limitations are imposed on the ATLAS to NASTRAN data conversion because of differences in the input data and technological capabilities of the two programs.

- a) All nodal coordinates are transformed, as necessary, to the GLOBAL rectangular reference frame. Therefore, local reference frames that may be defined via ATLAS Nodal data are not identified in the converted data.
- b) Material data records are not converted. All MAT cards for NASTRAN must be supplied by the user.

c) The following element types are not converted:

SCALAR
BRICK with more than 8 nodes
CPLATE
CCOVER

d) All elements with offsets are not converted.

e) Varying property values for an element are averaged prior to conversion.

f) Only the mass-matrix data generated by execution of the MASS Processor via OPTION=2 or OPTION=3 (ref. sec. 238.0) are converted.

The approximations and details of the ATLAS to NASTRAN data conversion are presented in table G-3. ATLAS input data records corresponding to each data category for conversion are identified via the same nomenclature used in the previous sections of this document. Optional data items are enclosed within < >. Two rows are shown for each NASTRAN bulk-data card image generated by the data conversion. The top row identifies the variables associated with the card image, whereas the second row identifies the corresponding data values generated by the conversion. A blank field indicates that a value is not generated for the corresponding variable. The nomenclature used for the card images is identical to that used in the NASTRAN User Manual (ref. G-5).

G.5.2 NASTRAN to ATLAS Interface

Conversion of NASTRAN structural and mass bulk-data card images to compatible ATLAS data records is performed by the NASTRAN to ATLAS interface. The converted data are automatically stored on a file (tape) in the form of an ATLAS data deck. Generally, the generated data deck must be modified selectively with hand-prepared data prior to execution by ATLAS. Additionally, the NASTRAN executive control, case-control data and plotting control data must be converted by hand.

Prior to conversion to ATLAS input-data records, the NASTRAN bulk data must be separated by the user into the following sequence of data categories:

- a) MATERIAL
- b) NODAL
- c) ELEMENT
- d) BC
- e) LOADS
- f) MASS

There are no requirements for the cards to be ordered within each of these categories except that the GRDSET card must be positioned before the GRID cards within the NODAL data, and if more than one card is required to define a certain type of data, the continuation cards must follow in order. The types of NASTRAN bulk-data cards that may be converted are summarized in table G-4. All other types of bulk-data cards are ignored during execution of the data conversion.

NASTRAN bulk data are converted to ATLAS input data records by execution of the NASTATL subroutine in the ATLAS_System library (ref. 1-1). This subroutine, which is called from an ATLAS Control Program, is described in section G.5.2.3.

G.5.2.1 Job Deck Structure

The ATLAS job deck required to convert NASTRAN bulk data to ATLAS problem-definition data is the same as illustrated by figure 11-1 in section 11.0 with the exception that ATLAS input data are not required.

G.5.2.2 CDC 6600 Job Control Cards

The following list shows the control cards required to generate an ATLAS data-deck tape, ATL, that may be used in subsequent execution of ATLAS.

```

Job Card
Account Card
GET (CATM/UN=UATLASU)
CALL (CATM,ATTACH)
CALL (CATM,CONTROL)
GET (Infile)
ATLAS (Infile)
REWIND (Savefile)
REQUEST (ATL,F=I,LB=KL,PO=AW) SAVE
COPYEI (Savefile,ATL)
RETURN (ATL)

```

Notes: Infile = Name of the file containing the NASTRAN data to be converted to ATLAS data.

Savefile = One of the ATLAS Savefiles (SAVESSF or SAVESS1 through SAVESS4) on which the ATLAS data-record images are stored. See the description of subroutine NASTATL section G.5.2.3.

G.5.2.3 ATLAS Control Program

The following example Control Program illustrates the statements required to convert NASTRAN data into an ATLAS data deck.

```

BEGIN CONTROL PROGRAM CONVERT
PROBLEM ID(CONVERSION OF NASTRAN DATA TO ATLAS DATA)
DIMENSION AA(115),BB(115),CC(115)
C OPEN FILES THAT ARE USED BY THE ROUTINE NASTATL.
CALL FETADD(SAVESS2,AA,115,1,0,IRR)
CALL FETADD(SC00SSF,BB,115,1,0,IRR)
CALL FETADD(SC01SSF,CC,115,1,0,IRR)
C CONVERT NASTRAN DATA ON FILE INFILE TO ATLAS
C INPUT DATA RECORDS AND STORE ON FILE SAVESS2.
CALL NASTATL(SAVESS2)
C OTHER ATLAS STATEMENTS MAY BE USED HERE AS REQUIRED.
END CONTROL PROGRAM

```

All of the working files required by the routine NASTATL must be opened as shown by the calls to FETADD. Only the output file of NASTATL (SAVESS2 in the example) may be different from that shown by the foregoing statements.

Subroutine NASTATL, which performs the data conversion, is called according to the following statement:

```
CALL NASTATL(Savefile)
```

Savefile = One of the ATLAS Savefiles (SAVESSF or SAVESS1 through SAVESS4). The ATLAS data-deck record images are stored on this file by execution of NASTATL. The appropriate BEGIN and END data records are inserted automatically to generate a complete ATLAS data deck.

G.5.2.4 Description of Data Conversion

The following limitations are imposed on the NASTRAN to ATLAS data conversion because of differences in the input data and technological capabilities of the two programs.

- a) Only the basic rectangular coordinate system is considered.
- b) If more than one card is required to define a certain type of NASTRAN data, the necessary continuation cards must follow in order.
- c) Only the NASTRAN bulk-data cards with 10 fields of 8 columns are converted.
- d) NSM (Non-Structural Masses) are ignored.
- e) All stress-recovery coefficients for NASTRAN structural elements are ignored.
- f) The maximum user element number is limited to 32678.
- g) A maximum of 49 different materials is allowed.
- h) The number of stages for BC (boundary-condition) data or SIDs for SPC and SPC1 cards is limited to 10.
- i) Only those concentrated masses defined by CONM2 cards are converted.

The approximations and details of the NASTRAN to ATLAS data conversion are presented in table G-4. NASTRAN bulk-data card images corresponding to each data category for conversion are identified by the same nomenclature used in the NASTRAN User Manual (ref G-5). The ATLAS data-record images generated by the conversion are also shown in table G-4.

Table G-3. Description of ATLAS to NASTRAN Data Conversion

Data Category	ATLAS Input Data	Remarks	Format and Contents of Generated NASTRAN Bulk Data Cards																			
			1	2	3	4	5	6	7	8	9	10										
MODAL (ref. sec. 14b.3)	M1, r, z, <A2>		1	GR1D	ID	CP	11	12	13	CD	P5											
			1	SR1D	ID	CP	11	12	13	CD	P5											
			2	GR1D	RL1	0	X	Y	Z	CD	P5											
ELEMENT (ref. sec. 157.0 and Appendix B)	REORDER FROM Nn		1	MP1C	31D	6	C	A	G	C	A											
			1	SE1P	ID	SE1D	ID	SE1P	10	11	12	13	14	15	16	17	18	19	20			
			1	CON1D	EN1D	EN1	EN2	M1D	A	J	C	BEM										
			1	CSAR	E1D	P1D	64	68	69	71	72	73	74	75	76	77	78	79	80			
			2	EN1C	PA	PA	Z1A	Z2A	Z3A	Z1B	Z2B	Z3B										
			3	PM1R	P1D	M1D	A	J	C	BEM												
			4	EN1C	E1	E2	E1	E2	E1	E2	E1	E2	E1	E2	E1	E2	E1	E2	E1	E2		
			5	EN1C	E1	E2	E1	E2	E1	E2	E1	E2	E1	E2	E1	E2	E1	E2	E1	E2		
			1	CON1D	E1D	EN1	EN2	M1D	A	J	C	BEM										
			2	CON1D	E1D	EN1	EN2	M1D	A	J	C	BEM										
			3	CON1D	E1D	EN1	EN2	M1D	A	J	C	BEM										
			4	CON1D	E1D	EN1	EN2	M1D	A	J	C	BEM										
			1	ET1RHEM	E1D	P1D	EN1	EN2	M1D	A	J	C	BEM									
			2	PT1RHEM	E1D	P1D	EN1	EN2	M1D	A	J	C	BEM									
			3	ET1RHEM	E1D	P1D	EN1	EN2	M1D	A	J	C	BEM									
			4	PT1RHEM	E1D	P1D	EN1	EN2	M1D	A	J	C	BEM									
			5	ET1RHEM	E1D	P1D	EN1	EN2	M1D	A	J	C	BEM									
			6	PT1RHEM	E1D	P1D	EN1	EN2	M1D	A	J	C	BEM									

Table continued on next page

Table G-4. Description of NASTRAN to ATLAS Data Conversion

Data Category	NASTRAN BULK DATA CARDS										Generated ATLAS Data Records	Remarks
	Card Type	Data Field Number										
	1	2	3	4	5	6	7	8	9	10		
MATERIAL	MATI	MID	E	G	MU	RHO	A	TREF	GE			
MODAL (a)	GRODSET	CP	CP	CP	CP	CP	CP	PS				
	GRID	ID	CP	I	X2	X3	CO	PS				
ELEMENT (b)	CC-ROD	E10	E10	G1	G2	M10	A	J	C	RSH		
	CROD	E10	P10	G1	G2	E10	P10	G1	G2			
	PROD	P10	M10	A	J	C	RSH					
	CHEXA (n=1, 2)	E10	E10	M10	G1	G2	G3	G4	G5	G6	ABC	
	*BC	G7	G8									
	CSHEAR PSHEAR	E10	P10	G1	G2	G3	G4					
	PSHEAR	P10	M10	T	RSH							
	CTUBE PTUBE	E10	P10	G1	G2							
	PTUBE	P10	M10	O0	T	RSH						
	BAROR	P10	P10	X1,60	X2	X3	X3	F				
	CBAR	E10	P10	GA	GB	X1,60	X2	X3	F	ABC		
	*BC	PA	PB									
	PBAR	P10	M10	A	I1	I2	J	RSH				
	*BC	C1	C2	D1	D2	E1	E2	F1	F2	DEF		
	*EF	K1	K2	I12								
	COMEN	E10	P10	G1	G2	G3	G4	TH				
	PODNEH	P10	M10	T	RSH							
	CTRMEN	E10	P10	G1	G2	G3	TH					
	PTRMEN	P10	M10	T	RSH							
	COUADZ	E10	P10	G1	G2	G3	G4	TH				
	POUADZ	P10	M10	T	RSH							
	CTR1AZ	E10	P10	G1	G2	G3	TH					
	PTR1AZ	P10	M10	T	RSH							
	COPLT	E10	P10	G1	G2	G3	G4	TH				
	POPLT	P10	M10	I	M102	T						
	CTRPLT	E10	P10	G1	G2	G3	TH					
	PTRPLT	P10	M10	I	M102	T						

Table continued on next page

Table G-4. Description of NASTRAN to ATLAS Data Conversion (Cont'd.)

Data Category	Card Type	NASTRAN BULK DATA CARDS										Generated ATLAS Data Records	Remarks
		Data Field Number											
		1	2	3	4	5	6	7	8	9	10		
LOADS (d)	CTRBSC	EID	PID	G1	G2	G3	G4	G5	G6	G7	G8	UPDATE Mode Icode Userid G1 G2 G3 G4 t TH /	Same as for TRPLT.
	PTRBSC	EID	PID	I1	I2	I3	I4	I5	I6	I7	I8		
	COUAD1	EID	PID	G1	G2	G3	G4	G5	G6	G7	G8		
	POUAD1	EID	PID	I1	I2	I3	I4	I5	I6	I7	I8		
	CTRIAI	EID	F10	F1	F2	F3	F4	F5	F6	F7	F8		
	PTRIAI	EID	F10	F1	F2	F3	F4	F5	F6	F7	F8		
	ASET	ID	C	D	E	F	G	H	I	J	K	RETAIN Freedom FOR ID /	
	ASETI	C	G1	G2	G3	G4	G5	G6	G7	G8	G9	RETAIN Freedom FOR G1 G2 G3 ... /	
												OR	
												RETAIN Freedom FOR ID1 TO ID2 /	
LOADS (d)	SPC	SID	G	C	D	E	F	G	H	I	J	STAGE SID /	The STAGE record is generated only if SID is different from that on the previous SPC or SPC1 card; The displacement record is generated only if D ≠ 0.
	SPC1	SID	C	B1	B2	B3	B4	B5	B6	B7	B8	STAGE SID /	
												OR	
												SUPPORT Freedom FOR G1 G2 ... GN /	
												OR	
												SUPPORT Freedom FOR G101 TO G102 /	
												CASE SID /	
												G F1 F2 F3 M1 M2 M3 /	
												CASE SID /	
												EID1 EID2 ... P /	
MASS	LOAD	SID	S	S1	L1	S2	L2	S3	L3	S4	L4	CASE SID COMBINE S5 S1 L1 S2 L2 ... S5N LN /	ID is the letter 'm' followed by the integer EID; m = 1-999.04; ij = 1-19, 206.04; The ATLAS associated mass subset number is assumed to be 010.
	CONM2	EID	G	G10	M	X1	X2	X3	X4	X5	X6	ID G M T1 T2 T3 T4 T5 T6 T7 /	
												(ref. sec. 130.0)	

a) The GROSSET card must be the first card in the MODAL data category; Only a GLOBAL rectangular coordinate system is assumed.

b) All NASTRAN NSM (Non-Structural Masses) are ignored; All stress-recovery coefficients for NASTRAN structural elements are ignored; The following ATLAS data items are generated: "Mcode" as described by the remarks for MAT1 cards; "Icode" is the letter 'm' followed by the integer TREF from the corresponding MAT1 card and "Userid" is the letter 'm' followed by the integer EID from the NASTRAN element card. Note that the user element numbers for the ATLAS data correspond to the EID entries specified by the NASTRAN data. Reference section 152.0 and Appendix B for descriptions of ATLAS element-definition data.

c) ASET and ASETI cards are not associated with a particular set of boundary conditions (a stage). Therefore, an ATLAS STAGE data-record is not generated for the corresponding RETAIN data records. Proper adjustments must be made by hand after the data conversion. "Freedom" is a list of kinematic labels selected from T1, T2, T3, T4, T5, T6, T7, R1, R2 for the NASTRAN "m" entries (e.g. T1, P1, R2 corresponds to a C-value of 26). Reference section 106.0 for descriptions of ATLAS BE data.

d) Reference section 134.0 for descriptions of ATLAS LOADS data.

APPENDIX H

ATLAS TEXT EDITOR

The ATLAS interactive text editor (sec. 200.5) can be used to edit files of line images. Each file may contain any number of logical records. Each logical record may contain any number of line images and each line may contain from 1 to 140 characters.

Although the capabilities of the ATLAS text editor are the same as the capabilities of the BCS text editor (CMEDIT) documented in reference 200-1, descriptions of the method of operation and the edit-request commands are reproduced herein for convenient referencing.

H.1 FILE SIZE

In order for a file to be edited, the entire file along with the editor program (slightly less than 15000B words) must fit in the job field length. Since line images may have different lengths, the number of lines that will fit is variable. Assuming a field length of 130000 octal words, approximately 37,000 words are available for the file.

H.2 METHOD OF OPERATION

The EDIT program uses a system of requests to locate a specific line and/or to perform a desired modification, insertion, or deletion. Specific lines in a file are referenced by means of a pointer. The pointer may be moved by execution of specific requests and is used by other requests. The pointer is not visible to the user, but he can easily determine its position in the file at any time by using the PRINT request.

H.3 KEYBOARD CONVENTIONS

Each EDIT request consists of the request name, followed by parameters if needed. The request name or its abbreviation, with prefix if desired (see sec. H.6), must be the first item on the line, and it must be separated from any parameters by at least one blank. Multiple parameters are also separated by blanks.

H.4 MODES OF OPERATION

The EDIT program operates in either edit mode or input mode. If an empty file is specified with the EDIT command, the input mode will be invoked automatically. If the file is not empty, the edit mode will be invoked. The mode of the editor is indicated by a prompt of E> in edit mode and I> in input mode. In the input mode, the editor accepts the entire input line as a new line image to be inserted in the file. To return from input to edit mode the user must respond to an I> prompt with a line consisting of only a carriage return, or two consecutive end-of-line characters (see LINEND request) at the end of the last line being entered. In the edit mode, each E> prompt expects a valid edit request to be entered. The input request allows the user to go back from edit to input mode.

H.5 ABBREVIATING REQUESTS

All requests may be abbreviated by simply deleting any number of trailing characters. However, certain requests require more than one character to uniquely identify them. They are BEFORE (BE), BLANK (BL), BRIEF (BR), DUP (DU), FILE (FIL), LINEND (LI), REMOVE (REM), REPEAT (REPE), START (ST), TABDEF (TA), and TABSET (TABS). In addition, if the INSERT request is used to insert a blank line, at least the characters "INS" are required to distinguish the request from the INPUT request; otherwise the single letter I is sufficient for INSERT.

H.6 PREFIXING REQUESTS

All requests may be prefixed with a "." (period) and/or a "-" (minus). If a "." is prefixed, the BRIEF-VERIFY mode is reversed for this request. For example, if VERIFY mode is in effect and an UP request is issued, the located line will be typed. However, if a .UP request were issued, no typeout would occur. The "." prefix has no effect in conditional BRIEF mode (BRIEF a b). If a "-" is prefixed, the pointer movement is reversed for this request. For example, a FIND request searches down the file, so a -FIND request will search up the file.

H.7 TAB CONTROL

The tab feature allows the user to shift input characters to preset locations in a line. The default tab settings are 1, 7, 11, 18, 30, 73. The first tab setting merely specifies the starting location of the first input character. The default tab character is the ">" ("\$" in batch). The tab character and the tab positions may be modified by the user with the TABDEF and TABSET requests, respectively.

H.8 MULTIPLE INPUT LINES AND EDIT REQUESTS

A number of logical input lines may be input as a single physical input line, by separating them with the logical end-of-line character (default is the ";"). Multiple requests may be entered in this manner to improve response time. The logical end-of-line character functions exactly like a CR (Carriage return).

H.9 CONTINUATION LINE

Since many terminals are limited to 72 or 80 characters per physical input line and the editor allows line images up to 140 characters, a continuation character (†) is available in either EDIT or INPUT mode to allow the user to continue his input line with the next line from the terminal. Typing the continuation character immediately before typing a CR will force concatenation of the next input line with the previous line.

H.10 BLOCK OPERATIONS

Blocks of text may be moved, duplicated, or deleted. The START and END requests define the block, and the MOVE, DUP, or REMOVE request performs the operation.

H.11 PREDEFINED REQUEST SEQUENCES

Each of two requests, X and Y, may be predefined to cause the execution of several EDIT requests. This predefined sequence may then be invoked any number of times by entering the X (or Y) request, thereby significantly reducing the time and cost of an edit session.

H.12 ENDS-OF-RECORD

A logical end-of-record mark can be written by inserting *WEOR as a line in the file. Multiple ends-of-record will produce empty records.

H.13 ENDS-OF-FILE

An end-of-file mark can be written by inserting *WEOF as a line in the file.

H.14 LENGTH OF TERMINAL LINE

The WIDTH request can be used to set the terminal line length for output from EDIT.

H.15 REMEMBERED PARAMETERS

The EDIT requests AFTER, BEFORE, BLANK, CHANGE, FIND, LOCATE, and OVERLAY each remember the parameters with which they were last used. The remembered parameters are used if the associated request is again given without any parameters.

H.16 PARAMETER CONVENTIONS

Several of the individual EDIT requests require a formal parameter N, where N is the number of lines subject to the operation of the request. If N is omitted, 1 is assumed. Otherwise the actual parameter must be an integer or an asterisk (*). In the latter case the effect of the request is to continue all the way to EOI (or BOI). Thus the request PRINT 7 will print 7 lines while PRINT * will print the rest of the file, in each case starting with the current line.

H.17 EDIT REQUESTS

Environment Selection:

<u>I</u> NPUT	-enter input mode
(type a (CR))	-return from input mode
<u>Q</u> UIT	-leave EDIT

Message Mode Selection:

<u>B</u> RIEF a b	-BRIEF mode
<u>V</u> ERIFY n	-VERIFY mode

Pointer Movement:

<u>B</u> OTTOM	-go to bottom of file
<u>F</u> IND line	-search for line
<u>L</u> OCATE :string1:string2:...	-search for a string
<u>N</u> EXT n	-advance n lines
<u>N</u> EXT :string1:string2:...	-advance to a string
<u>T</u> OP	-go to top of file
<u>U</u> P n	-go up n lines
<u>U</u> P :string1:string2:...	-go up to a string

Modification of Line Images:

<u>A</u> FTER :string1:string2: n m	-insert string2 after string1
<u>B</u> EFORE :string1:string2: n m	-insert string2 before string1
<u>B</u> LANK mask	-blank out characters
<u>C</u> HANGE :string1:string2: n m	-change string1 to string2
<u>D</u> ELETE n	-delete n lines
<u>D</u> ELETE :string1:string2:...	-delete until a string is found
<u>I</u> NSERT line	-insert "line" as line image
<u>O</u> VERLAY line	-overlay characters
<u>R</u> EPLACE line	-replace line image with "line"

File Handling:

<u>FILE</u> filename un pw	-save file and leave EDIT
<u>SAVE</u> filename un pw	-save file and return to input mode

Information Request:

<u>PRINT</u> n	-print n lines
<u>PRINT</u> :string1:string2:...	-print to a string

Special Character and Format Conventions:

<u>LINEND</u> character	-change end-of-line character
<u>TABDEF</u> character	-change tab character
<u>TABSET</u> n1 n2 ... nn	-change tab settings
<u>WIDTH</u> n	-specify line length
<u>ZONE</u> a b	-specify 1st and last columns

Block Operations:

<u>START</u>	-define start of block
<u>END</u>	-define end of block
<u>DUP</u>	-duplicate block
<u>MOVE</u>	-move block
<u>REMOVE</u>	-delete block

Miscellaneous:

<u>REPEAT</u> n	-repeat BLANK and OVERLAY requests
X /request1/request2/...	-save requests
X n	-execute saved requests n times
Y /request1/request2/...	-same as X
Y n	-same as X

<u>AFTER</u> [:string1:string2:[n[m]]]
--

This request inserts string2 after the mth occurrence of string1 in each of the next n lines, including the current line. If m has the value G, string2 will be inserted after every occurrence of string1 in the next n lines. If n is not specified, a value of 1 is assumed for n and it will not then be possible to specify any value for m except G. If m is not specified, 1 is assumed. If n=m=*, string2 will be inserted after every occurrence of string1 (from the current line to the end of the file). The pointer will advance n-1 lines whether or not any lines are changed.

If a zone is specified by the ZONE request, the AFTER request will treat the zone as if it were an entire line. Hence, nonblank characters will be deleted if they are shifted out of the zone without affecting characters outside the zone.

: is any delimiting character that does not appear in string1 or string2. The last delimiter may be omitted when no third or fourth parameters are specified and the last character of string2 is nonblank.

AFTER remembers the parameters from the last AFTER request.

Example: E>AFTER :B:X: 5
 READ(5,40) A,BX
 C=A*B
 WRITE(6,50) A,BX,C,BB
 E>

file before:
 READ(5,40) A,B <----- pointer
 C=A*B
 WRITE(6,50) A,B,C,BB
 50 FORMAT(1X,4E15.7)
 40 FORMAT(2E10.0)
 STOP
 END

file after:
 READ(5,40) A,BX
 C=A*B
 WRITE(6,50) A,BX,C,BB
 50 FORMAT(1X,4E15.7)
 40 FORMAT(2E10.0) <----- pointer
 STOP
 END

BEFORE[:string1:string2:[n[m]]]

This request inserts string2 before the mth occurrence of string1 in each of the next n lines, including the current line. If m has the value G, string2 will be inserted before every occurrence of string 1 in the next n lines. If n is not specified, a value of 1 is assumed for n and it will not then be possible to specify any value for m except G. If m is not specified, 1 is assumed. If n=m=*, string2 will be inserted before every occurrence of string1 (from the current line to end of file). The pointer will advance n-1 lines whether or not any lines are changed.

If a zone is specified by the ZONE request (see ZONE for description), the BEFORE request will treat the zone as if it were an entire line. Hence, nonblank characters will be deleted if they are shifted out of the zone without affecting characters outside the zone.

: is any delimiting character that does not appear in string1 or string2. The last delimiter may be omitted when no third or fourth parameters are specified and the last character of string2 is nonblank.

BEFORE remembers the parameters from the last BEFORE request.

Example: E>BEFORE :B:X: 5
 READ(5,40) A,XB
 C=A*XB
 WRITE(6,50) A,XB,C,BB
 E>

file before:
 READ(5,40) A,B <----- pointer
 C=A*B
 WRITE(6,50) A,B,C,BB
 50 FORMAT(1X,4E15.7)
 40 FORMAT(2E10.0)
 STOP
 END

file after:
 READ(5,40) A,XB
 C=A*XB
 WRITE(6,50) A,XB,C,BB
 50 FORMAT(1X,4E15.7)
 40 FORMAT(2E10.0) <----- pointer
 STOP
 END

BLANK[mask]

This request places blanks in the current line image wherever nonblank characters (other than the tab character) occur in mask. mask is separated from the BLANK request by one space. To BLANK the same character positions on a number of lines, the BLANK request may be preceded by the REPEAT request. See REPEAT for a further description.

BLANK remembers the parameters from the last BLANK request.

Example:

E>PRINT
 4030 FORMAT(2E10.0)
 E>BLANK XX
 40 FORMAT(2E10.0)
 E>

BOTTOM

This request repositions the pointer to the last line of the file.

Example:

```
E>BOTTOM
      END
E>
```

file:

```
      READ(5,40) A,B
      C=A*B
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0) <----- pointer before
50    FORMAT(1X.3E15.7)
      STOP
      END          <----- pointer after
```

BRIEF[a[b]]

There are two message modes in the EDIT environment: VERIFY (the default) and BRIEF. A request for either terminates the other. When any of certain EDIT requests (AFTER, BEFORE, BLANK, BOTTOM, CHANGE, FIND, LOCATE, NLXT, OVERLAY, and UP) is executed, the lines affected by the request are printed if in VERIFY mode; the affected lines are printed only from columns a to b if in BRIEF mode (printed from column a on if b is omitted); and the affected lines are not printed at all if in BRIEF mode with both a and b omitted.

Example:

```
E>BRIEF
E>CHANGE :E10:F7:
E>
```

```
E>BRIEF 15 20
E>CHANGE :E10:F7:
F7.0)
F>
```

line before:

```
40    FORMAT(2E10.0)
```

line after.

```
40    FORMAT(2F7.0)
```



```
CHANGE[ :string1;string2:[ n[ m]]]
```

This request changes the mth occurrence of string1 in each of the next n lines, including the current line, to string2. If m has the value G, every occurrence of string1 in the next n lines will be changed to string2. If n is not specified, a value of ; is assumed for n and it will not then be possible to specify any value for m except G. If m is not specified, 1 is assumed. If n=m=*, every occurrence of string1 (from the current line to the end of the file) will be changed to string2. The pointer will advance n-1 lines whether or not any lines are changed.

If a zone is specified by the ZONE request (see ZONE request), the CHANGE request will treat the zone as if it were an entire line. Hence, nonblank characters will be deleted if they are shifted out of the zone without affecting characters outside the zone. In addition, if characters are deleted from the zone, outside characters will not be shifted into it.

: is any delimiting character that does not appear in string1 or string2. The last delimiter may be omitted when no third or fourth parameters are specified and the last character of string2 is nonblank.

CHANGE remembers the parameters from the last CHANGE request.

Example 1:

```
E>CHANGE :B:X: 5
      READ(5,40) A,X
      C=A*X
      WRITE(6,50) A,X,C,BB
E>
```

file before:

```
      READ(5,40) A,B <----- pointer
      C=A*B
      WRITE(6,50) A,B,C,BB
50    FORMAT(1X,4E15.7)
40    FORMAT(2E10.0)
      STOP
      END
```

file after:

```
      READ(5,40) A,X
      C=A*X
      WRITE(6,50) A,X,C,BB
50    FORMAT(1X,4E15.7)
40    FORMAT(2E10.0) <----- pointer
      STOP
      END
```

C-9

Example 2:

```
E>CHANGE :E:F: 2 G
1000 FORMAT(2X,F10.0,5X,3(F10.2))
E>
```

file before:

```
PRINT 1000,X,(Y(I),I=1,3) <----- pointer
1000 FORMAT(2X,E10.0,5X,3(E10.2))
```

file after:

```
PRINT 1000,X,(Y(I),I=1,3)
1000 FORMAT(2X,F10.0,5X,3(F10.2)) <----- pointer
```

Example 3:

```
E>CHANGE :E:F: 2 2
1000 FORMAT(2X,E10.0,5X,3(F10.2))
E>
```

file before:

```
PRINT 1000,X,(Y(I),I=1,3) <----- pointer
1000 FORMAT(2X,E10.0,5X,3(E10.2))
```

file after:

```
PRINT 1000,X,(Y(I),I=1,3)
1000 FORMAT(2X,E10.0,5X,3(F10.2)) <----- pointer
```

DELETE[n] or DELETE :string1:string2:...

This request removes n lines starting with the current line. If n is not specified, a value of 1 is assumed. If n=*, all lines from the current pointer position to the end of the file are deleted. If strings are specified, all lines, starting with the current line and up to, but not including, the first line in which one of the strings is matched, will be deleted.

The pointer will move to the line just after the last line deleted.

: is any delimiting character that does not appear in any of the strings. The trailing delimiter may be omitted if the last string ends in a nonblank character.

Example 1:

```
E>DELETE 2
E>
```

file before:

```
WRITE(6,50) A,B,C
40 FORMAT(2E10.0) <----- pointer
50 FORMAT(1X,3E15.7)
STOP
END
```

```
file after:
  WRITE(6,50) A,B,C
  STOP          <----- pointer
  END
```

Example 2:

```
E>DELETE :END:STOP:
E>
```

```
file before:
  WRITE(6,50) A,B,C
40  FORMAT(2E10.0) <----- pointer
50  FORMAT(1X,3E15.7)
  STOP
  END
```

```
file after:
  WRITE(6,50) A,B,C
  STOP          <----- pointer
  END
```

FILE[filename[un[pw]]]

This request writes a copy of the edited file on disk, replacing the existing copy of the working file. EDIT is then terminated. When the filename is specified, EDIT will also replace the permanent file (indirect-access) of that name with a copy of the edited file. The name of the permanent file does not have to match the name of the working file. Both a user number (un) and a file password (pw) may be specified when replacing, but not creating, a permanent file. A password may not be specified unless a user number has been specified; similarly a user number requires a filename.

Example:

```
A>EDIT,NAME1
E> (various edit requests)
E>FILE NAME2
/* 015000 OCTAL REQUIRED TO EDIT
/*END
A>
```

The file NAME1 is edited and a permanent file NAME2 is saved, which is a copy of NAME1. If NAME1 was a permanent file as well as a working file before the above activity, the permanent file NAME1 will still exist in its original state.

FIND[line]

Starting with the next line image, the file will be searched for a sequence of characters corresponding to line. This request is position dependent, in that each character of line is compared to the corresponding character position in a line image. One blank separates the request from line.

Any blanks found in line will be ignored and the corresponding character position in a line image will not be compared.

FIND remembers the parameters from the last FIND request.

Example_1:

```
E>FIND 50
50  FORMAT(1X,3E15.7)
    READ(5,40) A,B
```

file:

```
    READ(5,40) A,B
    C=A*B           <----- pointer before
    WRITE(6,50) A,B,C
40  FORMAT(2E10.0)
50  FORMAT(1X,3E15.7) <----- pointer after
    STOP
    END
```

Example_2:

```
E>FIND > R           (R in column 10)
40  FORMAT(2E10.0)
E>
```

file:

```
    READ(5,40) A,B
    C=A*B           <----- pointer before
    WRITE(6,50) A,B,C
40  FORMAT(2E10.0)  <----- pointer after
50  FORMAT(1X,3E15.7)
    STOP
    END
```

INPUT

This request invokes the input mode of EDIT. In the input mode, the editor accepts the lines next entered as text, to be inserted after the current line. The pointer is advanced to each new line as it is entered. The special characters ">" (" \$" in batch) and ";" are interpreted as tab and logical end-of-line respectively, so if either of those characters is to be used as text in the input, the TABDEF and/or LINENC request must first be issued in the edit mode to define some other characters for tab and line-end. To return to the edit mode, type a line consisting only of a **CR** after the last line of data, or type the end-of-line character twice.

Example:

```
E>INPUT
I>C>CCMPUTE C
I>(type a CR)
E>
```

file before:

```
      READ(5,40) A,B <----- pointer
      C=A*B
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0)
50    FORMAT(1X,3E15.7)
      STOP
      END
```

file after:

```
      READ(5,40) A,B
C     COMPUTE C <----- pointer
      C=A*B
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0)
50    FORMAT(1X,3E15.7)
      STOP
      END
```

INSERT line or INSERT

This request inserts the specified line as a new line image after the current line, advancing the pointer to the new line. To insert a blank line, at least the characters INS must be typed; otherwise I is a sufficient abbreviation. One blank separates the request from "line".

Example:

```
E>INSERT C>COMPUTE C
E>
```

file before:

```
    READ(5,40) A,B <----- pointer
    C=A*B
    WRITE(6,50) A,B,C
40   FORMAT(2E10.0)
50   FORMAT(1X,3E15.7)
    STOP
    END
```

file after:

```
    READ(5,40) A,B
C    COMPUTE C      <----- pointer
    C=A*B
    WRITE(6,50) A,B,C
40   FORMAT(2E10.0)
50   FORMAT(1X,3E15.7)
    STOP
    END
```

LINEND[character]

This request redefines the logical end-of-line character to the character specified. If no character is specified, EDIT proceeds with no logical end-of-line character defined.

The LINEND request permits a number of logical input lines, separated by end-of-line characters, to be typed as a single physical input line. Multiple requests per line may be issued to EDIT in this manner.

The semicolon (;) is the default LINEND character.

Example:

```
E>TOP;LINEND /
E>FIND 50/CHANGE :H:X:
50   FORMAT(1H,3E15.7)
50   FORMAT(1X,3E15.7)
E>
```

file before:

```
    READ(5,40) A,B <----- pointer
    C=A*B
    WRITE(6,50) A,B,C
40   FORMAT(2E10.0)
50   FORMAT(1H,3E15.7)
    STOP
    END
```

file after:

```
    READ(5,40) A,B
    C=A*B
    WRITE(6,50) A,B,C
40   FORMAT(2E10.0)
50   FORMAT(1X,3E15.7)<--- pointer
    STOP
    END
```

LOCATE[:string1:string2:...]

Starting with the next line, all the characters in the specified zone (see ZONE request) of each line are scanned until one of the specified strings is found.

: is any delimiting character that does not appear in any of the strings. The trailing delimiter may be omitted if the last string ends in a nonblank character.

LOCATE remembers the parameters from the last LOCATE request.

Example:

```
E>LOCATE :50:
      WRITE(6,50) A,B,C
E>
```

file:

```
      READ(5,40) A,B      <----- pointer before
      C=A*B
      WRITE(6,50) A,B,C  <----- pointer after
40     FORMAT(2E10.0)
50     FORMAT(1X,3E15.7)
      STOP
      END
```

Note: If the user wishes to move the search pointer to the next line whose first three characters are XYZ, and he knows that the string XYZ occurs only at the beginning of a line, he can accomplish this by either of the following requests:

E>LOCATE :XYZ: or E>FIND XYZ

However, the LOCATE request would take between 70 and 140 times as long to execute as the FIND request (depending on the line length); this could mean, in a large file, that the response time might be, say, two seconds for FIND and four minutes for LOCATE. Since many needs to move the search pointer can be satisfied by FIND, the point is worth emphasis. If the situation demands LOCATE, but it is known that the desired string is a long way down the file, it is worthwhile to enter, say, NEXT 2000 to move down to an area much closer to the desired string.

NEXT[n] or NEXT :string1:string2:...

This request advances the pointer n lines. If n is not specified, a value of 1 is assumed. If n is greater than the number of lines remaining, the pointer is positioned on the last line. If n is negative, the pointer is positioned n lines before the current line.

: is any delimiting character that does not appear in any of the strings. The trailing delimiter may be omitted if the last string ends in a nonblank character.

If strings are specified, the NEXT request functions exactly like the LOCATE request.

NEXT does not remember the parameter(s) used on the previous NEXT request. Use LOCATE or FIND if multiple searches for the same string are desired.

Example 1:

```
E>NEXT 3
      STOP
E>
```

file:

```
      READ(5,40) A,B
      C=A*B
      WRITE(6,50) A,B,C <----- pointer before
40    FORMAT(2E10.0)
50    FORMAT(1X,3E15.7)
      STOP <----- pointer after
      END
```

Example 2:

```
E>NEXT -2
40    FORMAT(2E10.0)
E>
```

file:

```
      READ(5,40) A,B
      C=A*B
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0) <----- pointer after
50    FORMAT(1X,3E15.7)
      STOP <----- pointer before
      END
```


OVERLAY[line]

This request takes nonblank characters (other than the tab character) of "line" and places them in the corresponding position of the current line image. line is separated from the OVERLAY request by one space. To OVERLAY the same character positions on a number of lines, the REPEAT request may be used ahead of the OVERLAY request. See REPEAT for a further description.

OVERLAY remembers the parameters from the last OVERLAY request.

Example:

```
E>PRINT
150 ABCD=35.
E>OVERLAY >DIST
150 DIST=35.
E>
```

PRINT[n] or PRINT :string1:string2:...

This request prints n lines from the file, starting with the current line. The pointer is advanced to the last line printed. If n is omitted, 1 is assumed. If n=*, all lines from the pointer position to the end-of-file will be printed.

If strings are specified, all lines, starting with the current line to the first line in which one of the strings is matched, will be printed.

: is any delimiting character that does not appear in any of the strings. The trailing delimiter may be omitted if the last string ends in a nonblank character.

Example:

```
E>PRINT 3
      C=A*B
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0)
E>
```

file:

```
      READ(5,40) A,B
      C=A*B          <----- pointer before
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0) <----- pointer after
50    FORMAT(1X,3E15.7)
      STOP
      END
```

QUIT

With this request, EDIT returns control to the ATLAS interactive mode without altering the original file.

Example:

```
E>CHANGE :A:X: * *
      REXD(5,40) X,B
      C=X*B
      WRITE(6,50) X,B,C
40    FORMXT(2E10.0)
50    FORMXT(1X,3E15.7)
EOI:
E>QUIT
/* 015000 OCTAL REQUIRED TO EDIT
/*END
A>
```

file before:

```
      READ(5,40) A,B <----- pointer
      C=A*B
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0)
50    FORMAT(1X,3E15.7)
      STOP
      END
```

The above example shows an unwanted global change to the file. The QUIT request allows the user to preserve the original file.

REPEAT n

This request executes the next BLANK or OVERLAY request n times. If n is omitted, 1 is assumed. The pointer is advanced n lines after the BLANK or OVERLAY request.

Example:

Delete sequence numbers from columns 73-80 of a file.

```
A> EDIT,TEST
E>TABSET 1 73
E>BRIEF
E>REPEAT 10000
E>BLANK >XXXXXXXXX
EOI:
E>
```

If a line image contains trailing blanks, EDIT will delete the blanks, after executing the BLANK request. Hence a file can be condensed by using the above example, even if the file has no sequence numbers but does contain trailing blanks.

REPLACE line

This request replaces the current line image with "line". If no line is specified, input mode is invoked and the first line typed in input mode will replace the current line image. One blank separates the request from "line".

Example:

```
E>REPLACE >C=A/B**2
E>
```

file before:

```
      READ(5,40) A,B
      C=A*B          <----- pointer
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0)
50    FORMAT(1X,3E15.7)
      STOP
      END
```

file after:

```
      READ(5,40) A,B
      C=A/B**2      <----- pointer
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0)
50    FORMAT(1X,3E15.7)
      STOP
      END
```

SAVE[filename[un[pw]]]

This request writes a copy of the edited file on disk, replacing the existing copy of the working file, and invokes the input mode of EDIT. When the filename is specified, EDIT will also replace the permanent file (indirect-access) of that name with a copy of the edited file. The name of the permanent file need not match the name of the working file. Both a user number (un) and a password (pw) may be specified when replacing, but not creating, a permanent file. A password may not be specified unless a user number has been specified; similarly, a user number requires a filename. The pointer is not moved by SAVE.

Example:

```
E>EDIT,NAME1
E>(various edit requests)
E>SAVE NAME2 ABCD99 PASS
I>
```

The file NAME1 is edited, and a copy is saved as permanent file NAME2 with user number ABCD99 and file password PASS. If NAME1 was a permanent file before the above activity, the permanent file NAME1 will still exist in its original state.

START
END
MOVE
DUP
REMOVE

These requests provide a means to move, duplicate, or delete blocks of line images in a file. Before a MOVE, DUP, or REMOVE request is issued, the first and last lines of the block must be defined. This is done by issuing the START and END requests when the pointer is pointing to the appropriate line images. The END of the block cannot occur before the START. The pointer must then be moved to the line after which the block is to be inserted before the MOVE or DUP request is issued. The line after which the block is to be inserted must not be located within the block unless it is the last line of the block. The block starting and ending locations will be reset to the new block.

The requests outlined above must be given in the order: START, END, MOVE (or DUP). However, additional EDIT requests may be given between block requests. The block starting and ending locations will be adjusted if modifications are made to the file.

An additional combination of requests is available for removing blocks of data. The first line of the block of data to be removed must be located and the START request issued. The pointer should then be moved to the last line of data to be removed and the REMOVE request issued.

If the block definition does not exist, the diagnostic "INVALID EDIT REQUEST:" will be displayed whenever the MOVE, DUP, or REMOVE request is issued. The diagnostic will also be displayed if an attempt is made to include the first line (the null line) of the file in the block definition, to place the end of the block in front of the START, or to duplicate or move a block into itself.

Example:

```
E>PRINT
      READ(5,40) A,B
E>START
E>NEXT 2
      WRITE(6,50) A,B,C
E>END
E>NEXT 2
50    FORMAT(1X,3E15,7)
E>DUF
E>
```

file before:

```

      READ(5,40) A,B    <----- pointer
      C=A*B
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0)
50    FORMAT(1X,3E15.7)
      STOP
      END

```

file after:

```

      READ(5,40) A,B
      C=A*B
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0)
50    FORMAT(1X,3E15.7)
      READ(5,40) A,B
      C=A*B
      WRITE(6,50) A,B,C <----- pointer
      STOP
      END

```

TABDEF[character]

This request redefines the logical tab character to the character specified. If no character is specified, the logical tab character reverts to the default character ">".

Example:

```

E>..BDEF $
E>NEXT
      C=A*B
E>REPLACE $C=A*B
E>PRINT
      C=A*B
E>

```

TABSET[n1 n2 ... nn]

n1 is the column in the line at which the line is to begin. n2 ... nn are column positions for logical tab settings. If parameters are omitted, default tab settings (1,7,11,18,30,73) will be used. The maximum number of tab settings is 12.

The tab character (">" is the default) has the desired effect when used with the INSERT command, but not when used with the CHANGE command. That is, on the INSERT command, any appearance of ">" causes the next character to be stored in the next tab position defined by TABSET, and the ">" is of course not stored. But on the CHANGE command the character ">" is not given any special significance (see second example).

Example 1:

```
E>TABSET 5 11 21
E>INSERT 5.>11.>21.
E>PRINT
    5.    11.    21.
E>
```

Example 2:

```
E>TABSET 5 11 21
E>INSERT 5.>11.21.
E>PRINT
    5.    11.21.
E>CHANGE :11.:11.>:
    5.    11.>21.
E>
```

TCP

This request repositions the pointer to the top of the file (the null line in front of the user's first line).

Example:

```
E>TOP
E>
```

file:

```
                                <----- pointer after
    READ(5,40) A,B
    C=A*B
    WRITE(6,50) A,B,C
40  FORMAT(2E10.0) <----- pointer before
50  FORMAT(1X,3E15.7)
    STOP
    END
```

UP[n] or UP :string1:string2:...

This request repositions the pointer n lines before the current line. If n is not specified, a value of 1 is assumed. If n is greater than the number of lines to the top of the file, the pointer is positioned at the top. UP may also be used to search upwards for one of the specified strings.

: is any delimiting character that does not appear in any of the strings. The trailing delimiter may be omitted if the last string ends in a nonblank character.

UP does not remember the parameter(s) used on the previous UP request. Use LOCATE or FIND if multiple upward searches for the same string are desired.

Example:

```
E>UP 4
      C=A*B
E>
```

file:

```
      READ(5,40) A,B
      C=A*B           <----- pointer after
      WRITE(6,50) A,B,C
40     FORMAT(2E10.0)
50     FORMAT(1X,3E15.7)
      STOP           <----- pointer before
      END
```

VERIFY[n]

This request terminates the BRIEF mode in the edit environment (see BRIEF request). The parameter n is the number of columns to verify in each line. The default is all columns. The EDIT requests that are controlled by VERIFY are AFTER, BEFORE, BLANK, BOTTOM, CHANGE, FIND, LOCATE, NEXT, OVERLAY, and UP.

Example:

```
E>BRIEF
E>NEXT
E>VERIFY
E>NEXT
40     FORMAT(2E10.0)
E>
```

file:

```
      READ(5,40) A,B
      C=A*B           <----- pointer before
      WRITE(6,50) A,B,C
40     FORMAT(2E10.0) <----- pointer after
50     FORMAT(1X,3E15.7)
      STOP
      END
```

WIDTH[n]

This request sets the upper limit n for the number of characters printed by EDIT on a terminal line. If no value for n is specified, the line width reverts to the default value 70

Example:

```
E>PRINT
40   FORMAT(2E10.0)
E>WIDTH 10
E>PRINT
40   FORM
AT(2E10.0)
E>
```

```
X /request1/request2/...
Y /request1/request2/...
X[ n ]
Y[ n ]
```

The form X /request1/request2/... saves a set of EDIT requests for later execution and temporarily names it X (similarly for Y). Two sets of requests may be saved concurrently.

The form X n (or Y n) allows the set of requests named X (or Y) to be executed n times. The default value for n is 1.

/ is any delimiting character that does not appear in any of the requests. The trailing delimiter may be omitted if the last request ends in a nonblank character.

Example:

```
E>X /LOCATE :STOP:/DELETE/
E>X
      STOP
E>
```

file before:

```
      READ(5,40) A,B
      C=A*B           <----- pointer before
      WRITE(6,50) A,B,C
40   FORMAT(2E10.0)
50   FORMAT(1X,3E15.7)
      STOP
      END
```

file after:

```
      READ(5,40) A,B
      C=A*B
      WRITE(6,50) A,B,C
40   FORMAT(2E10.0)
50   FORMAT(1X,3E15.7)
      END           <----- pointer after
```


ZONE[a[b]]

Parameters a and b are the first and last columns of a line to be scanned with the LOCATE, NEXT, or UP request. For the AFTER, BEFORE, or CHANGE request, a and b define the limits of their effects. The default is all columns (ZONE 1 140).

Example:

```
E>ZONE 10 15
E>LOCATE :A:
4C   FORMAT(2E10.0)
E>
```

file:

```
      READ(5,40) A,B <----- pointer before
      C=A*B
      WRITE(6,50) A,B,C
40    FORMAT(2E10.0) <----- pointer after
50    FORMAT(1X,3E15.7)
      STOP
      END
```

REFERENCES

REFERENCES

- 1-1 W. J. Erickson, ed., "ATLAS--An Integrated Structural Analysis and Design System, System Design Document," NASA CR-159042, July 1979.
- 1-2 F. P. Gray, ed., "ATLAS--An Integrated Structural Analysis and Design System. Random Access File Catalog," NASA CR-159044, July 1979.
- 1-3 W. J. Erickson, "SNARK User's Manual," BCS-G0686, Boeing Computer Services, June 1975.
- 104-1 T. Theodorsen, "General Theory of Aerodynamic Instability and the Mechanism of Flutter," NACA TR-496, 1934.
- 104-2 L. D. Richmond, "A Rational Method of Obtaining Three-Dimensional Unsteady Aerodynamic Derivatives of Intersecting Airfoils in Subsonic Flow," D6-7401, Boeing Commercial Airplane Company, April 1962.
- 116-1 J. P. Giesing, T. P. Kalman and W. P. Rodden, "Subsonic Unsteady Aerodynamics for General Configurations," Part I, Vols. I and II, AFFDL-TR-71-5, Feb. 1971.
- 116-2 J. M. Yi and D. F. Tankersley, "Unsteady Aerodynamic Interaction for Advanced Aircraft Configurations in Subsonic Flow (Improved Doublet-Lattice TEV 155)," D6-41258, Boeing Commercial Airplane Company, Dec. 1973.
- 122-1 R. N. Desmarais and R. M. Bennett, "An Automated Procedure for Computing Flutter Eigenvalues," AIAA Paper No. 73-393, presented at the AIAA/ASME/SAE 14th Structures, Structural Dynamics and Materials Conference, Williamsburg, Virginia, March 20-22, 1973.
- 136-1 J. M. Yi, C. J. Borland and J. R. Hogley, "Prediction of Unsteady Aerodynamic Loadings of Non-Planar Wings and Wing-Tail Configurations in Supersonic Flow," Part I--Theoretical Development, Program Usage and Application, AFFDL-TR-71-108, Wright-Patterson Air Force Base, Ohio, March 1972.
- 140-1 "Structural Design Guide for Advanced Composite Applications," Vol. II, Second edition, prepared under Contract F33615-69-C-1368 by North American Rockwell Corporation, Los Angeles Div. for Advanced Composites Divs. Air Force Materials Laboratory, Wright-Patterson

Air Force Base, Ohio, Air Force Systems Command, Jan. 1971.

- 140-2 T. W. Eichenberger, Coordination Sheet defining the ATLAS Standard-Material Properties, Coordination Sheet No. SMA-01-298-R2, The Boeing Company, Aug. 24, 1972.
- 150-1 W. S. Rowe, R. A. Winther and M. C. Redman, "Prediction of Unsteady Aerodynamic Loadings Caused by Trailing Edge Control Surface Motions in Subsonic Compressible Flow," NASA CR-2003, June 1972.
- 200-1 "MAINSTREAM-EKS KRONOS Interactive Timesharing (KIT), User's Manual," BCS-10208-005, Boeing Computer Services, Inc., July 1975.
- 208-1 R. S. Martin and J. H. Wilkinson, "Reduction of the Symmetric Eigenproblem $Ax = \lambda Bx$ and Related Problems to Standard Form," Numerische Mathematik, No. 11, pp. 99-110, 1968.
- 208-2 J. H. Wilkinson, "The Algebraic Eigenvalue Problem," Clarendon Press, Oxford, 1965.
- 210-1 A. Ralston, "A First Course in Numerical Analysis," McGraw-Hill Book Co., 1965.
- 210-2 A. S. Householder, "The Theory of Matrices in Numerical Analysis," Blaisdell Publishing Co., 1965.
- 212-1 B. F. Backman, "ATLAS--An Integrated Structural Analysis and Design System, Design Module Theory Document," NASA CR-159046, July 1979.
- 212-2 Proceedings of the International Symposium on Computer-Aided Structural Design, July 10-14, 1972.
- 212-3 AGARD Conference Proceedings No. 36, Symposium on Structural Optimization, Tarabya, Istanbul, Turkey, Oct. 8-10, 1969.
- 212-4 R. Hill, "The Mathematical Theory of Plasticity," Oxford University Press, 1960.
- 212-5 J. H. Johnson, Jr., "Critical Buckling Stresses of Simply Supported Rectangular Plates Under Combined Longitudinal Compression, Transverse Compression and Shear," Journal of the Aeronautical Sciences, p. 41, June 1954.

- 212-6 H. M. Adelman, J. L. Walsh and R. Narayanaswami, "An Improved Method for Optimum Design of Mechanically and Thermally Loaded Structures," NASA TN-D-7965, Aug. 1975.
- 220-1 M. J. Turner et al., "Investigation of Residual Flexibility Effects in Flutter Analysis," D6-40881, Boeing Commercial Airplane Company, Dec. 1972.
- 232-1 R. L. Harder and R. N. Desmarais, "Interpolation Using Surface Splines," Journal of Aircraft, Vol. 9, No. 2, pp. 189-191, Feb. 1972.
- 234-1 R. A. Samuel, "ATLAS--An Integrated Structural Analysis and Design System, Loads Module Theory Document," D6-25400-0101, Boeing Commercial Airplane Company, Oct. 1975.
- 238-1 R. J. Guyan, "Reduction of Stiffness and Mass Matrices," Journal of the American Institute of Aeronautics and Astronautics, Vol. 3, No. 2, p. 380, 1965.
- 252-1 J. S. Przemieniecki, "Theory of Matrix Structural Analysis," McGraw-Hill Book, Co., 1968.
- 258-1 R. S. Martin and J. H. Wilkinson, "Reduction of the Symmetric Eigenproblem $Ax = \lambda Bx$ and Related Problems to Standard Form," Numerische Mathematik, No. 11, pp. 99-110, 1968.
- 258-2 J. H. Wilkinson, "The Algebraic Eigenvalue Problem," Clarendon Press, Oxford, 1965.
- F-1 R. A. Samuel, "ATLAS--An Integrated Structural Analysis and Design System, System Demonstration Problems," NASA CR-159045, July 1979.
- G-1 C. B. Craidon, "Description of a Digital Computer Program for Airplane Configuration Plots," NASA TM X-2074, Sept. 1970.
- G-2 W. D. Middleton, J. L. Lundry and R. G. Coleman, "Aerodynamic Design and Analysis System for Supersonic Aircraft," NASA CR-2521, March 1975.
- G-3 L. Gotkin, "User's Guide for the Wing and Body Geometry Control System," D6-4967, Boeing Commercial Airplane Company, Jan. 1979.

- G-4 G. R. Hink et al., "A Method for Predicting the Stability Characteristics of an Elastic Airplane," Vol. III--FLEXSTAB Program Manual, NASA CR-114714, Ames Research Center, Oct. 1974.
- G-5 C. W. McCormick, "The NASTRAN User's Manual," NASA SP-220(01), June 1972.
- G-6 R. M. Thomas and F. P. Gray, "Peripheral Loads Programs for the ATLAS-FLEXSTAB System," D6-42315TN, Boeing Commercial Airplane Company, Feb. 1977.