AFWAL-TR-87-3069
VOLUME V

EXPERIMENTAL MODAL ANALYSIS AND DYNAMIC COMPONENT SYNTHESIS

VOL V - Universal File Formats

Dr. Randall J. Allemang, Dr. David L. Brown
Structural Dynamics Research Laboratory
Department of Mechanical and Industrial Engineering
University of Cincinnati
Cincinnati, Ohio 45221-0072

December 1987

Final Technical Report for Period November 1983 - January 1987

Approved for public release; distribution is unlimited

FLIGHT DYNAMICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6553

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely Government-related procurement, the United States Government incurs no responsibility or any obligation whatsoever. The fact that the Government may have formulated or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication, or any other person or corporation; or as conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Office of Public Affairs (ASD/PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public , including foreign nations.

This technical report has been reviewed and is approved for publication.

OTTO F. MAURER, Principal Engineer
Structural Dynamics Branch
Structures Division

JEROME PEARSON, Chief
Structural Dynamics Branch
Structures Division

FOR THE COMMANDER

ROBERT M. BADER, Ass't Chief
Structures Division
Flight Dynamics Laboratory

If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFWAL/FIBG, Wright-Patterson AFB, OH 45433-6553 to help us maintain a current mailing list.

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S): AFWAL-TR-87-3069 VOL.V
6a. NAME OF PERFORMING ORGANIZATION: UNIVERSITY OF CINCINNATI; 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: AIR FORCE WRIGHT AERONAUTICAL LABORATORIES, FLIGHT DYNAMICS LABORATORY
6c. ADDRESS (City, State and ZIP Code): CINCINNATI OH 45221-0072	7b. ADDRESS (City, State and ZIP Code): AFWAL/FIBG Wright-Patterson AFB OH 45433-6553
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: Armament Test Laboratory; 8b. OFFICE SYMBOL (If applicable): AFATL	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: F33615-83-C-3218
8c. ADDRESS (City, State and ZIP Code): Eglin AFB FL 32542	10. SOURCE OF FUNDING NOS.

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.
62201F	2401	04	16

11. TITLE (Include Security Classification) EXPERIMENTAL MODAL ANALYSIS AND DYNAMIC COMPONENT SYNTHESIS (UNCLASSIFIED) Vol. V - Universal File Formats

12. PERSONAL AUTHOR(S) DR. RANDALL J. ALLEMANG DR. DAVID L. BROWN

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
FINAL	FROM NOV 1983 TO JAN 1987	DECEMBER 1987	47

16. SUPPLEMENTARY NOTATION
The computer software contained herein are theoretical and/or references that in no way reflect Air Force-owned or developed computer software.

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB. GR.	
22	01		MODAL TESTING VIBRATION TESTING DYNAMICS

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The concept of an ASCII, 80 character per record, data base format, referred to as a Universal File, is presented to serve as the basis for data exchange between different hardware and software environments. Since the data involved is concerned with the structural dynamics testing and analysis areas, the Universal File formats include initialization files, geometry files, measurement files, and analysis data files. The Universal File format that is utilized originated with the Structural Dynamics Research Corporation (SDRC).

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL: OTTO F. MAURER	22b. TELEPHONE NUMBER (Include Area Code): (513)255-5236; 22c. OFFICE SYMBOL: AFWAL/FIBG

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE.

SECURITY CLASSIFICATION OF THIS PAGE

SUMMARY

The concept of an ASCII, 80 character per record, data base format, referred to as a Universal File, is presented to serve as the basis for data exchange between different hardware and software environments. Since the data involved is concerned with the structural dynamics testing and analysis areas, the Universal File formats include initialization files, geometry files, measurement files, and analysis data files. The Universal File format that is utilized originated with the Structural Dynamics Research Corporation (SDRC).

PREFACE

This report is one of six Technical Reports that represent the final report on the work involved with United States Air Force Contract F33615-83-C-3218, Experimental Modal Analysis and Dynamic Component Synthesis. The reports that are part of the documented work include the following:

AFWAL-TR-87-3069

VOLUME I	Summary of Technical Work
VOLUME II	Measurement Techniques for Experimental Modal Analysis
VOLUME III	Modal Parameter Estimation
VOLUME IV	System Modeling Techniques
VOLUME V	Universal File Formats
VOLUME VI	Software User's Guide

For a complete understanding of the research conducted under this contract, all of the Technical Reports should be referenced.

ACKNOWLEDGEMENTS

The University of Cincinnati Structural Dynamics Research Laboratory (UC-SDRL) would like to acknowledge the following groups and persons who have contributed to the writing of this report:

- SDRC, particularly John Crowley, Steve Crowley and Tom Rocklin for assistance with the Universal File structure.

- Otto Maurer, Contract Monitor, for guidance, advice, and assistance during the course of this contract.

The following members of the staff of the University of Cincinnati Structural Dynamics Research Laboratory have contributed to the writing of this report as follows:

- Max L. Wei, for the development and evaluation of the software implementing the Universal File structure.
- Allyn W. Phillips, for his assistance in preparing the final form of this report.

TABLE OF CONTENTS

1. OVERVIEW

1.1 Introduction

One of the significant problems of experimental and analytical structural analysis involves combining, comparing, and correlating data that exists in different formats, in different software and in different hardware. This problem is not a technological problem so much as it is a logistical problem. In order to address this problem, a standardized data base structure needs to be identified and supported by all of the organizations operating in the structural dynamics area. While this goal cannot be accomplished, ultimately, until an official standard exists, it is possible to alleviate the problem by identifying the basis for a data base structure and providing this information to the organizations that would eventually be involved in the development of an official standard. The objective of this report is to document a data base standard that can provide a means for data exchange.

The requirements for a data base standard that can be applicable to different software and hardware environments must be very general so that any level of user can support the data base standard. For this reason, an eighty character per record, ASCII format is the only basis for the data base structure that can be supported in the required environments. It is important to note that this data base format is not intended to be used as an internal format within software or as the basis of a hardware format. This sort of format is only useful as a mechanism for input and output to media that is compatible with the different environments that may need to be utilized.

1.2 Format Development

In order to develop the data base structure, the types of formats or capabilities that were needed were first identified. The basic requirements included a file structure that could define the geometry of the nodal degrees of freedom, measurement data at the nodal degrees of freedom, and modal parameters associated with the nodal degrees of freedom. In addition to these basic requirements, information concerned with the source of the file information and the units of the data is needed to qualify the information in the files that belong to a specific data base.

Once the basic requirements were identified, existing data base structures were evaluated to determine whether a current format would be sufficient or could be modified to meet the basic requirements. In this regard, consideration was given to the basic requirement that the format be ASCII, to whether the data base already included the required formats, to whether the data base is being utilized at the present time, etc. Several possibilities existed with respect to an internal data base developed at the University, to data bases utilized by finite element programs, and to data bases utilized by experimentally based programs. For example, the University of Cincinnati Structural Dynamics Laboratory (UC-SDRL) had developed an ASCII format data base in order to compare finite element and experimental test data. This format was limited to nodal geometry and modal parameters and would have to be expanded in order to service all of the needs that exist in the analytical and experimental structural dynamics area.

As a result of this review and deliberation, the Universal File[1,2] concept utilized by Structural Dynamics Research Corporation (SDRC) was adopted as the basis for the data base structure. In general, this Universal File concept addressed the needs of both the analytical and experimental aspects of the structural dynamics area. Also, there is considerable experience and history of the use of this Universal File structure in both the analytical and experimental programs that SDRC has developed. The structure of the Universal File is documented very well and has already been adopted by other organizations as the basis for internal data base structures. Additionally, SDRC supported the concept of a wider application of the Universal File concept and has added Universal File structures to address potential needs that previously had not been identified. For example, the Units File (File Type 156) has been added to facilitate the different units that occur when data originates from different hardware and software vendors.

1.3 Universal File Concept

A Universal File is a physical file, card deck, magnetic tape, paper tape, etc. containing symbolic data in physical records with a maximum record length of 80 characters.

On the physical file, data is contained in logical data sets with the following characteristics:

a. The first record of the data set contains "-1" right justified in columns 1 through 6. Columns 7 through 80 of the physical record are blanks.

b. The second record of the data set contains the data type number, numeric range 1 through 32767, right justified in columns 1 through 6. Columns 7 through 80 of this physical record are blanks.

c. The last record of the data set contains "-1" right justified in columns 1 through 6. Columns 7 through 80 of the physical record are blanks.

d. The specification of data on the remaining records of the data set are totally dependent on the data set type.

For example:

```
  -1
 xxx
  .
  .
(data pertaining to the data set type)
  .
  .
  -1
```

Although the data organization is built around 80 character records, the capacity for data record blocking has been provided. Its principle use would be for magnetic tapes where the overhead associated with 80 character records is excessive. As such, a preferred physical/logical record blocking of 80 logical records per physical record is recommended. This improves system capacity and response dramatically.

1.4 Future Considerations

If further data base structures become necessary, several options can be pursued. First of all, the Universal Files documented in later sections of this report are a subset of the Universal Files supported by SDRC. Other Universal File formats may already exist which satisfy future requirements.[1,2] If another Universal File format does not already exist to service the intended needs, a new format can be developed as long as the Universal File format number is unique.

Another future consideration is the development of other similar formats. A similar concept to Universal Files is being developed in Europe, called Neutral Files and Meta Files,[3-6] to serve the same purpose. If future standards are developed and adopted, conversion programs to convert from the Universal File format to the new formats should be straight forward.

2. INITIALIZATION FILES

2.1 Introduction

This section describes the formats of the files needed to initialize the file system, specifically, the Header File (151), the Units File (156), and the Component Header Data File (241). Each of these files is optional and, if not present, default values are used for any required information.

The Header File contains the information for the model's name, description, and the generating program. It also contains information for the time and date of the file's creation and last access.

The Units File contains the units and the description of the units for the data set, as well as, the factors for converting the file units to SI.

The Component Header Data File gives the component kind, name, description, and the analysis date, machine, and program.

2.2 Data Set Type 151 - Header File

Dataset Type: 151

Description: Header File

Record 1: FORMAT(80A1)

Field 1 - Model file name

Record 2: FORMAT(80A1)

Field 1 - Model file description

Record 3: FORMAT(80A1)

Field 1 - Program which created DB

Record 4: FORMAT(10A1,10A1)

Field 1 - Date database created (DD-MMM-YY)
Field 2 - Time database created (HH:MM:SS)

Record 5: FORMAT(10A1,10A1)

Field 1 - Date database last saved (DD-MMM-YY)
Field 2 - Time database last saved (HH:MM:SS)

Record 6: FORMAT(80A1)

Field 1 - Program which created universal file

Record 7: FORMAT(10A1,10A1)

Field 1 - Date universal file written (DD-MMM-YY)
Field 2 - Time universal file written (HH:MM:SS)

2.3 Data Set Type 156 - Units File

Dataset Type: 156

Description: Units File

Record 1: FORMAT(I10,20A1)

Field 1 - Units code
- 1 : SI - METRIC_ABS_(SI)
- 2 : BG - BRITISH_GRAV
- 3 : MG - METRIC_GRAV
- 4 : BA - BRITISH_ABS
- 5 : MM - MODIFIED_SI_(MM)
- 6 : CM - MODIFIED_SI_(CM)
- 7 : IN - BRITISH_GRAV_(MOD)
- 8 : GM - METRIC_GRAV_(MOD)
- 9 : US - USER_DEFINED

Field 2 - Units description (used for documentation only)

Record 2: FORMAT(3E13.5)

Field 1 - Length
Field 2 - Force
Field 3 - Temperature
NOTE:
Unit factor for converting universal file units to SI. To convert from universal file units to SI divide by the appropriate factor listed above.

2.4 Data Set Type 241 - Component Header Data

Dataset Type: 241

Description: Component Header Data

Record 1: FORMAT(I6)

Field 1 - Component Kind
6 - General Matrix

Record 2: FORMAT(2A2)

Field 1 - Component Name (4 character max)

Record 3: FORMAT(40A2)

Field 1 - Component Description (80 character max)

Record 4: FORMAT(5A2)

Field 1 - Analysis Date (dd-mmm-yy)

Record 5: FORMAT(2I6)

Field 1 - Analysis Machine
1 - VAX
2 - CDC
3 - IBM

Field 2 - Analysis Program
1 - NASTRAN
2 - SUPERB
3 - DAGS
4 - FSI
5 - ANSYS

3. GEOMETRY

3.1 Introduction

This section describes the formats of the files needed to define the model geometry, specifically, the Grid Point File (15), the Trace Line File (82), and the Coordinate Trace File (83).

The Grid Point File specifies the location, coordinate system, color, and coordinates of each grid point in the model. The coordinates used in this file must be absolute (global). Local coordinates are not allowed.

The Trace Lines File specifies the trace line number, color, identification, and the entries defining the trace. This file gives the connectivity that is used together with the grid points to display the configuration if the system under test or analysis.

The Coordinate Trace File specifies the coordinate trace number, color, identification, grid point, direction, and sense.

3.2 Data Set Type 15 - Grid Points

Dataset Type: 15

Description: Grid Points

Record 1: FORMAT(4I10,3E13.5)

Field 1	-	node tag number (location label)
Field 2	-	definition coordinate system (>=0)
Field 3	-	displacement coordinate system (>=0)
Field 4	-	color
Field 5-7	-	3-dimensional coordinates of node

Record 1 is repeated for each grid point in the model.

For example:

```
    -1
    15
         1         0         0         8  0.00000E+00  0.00000E+00  0.00000E+00
         2         0         0         8  5.00000E-01  0.00000E+00 -5.00000E-02
         .
         .
         .
       100         0         0         8  1.20000E+01  1.20000E+01 -4.50000E+00
    -1
```

Notes:

1. Any non-zero coordinate system must exist in the SDRC SYSTAN database before this dataset can be read. A value of 0 refers to the entity definition coordinate system.

3.3 Data Set Type 82 - Trace Lines

Dataset Type: 82

Description: Trace Lines

Record 1: FORMAT(3I10)

Field 1 - Trace Line number
Field 2 - Number of entries defining trace
Field 3 - Color

Record 2: FORMAT(80A1)

Field 1 - Identification Line

Record 3: FORMAT(8I10)

Fields 1-N - Entries defining trace

Notes:

1. A non-zero trace line entry means to draw a line to the grid point. A zero trace line entry means to move to the grid point without a draw. A move to the first grid point is implied.
2. The maximum number of entries defining a trace must not exceed 250.
3. SDRC MODAL-PLUS and SDRC MODALX grid point numbers must not exceed 8000.
4. The identification line must not be blank. If no information is required, the word "NONE" must appears in columns 1 through 4.
5. SDRC SYSTAN only uses the first 60 characters of the identification text.
6. SDRC SYSTAN does not process color on a trace line by trace line basis. Each trace line is displayed using the color of the component to which each belongs.
7. SDRC MODAL-PLUS and SDRC MODALX do not support trace lines longer than 125 grid points.

3.4 Data Set Type 83 - Coordinate Trace

Dataset Type: 83

Description: Coordinate Trace

Record 1: FORMAT(3I10)

Field 1 - Coordinate Trace number
Field 2 - Number of entries defining trace
Field 3 - Color

Record 2: FORMAT(80A1)

Field 1 - Identification Line

Record 3: FORMAT(6(I10,2A1))

Field 1 - Grid point number portion of the coordinate specification
Field 2 - Direction identification character (must be "X", "Y", or "Z")
Field 3 - Sense identification character (must be "+" or "-")

Fields 1 through 3 are repeated for each coordinate.

Notes:

1. A coordinate must contain all three fields.
2. The maximum number of entries defining a trace must not exceed 125.
3. SDRC MODAL-PLUS and SDRC MODALX grid point numbers must not exceed 8000.
4. The identification line must not be blank. If no information is required, the word "NONE" must appears in columns 1 through 4.

4. MEASUREMENT

4.1 Introduction

This section describes the format of the Function at Nodal DOF File (58). The file documents any time or frequency domain function with several lines of ID information, response and reference location, direction, and name. The ordinate data type and abscissa spacing along with the specific data type, axis labels, and units are also included in the file.

4.2 Data Set Type 58 - Function at Nodal DOF

Dataset Type: 58

Description: Function at Nodal DOF

Record 1: FORMAT(80A1)

Field 1 - ID Line 1
NOTE:
ID Line 1 is generally used for the function description

Record 2: FORMAT(80A1)

Field 1 - ID Line 2

Record 3: FORMAT(80A1)

Field 1 - ID Line 3
NOTE:
ID Line 3 is generally used to identify when the function was created. The date is in the form DD-MMM-YY, and the time is in the form HH:MM:SS, with a general FORMAT(9A1,1X,8A1).

Record 4: FORMAT(80A1)

Field 1 - ID Line 4

Record 5: FORMAT(80A1)

Field 1 - ID Line 5

Record 6: FORMAT(2(I5,I10),2(1X,10A1,I10,I4))

DOF Identification

Field 1 - Function Type

0	:	General or Unknown
1	:	Time Response
2	:	Auto Spectrum
3	:	Cross Spectrum
4	:	Frequency Response Function
5	:	Transmissibility
6	:	Coherence
7	:	Auto Correlation
8	:	Cross Correlation
9	:	Power Spectral Density (PSD)
10	:	Energy Spectral Density (ESD)
11	:	Probability Density Function
12	:	Spectrum

Field 2 - Function Identification Number

Field 3 - Version Number, or sequence number

Field 4 - Load Case Identification Number
0 : Single Point Excitation

Field 5 - Response Entity Name ("NONE" if unused)

Field 6 - Response Node

Field 7 - Response Direction

0	:	Scalar			
1	:	+X Translation	4	:	+X Rotation
-1	:	-X Translation	-4	:	-X Rotation
2	:	+Y Translation	5	:	+Y Rotation
-2	:	-Y Translation	-5	:	-Y Rotation
3	:	+Z Translation	6	:	+Z Rotation
-3	:	-Z Translation	-6	:	-Z Rotation

Field 8 - Reference Entity Name ("NONE" if unused)

Field 9 - Reference Node

Field 10 - Reference Direction (same as field 7)
NOTE:
Fields 8, 9, and 10 are only relevant if field 4 is zero.

Record 7: *FORMAT(3I10,3E13.5)*

Data Form

Field 1 - Ordinate Data Type

2	:	real, single precision
4	:	real, double precision
5	:	complex, single precision
6	:	complex, double precision

Field 2 - Number of data pairs for uneven abscissa spacing, or number of data values for even abscissa spacing

Field 3 - Abscissa Spacing
0 : uneven
1 : even (no abscissa values stored)

Field 4 - Abscissa minimum (0.0 if spacing uneven)

Field 5 - Abscissa increment (0.0 if spacing uneven)

Field 6 - Z-axis value (0.0 if unused)

Record 8: FORMAT(I10,3I5,2(1X,20A1))

Abscissa Data Characteristics

Field 1 - Specific Data Type

0	:	unknown
1	:	general
2	:	stress
3	:	strain
5	:	temperature
6	:	heat flux
8	:	displacement
9	:	reaction force
11	:	velocity
12	:	acceleration
13	:	excitation force
15	:	pressure
16	:	mass
17	:	time
18	:	frequency
19	:	rpm

Field 2 - Length units exponent

Field 3 - Force units exponent

Field 4 - Temperature units exponent

NOTE:

Fields 2, 3 and 4 are relevant only if the Specific Data Type is General, or in the case of ordinates, the response/reference direction is a scalar. See Appendix 'A' for the units exponent table.

Field 5 - Axis label ("NONE" if not used)

Field 6 - Axis units label ("NONE" if not used)

NOTE:

If fields 5 and 6 are supplied, they take precendence over program generated labels and units.

Record 9: FORMAT(I10,3I5,2(1X,20A1))

Ordinate (or ordinate numerator) Data Characteristics

Record 10: FORMAT(I10,3I5,2(1X,20A1))

Ordinate Denominator Data Characteristics

Record 11: FORMAT(I10,3I5,2(1X,20A1))

Z-axis Data Characteristics

NOTE:

Records 9, 10, and 11 are always included and have fields the same as record 8. If records 10 and 11 are not used, set field 1 to zero.

Record 12:

	Data Values			
	Ordinate		Abscissa	
Case	Type	Precision	Spacing	FORMAT
1	real	single	even	6E13.5
2	real	single	uneven	6E13.5
3	complex	single	even	6E13.5
4	complex	single	uneven	6E13.5
5	real	double	even	4E20.12
6	real	double	uneven	2(E13.5,E20.12)
7	complex	double	even	4E20.12
8	complex	double	uneven	E13.5,2E20.12

NOTE: See Appendix 'B' for typical FORTRAN READ/WRITE statements for each case.

General Notes:

1. ID lines may not be blank. If no information is required, the word "NONE" must appear in columns 1 through 4.
2. ID line 1 appears on plots in OUTPUT DISPLAY.
3. Dataloaders use the following ID line conventions
   - ID Line 1 - Model Identification
   - ID Line 2 - Run Identification
   - ID Line 3 - Run Date and Time
   - ID Line 4 - Load Case Name
4. Coordinates codes from SDRC MODAL-PLUS and SDRC MODALX are decoded into node (grid point) and direction.
5. Entity names used in SDRC SYSTAN have a 4 character maximum.

5. ANALYSIS

5.1 Introduction

This section defines the format of the Analysis Data at Nodes File (55) and the Entry Definition Matrix File (250).

The Analysis Data at Nodes File describes the specific kind of analysis performed, as well as, the data model and characteristics, specific data type, and number of data values.

The Entry Definition Matrix File provides the information for the matrix identifier, matrix data type, and the number of rows and columns. The file also contains the information necessary for the submatrix reconstruction and the actual matrix data. This file is useful for storing any matrix information that might be required. One example would be the storage of inertia restraint or residual flexibility terms associated with a Frequency Response Function matrix.

5.2 Data Set Type 55 - Analysis Data at Nodes

Dataset Type: 55

Description: Analysis Data at Nodes

Record 1: FORMAT(80A1)

Field 1 - ID Line 1

Record 2: FORMAT(80A1)

Field 1 - ID Line 2

Record 3: FORMAT(80A1)

Field 1 - ID Line 3

Record 4: FORMAT(80A1)

Field 1 - ID Line 4

Record 5: FORMAT(80A1)

Field 1 - ID Line 5

Record 6: FORMAT(6I10)

Data Definition Parameters

Field 1 - Model Type
0 : Unknown
1 : Structural
2 : Heat Transfer
3 : Fluid Flow

Field 2 - Analysis Type
0 : Unknown
1 : Static
2 : Normal Mode
3 : Complex Eigenvalue, first order
-3 : Complex Eigenvalue, first order (conjugate pairs)
4 : Transient Response
5 : Frequency Response
6 : Buckling
7 : Complex Eigenvalue, second order

Field 3 - Data Characteristics
- 0 : Unknown
- 1 : Scalar
- 2 : 3 DOF Global Translation Vector
- 3 : 6 DOF Global Translation and Rotation Vector
- 4 : Symmetric Global Tensor
- 5 : General Global Tensor

Field 4 - Specific Data Type
- 0 : Unknown
- 1 : General
- 2 : Stress
- 3 : Strain
- 4 : Elemental Force
- 5 : Temperature
- 6 : Heat Flux
- 7 : Strain Energy
- 8 : Displacement
- 9 : Reaction Force
- 10 : Kinetic Energy
- 11 : Velocity
- 12 : Acceleration

Field 5 - Data Type
- 2 : Real
- 5 : Complex

Field 6 - Number of data values per node (NDV)

Records 7 and 8 are analysis type specific.

General Form

Record 7: FORMAT(8I10)

Field 1 - Number of integer data values
1 < or = NINT < or = 10
Field 2 - Number of Real data values
1 < or = NRVAL < or = 12
Fields 3-N - Type specific integer parameters

Record 8: FORMAT(6E13.5)

Fields 1-N - Type specific real parameters

For Analysis Type = 0, Unknown

Record 7:

Field 1	-	1
Field 2	-	1
Field 3	-	ID Number

Record 8:

Field 1	-	0.0

For Analysis Type = 1, Static

Record 7:

Field 1	-	1
Field 2	-	1
Field 3	-	Load Case Number

Record 8:

Field 1	-	0.0

For Analysis Type = 2, Normal Mode

Record 7:

Field 1	-	2
Field 2	-	4
Field 3	-	Load Case Number
Field 4	-	Mode Number

Record 8:

Field 1	-	Frequency (Hertz)
Field 2	-	Modal Mass (see note 17)
Field 3	-	Modal Viscous Damping Ratio
Field 4	-	Modal Hysteretic Damping Ratio

For Analysis Type = 3, Complex Eigenvalue, first order

Record 7:

Field 1	- 2
Field 2	- 6
Field 3	- Load Case Number
Field 4	- Mode Number

Record 8:

Field 1	- Real Part of Eigenvalue
Field 2	- Imaginary Part of Eigenvalue
Field 3	- Real Part of Modal A (see note 18)
Field 4	- Imaginary Part of Modal A
Field 5	- Real Part of Modal B (see note 18)
Field 6	- Imaginary Part of Modal B

For Analysis Type = 4, Transient Response

Record 7:

Field 1	- 2
Field 2	- 1
Field 3	- Load Case Number
Field 4	- Time Step Number

Record 8:

Field 1	- Time (seconds)

For Analysis Type = 5, Frequency Response

Record 7:

Field 1	- 2
Field 2	- 1
Field 3	- Load Case Number
Field 4	- Frequency Step Number

Record 8:

Field 1	- Frequency (Hertz)

For Analysis Type = 6, Buckling

Record 7:

Field 1	- 1
Field 2	- 1
Field 3	- Load Case Number

Record 8:

Field 1	- Eigenvalue

For Analysis Type = 7, Complex Eigenvalue, second order

Record 7:

Field 1	-	2
Field 2	-	6
Field 3	-	Load Case Number
Field 4	-	Mode Number

Record 8:

Field 1	-	Real Part of Eigenvalue
Field 2	-	Imaginary Part of Eigenvalue
Field 3	-	Real Part of Modal A (see note 18)
Field 4	-	Imaginary Part of Modal A
Field 5	-	Real Part of Modal B (see note 18)
Field 6	-	Imaginary Part of Modal B

Record 9: FORMAT(I10)
Field 1 - Node Number

Record 10: FORMAT(6E13.5)
Fields 1-N - Data at this Node
(NDV Real or Complex Values)

Records 9 and 10 are repeated for each node.
Notes:

1. ID Lines may not be blank. If no information is required, the word "NONE" must appear in columns 1 through 4.

2. For complex data there will be 2*NDV data items at each node. The order is real part for VALUE 1, imaginary part for VALUE 1, real part for VALUE 2, imaginary part for VALUE 2, etc.

3. The order of values for various data characteristics is:

3 DOF GLOBAL VECTOR:	X, Y, Z
6 DOF GLOBAL VECTOR:	X, Y, Z, RX, RY, RZ
SYMMETRIC GLOBAL TENSOR:	SXX, SXY, SYY, SXZ, SYZ, SZZ
GENERAL GLOBAL TENSOR:	SXX, SYX, SZX, SXY, SYY, SZY, SXZ, SYZ, SZZ

4. ID Line 1 always appears on plots in OUTPUT DISPLAY.

5. If specific data type is "UNKNOWN", ID Line 2 is displayed as data type in OUTPUT DISPLAY.

6. Typical FORTRAN I/O statements for the data section are:

```
     READ (LUN,1000) NUM
     WRITE
1000 FORMAT (I10)
     READ (LUN,1010) (VAL(I),I=1,NDV)
     WRITE
1010 FORMAT (6E13.5)
```

where: NUM is node number

VAL is real or complex data array
NDV is number of data values per node

7. Data characteristic values imply the following values of NDV:

   3 DOF GLOBAL VECTOR: 3
   6 DOF GLOBAL VECTOR: 6
   SYMMETRIC GLOBAL TENSOR: 6
   GENERAL GLOBAL TENSOR: 9

8. Data associated with SDRC MODAL-PLUS and SDRC MODALX has the following special form of ID Line 5.

   FORMAT (4I10)

   Field 1 : Reference Coordinate Label (1-8000)

   Field 2 : Reference Coordinate Direction
   1 : +X Direction
   2 : -X Direction
   3 : +Y Direction
   4 : -Y Direction
   5 : +Z Direction
   6 : -Z Direction

   Field 3 : Numerator Signal Code
   0 : unknown
   2 : stress
   3 : strain
   5 : temperature
   8 : displacement
   11 : velocity
   12 : acceleration
   13 : excitation force
   15 : pressure

   Field 4 : Denominator Signal Code
   0 : unknown
   2 : stress
   3 : strain
   5 : temperature
   8 : displacement
   11 : velocity
   12 : acceleration
   13 : excitation force
   15 : pressure

9. ID Line 5 for SDRC MODAL-PLUS and SDRC MODALX, and the information included in record 6 is provided only to inform the user. The data is not used to alter the modal parameters on record 8. The modal parameters on record 8 are accepted exactly as entered.

10. Any record with all 0.0 data entries need not, but may appear.

11. A direct result of the previous note is that if no records 9 and 10 appear, all data for the data set is 0.0.

12. When new analysis types are added, record 7 fields 1 and 2 are always > or = 1 with dummy integer and real data if data is not required. If complex data is needed, it is treated as two real numbers, real part followed imaginary point.

13. Data loaders use the following ID line convention:

    1. (80A1) Model Indentification
    2. (80A1) Run Identification
    3. (80A1) Run Date/Time
    4. (80A1) Load Case Name

    For Static:

    5. (17H LOAD CASE NUMBER:, I10)

    For normal mode:

    5. (10H mode same, I10, 10H frequency, E13.5)

14. Maximum value for NDV is 9.

15. Typical FORTRAN I/O statements for processing records 7 and 8 are:

```
      READ (LUN,1000) NINT,NRVAL,(IPAR(I),I=1,NINT)
1000  FORMAT (8I10)
      READ (LUN,1010) (RPAV(I),I=1,NRVAL)
1010  FORMAT (6E13.5)
```

16. For situations with reduced number DOF, use 3 DOF translations or 6 DOF tranlation and rotation with unused values equal to 0.0.

17. Record 8 for real mode shapes contains the resonance frequency, modal mass, and modal viscous damping ratio. The modal mass is calculated based on the following relations for each data type. The data type is taken from the modal parameter data set, not the mode shape data set.

$$m_r = \frac{X_1 * X_2}{2*A*\Omega_r*\sqrt{1-\varsigma^2}} \qquad \text{for D/F}$$

$$m_r = \frac{X_1 * X_2}{2*A*\sqrt{1-\varsigma^2}} \qquad \text{for V/F}$$

$$m_r = \frac{\Omega_r * X_1 * X_2}{2*A*\sqrt{1-\varsigma^2}} \qquad \text{for A/F}$$

where :

m_r = modal mass for mode r
X_1 = mode shape coefficient of reference coordinate
X_2 = mode shape coefficient of response coordinate
Ω_r = undamped natural frequency in rad/sec
A = residue amplitude, or modal amplitude
ς = modal viscous damping ratio

18. Record 8 for complex mode shapes contains the complex eigenvalue, the complex Modal A value, and the complex Modal B value. The complex eigenvalue is calculated through the following relation.

$$s = -\varsigma*\Omega_r + j*\Omega_r*\sqrt{1-\varsigma^2}$$

where :

s = complex eigenvalue
Ω_r = undamped natural frequency in rad/sec
ζ = modal viscous damping ratio

The complex Modal A value is calculated based on the following relations for each data type. The data type is taken from the modal parameter data set, not the mode shape data set.

$$MA_r = \frac{X_1 * X_2}{A} \qquad \text{for D/F}$$

$$MA_r = \frac{j * \Omega_r * X_1 * X_2}{A} \qquad \text{for V/F}$$

$$MA_r = \frac{-\Omega_r^2 * X_1 * X_2}{A} \qquad \text{for A/F}$$

where :

MA_r = complex Modal A value for mode r
X_1 = complex mode shape coefficient of reference coordinate
X_2 = complex mode shape coefficient of response coordinate
Ω_r = undamped natural frequency in rad/sec
A = complex residue (residue amplitude and phase)

The complex Modal B value is the product of the complex eigenvalue and the complex Modal A value.

5.3 Data Set Type 250 - Entry Definition Matrix

Dataset Type: 250

Description: Entry Definition Matrix

Record 1: FORMAT(I10)

Field 1 - Matrix Identifier (IMAT) - Refer to Table E-2

Record 2: FORMAT(5I10)

Field 1 - Matrix Data Type (MDTYPE)
- 1 - Integer
- 2 - Real
- 4 - Double Precision
- 5 - Complex
- 6 - Complex Double Precision

Field 2 - Matrix Form (MFORM)
- 3 - General Rectangular

Field 3 - No. of Rows (NROWS)
Field 4 - No. of Cols (NCOLS)
Field 5 - Storage Key (MKEY)
- 1 - Row
- 2 - Column (suggested)

Record 3: FORMAT(6I10)

Field 1 - Starting Row for Submatrix (ISR)
Field 2 - Starting Column for Submatrix (ISC)
Field 3 - No. of Rows in Submatrix (NR)
Field 4 - No. of Columns in Submatrix (NC)
Field 5 - Submatrix Form (MFORMS)
- 3 - General Rectangular
- 5 - Diagonal

Field 6 - Submatrix Storage Key (MKEYS)
- 1 - Row
- 2 - Column (suggested)

Record 4: Matrix Data

FORMAT(8I10)	INTEGER
FORMAT(4E20.12)	REAL
FORMAT(4D20.12)	DOUBLE PRECISION
FORMAT(2(2E20.12))	COMPLEX
FORMAT(2(2D20.12))	COMPLEX DOUBLE PRECISION

(Record 4 repeated as necessary to fulfill requirements of record 3)

(Records 3 and 4, as a group, are repeated as necessary to define all non-zero submatrices)

Notes:

1. Submatrix data is added to current components.

2. Submatrix data not present is assumed equal to zero. If records 3 and 4 are not present, a zero matrix is created.

3. Matrix 148 will have its diagonal overwritten with the identity matrix [I]. The independent-independent portion of matrix 31 will be overwritten with [I].

Table E-2. Valid Matrices for SDRC SYSTAN Components.

IMAT	Description	Component					
		FM	FS	E	S	R	G
6	Mass		•		•		
7	Viscous		•		•		
8	Histeretic		•		•		
9	Stiffness		•		•		
11	Modal Displacement	•		•			
13	Modal Mass	•		•			
14	Modal Viscous	•		•			
15	Modal Hysteretic	•		•			
16	Modal Stiffness	•		•			
31	Rigid Body Constraint					•	
32	Rigid Body Mass					•	
131	Mass (I-I)						•
132	Mass (I-D)						•
133	Mass (D-I)						•
134	Mass (D-D)						•
135	Viscous (I-I)						•
136	Viscous (I-D)						•
137	Viscous (D-I)						•
138	Viscous (D-D)						•
139	Stiffness (I-I)						•
140	Stiffness (I-D)						•
141	Stiffness (D-I)						•
142	Stiffness (D-D)						•
143	Hysteretic (I-I)						•
144	Hysteretic (I-D)						•
145	Hysteretic (D-I)						•
146	Hysteretic (D-D)						•
147	Constraint (D-I)						•
148	Constraint (D-D)						•

Key

I	-	Independent	E	-	Experimental modal synthesis
D	-	Dependent	S	-	SDRC SYSTAN finite element
FM	-	Finite element modal synthesis	R	-	Rigid body
FS	-	Finite element substructure	G	-	General matrix

6. REFERENCES

[1] *SDRC I-deas Level 3 User's Guide,* "Section VI, Universal File Datasets", 1986, pp. 306-470.

[2] *Reference Manual for Modal-Plus 9.0,* "Appendix A, SDRC Universal File Formats", SDRC GE-CAE International, 1985, 26 pp.

[3] Ghijs, C., Helpenstein, H., Splid, A., Maanen, J.; *Design of Neutral File 1 to 8,* Rutherford Appleton Laboratory, CAD*I Paper RAL-012-85, 1985, 10 pp.

[4] Leuridan, J., *Contents of the Common Database for Experimental Modal Analysis,* Leuven Measurement and Systems, CAD*I Paper LMS-007-85, 1985.

[5] *Proposal for ESPRIT CAD Interfaces,* ESPRIT Project Reference Number 5.2.1, Technical Annex, 1984.

[6] Heylen, W., *Preliminary List of Keywords for Neutral Files 7 and 8,* CAD*I Paper KUL-017-85, 1985.

NOMENCLATURE

Matrix Notation

{..}	braces enclose column vector expressions
$\{..\}^T$	row vector expressions
[..]	brackets enclose matrix expressions
$[..]^H$	complex conjugate transpose, or Hermitian transpose, of a matrix
$[..]^T$	
$[..]^{-1}$	inverse of a matrix
$[..]^+$	generalized inverse (pseudoinverse)
$[..]_{q \times p}$	size of a matrix: q rows, p columns
$\lceil .. \rfloor$	diagonal matrix

Operator Notation

A^*	complex conjugate
F	Fourier transform
F^{-1}	inverse Fourier transform
H	Hilbert transform
H^{-1}	inverse Hilbert transform
ln	natural logrithm
L	Laplace transform
L^{-1}	inverse Laplace transform
Re + jIm	complex number: real part "Re", imaginary part "Im"
$\dot{x}$	first derivative with respect to time of dependent variable x
$\ddot{x}$	second derivative with respect to time of dependent variable x
$\overline{y}$	mean value of y
$\hat{y}$	estimated value of y
$\sum_{i=1}^{n} A_i B_i$	summation of $A_i B_i$ from i = 1 to n
$\frac{\partial}{\partial t}$	partial derivative with respect to independent variable "t"
det[..]	determinant of a matrix
$\lVert .. \rVert_2$	Euclidian norm

Roman Alphabet

A_{pqr}	residue for response location p, reference location q, of mode r
C	damping
COH	ordinary coherence function†
COH_{ik}	ordinary coherence function between any signal i and any signal k†
COH^n	conditioned partial coherence†
e	base e (2.71828...)
F	input force
F_q	spectrum of q^{th} reference†
GFF	auto power spectrum of reference†

GFF_{qq}	auto power spectrum of reference q†
GFF_{ik}	cross power spectrum of reference i and reference k†
$[GFFX]$	reference power spectrum matrix augmented with the response/reference cross power spectrum vector for use in Gauss elimination
GXF	cross power spectrum of response/reference†
GXX	auto power spectrum of response†
GXX_{pp}	auto power spectrum of response p†
$h(t)$	impulse response function†
$h_{pq}(t)$	impulse response function for response location p, reference location q †
$H(s)$	transfer function†
$H(\omega)$	frequency response function, when no ambiguity exist, H is used instead of $H(\omega)$†
$H_{pq}(\omega)$	frequency response function for response location p, reference location q, when no ambiguity exist, H_{pq} is used instead of $H_{pq}(\omega)$†
$H_1(\omega)$	frequency response function estimate with noise assumed on the response, when no ambiguity exist, H_1 is used instead of $H_1(\omega)$†
$H_2(\omega)$	frequency response function estimate with noise assumed on the reference, when no ambiguity exist, H_2 is used instead of $H_2(\omega)$†
$H_S(\omega)$	scaled frequency response function estimate, when no ambiguity exist, H_S is used instead of $H_S(\omega)$†
$H_v(\omega)$	frequency response function estimate with noise assumed on both reference and response, when no ambiguity exist, H_v is used instead of $H_v(\omega)$†
$[I]$	identity matrix
j	$\sqrt{-1}$
K	stiffness
L	modal participation factor
M	mass
M_r	modal mass for mode r
$MCOH$	multiple coherence function†
N	number of modes
N_i	number of references (inputs)
N_o	number of responses (outputs)
p	output, or response point (subscript)
q	input, or reference point (subscript)
r	mode number (subscript)
R_I	residual inertia
R_F	residual flexibility
s	Laplace domain variable
t	independent variable of time (sec)
t_k	discrete value of time (sec) $t_k = k\,\Delta t$
T	sample period
x	displacement in physical coordinates
X	response
X_p	spectrum of p^{th} response†
z	Z domain variable

Greek Alphabet

$\delta(t)$	Dirac impulse function
Δf	discrete interval of frequency (Hertz or cycles/sec)
Δt	discrete interval of sample time (sec)
ϵ	small number

η	noise on the output
λ_r	r^{th} complex eigenvalue, or system pole $\lambda_r = \sigma_r + j\omega_r$
$[\Lambda]$	diagonal matrix of poles in Laplace domain
ν	noise on the input
ω	variable of frequency (rad/sec)
ω_r	imaginary part of the system pole, or damped natural frequency, for mode r (rad/sec) $\omega_r = \Omega_r \sqrt{1 - \varsigma_r^2}$
Ω_r	undamped natural frequency (rad/sec) $\Omega_r = \sqrt{\sigma_r^2 + \omega_r^2}$
ϕ_{pr}	scaled p^{th} response of normal modal vector for mode r
$\{\phi\}_r$	scaled normal modal vector for mode r
$[\Phi]$	scaled normal modal vector matrix
$\{\psi\}$	scaled eigenvector
ψ_{pr}	scaled p^{th} response of a complex modal vector for mode r
$\{\psi\}_r$	scaled complex modal vector for mode r
$[\Psi]$	scaled complex modal vector matrix
σ	variable of damping (rad/sec)
σ_r	real part of the system pole, or damping factor, for mode r
ς	damping ratio
ς_r	damping ratio for mode r
†	vector implied by definition of function

APPENDIX A: UNITS CONVERSION

In order to correctly perform units conversion, length, force, and temperature exponents must be supplied for a specific data type of General; that is, Record 8 Field 1 = 1. For example, if the function has the physical dimensionality of Energy (Force * Length), then the required exponents would be as follows:

Length = 1
Force = 1 Energy = L * F
Temperature = 0

Units exponents for the remaining specific data types should not be supplied. The following exponents will automatically be used.

Table - Unit Exponents

Specific Data Type	Direction: Translational Length	Translational Force	Translational Temp	Rotational Length	Rotational Force	Rotational Temp
0	0	0	0	0	0	0
1	(requires input to fields 2,3,4)					
2	-2	1	0	-1	1	0
3	0	0	0	0	0	0
5	0	0	1	0	0	1
6	1	1	0	1	1	0
8	1	0	0	0	0	0
9	0	1	0	1	1	0
13	0	1	0	1	1	0
15	-2	1	0	-1	1	0
16	-1	1	0	1	1	0
17	0	0	0	0	0	0
18	0	0	0	0	0	0
19	0	0	0	0	0	0

NOTE: Units exponents for scalar points are defined within SDRC SYSTAN prior to reading this dataset.

APPENDIX B: READ/WRITE OPERATIONS

There are 8 distinct combinations of parameters which affect the details of READ/WRITE operations. The parameters involved are Ordinate Data Type, Ordinate Data Precision, and Abscissa Spacing. Each combination is documented in the examples below. In all cases, the number of data values (for even abscissa spacing) or data pairs (for uneven abscissa spacing) is NVAL. The abscissa is always real single precision. Complex double precision is handled by two real double precision variables (real part followed by imaginary part) because most systems do not directly support complex double precision.

CASE 1

REAL
SINGLE PRECISION
EVEN SPACING

Order of data in file

```
Y1  Y2  Y3  Y4  Y5  Y6
Y7  Y8  Y9  Y10 Y11 Y12
.
.
.
```

Input

```
      REAL Y(6)
      .
      .
      .
      NPRO=0
   10 READ(LUN,1000,ERR= ,END= )(Y(I),I=1,6)
 1000 FORMAT(6E13.5)
      NPRO=NPRO+6
      .
      .  code to process these six values
      .
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```

Output

```
      REAL Y(6)
      .
      .
      .
      NPRO=0
   10 CONTINUE
      .
      .  code to set up these six values
      .
      WRITE(LUN,1000,ERR= )(Y(I),I=1,6)
 1000 FORMAT(6E13.5)
      NPRO=NPRO+6
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```

CASE 2

REAL
SINGLE PRECISION
UNEVEN SPACING

Order of data in file

```
X1  Y1  X2  Y2  X3  Y3
X4  Y4  X5  Y5  X6  Y6
.
.
.
```

Input

```
      REAL X(3),Y(3)
      .
      .
      .
      NPRO=0
   10 READ(LUN,1000,ERR= ,END= )(X(I),Y(I),I=1,3)
 1000 FORMAT(6E13.5)
      NPRO=NPRO+3
      .
      .  code to process these three values
      .
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```

Output

```
      REAL X(3),Y(3)
      .
      .
      .
      NPRO=0
   10 CONTINUE
      .
      .  code to set up these three values
      .
      WRITE(LUN,1000,ERR= )(X(I),Y(I),I=1,3)
 1000 FORMAT(6E13.5)
      NPRO=NPRO+3
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```

CASF 3

COMPLEX
SINGLE PRECISION
EVEN SPACING

Order of data in file

```
RY1 IY1 RY2 IY2 RY3 IY3
RY4 IY4 RY5 IY5 RY6 IY6
.
.
.
```

Input

```
         COMPLEX Y(3)
         .
         .
         .
         NPRO=0
    10   READ(LUN,1000,ERR=  ,END=  )(Y(I),I=1,3)
  1000   FORMAT(6E13.5)
         NPRO=NPRO+3
         .
         .  code to process these six values
         .
         IF(NPRO.LT.NVAL)GO TO 10
         .
         .  continued processing
         .
```

Output

```
         COMPLEX Y(3)
         .
         .
         .
         NPRO=0
    10   CONTINUE
         .
         .  code to set up these three values
         .
         WRITE(LUN,1000,ERR=  )(Y(I),I=1,3)
  1000   FORMAT(6E13.5)
         NPRO=NPRO+3
         IF(NPRO.LT.NVAL)GO TO 10
         .
         .  continued processing
         .
```

CASE 4

COMPLEX
SINGLE PRECISION
UNEVEN SPACING

Order of data in file

```
X1  RY1 IY1 X2  RY2 IY2
X3  RY3 IY3 X4  RY4 IY4
.
.
.
```

Input

```
        REAL X(2)
        COMPLEX Y(2)
        .
        .
        .
        NPRO=0
     10 READ(LUN,1000,ERR=  ,END=  )(X(I),Y(I),I=1,2)
   1000 FORMAT(6E13.5)
        NPRO=NPRO+2
        .
        .  code to process these two values
        .
        IF(NPRO.LT.NVAL)GO TO 10
        .
        . continued processing
        .
```

Output

```
        REAL X(2)
        COMPLEX Y(2)
        .
        .
        .
        NPRO=0
     10 CONTINUE
        .
        .  code to set up these two values
        .
        WRITE(LUN,1000,ERR=  )(X(I),Y(I),I=1,2)
   1000 FORMAT(6E13.5)
        NPRO=NPRO+2
        IF(NPRO.LT.NVAL)GO TO 10
        .
        . continued processing
        .
```

CASE 5

REAL
DOUBLE PRECISION
EVEN SPACING

Order of data in file

```
Y1  Y2  Y3  Y4
Y5  Y6  Y7  Y8
.
.
.
```

Input

```
      DOUBLE PRECISION Y(4)
      .
      .
      .
      NPRO=0
   10 READ(LUN,1000,ERR=  ,END=  )(Y(I),I=1,4)
 1000 FORMAT(4E20.12)
      NPRO=NPRO+4
      .
      .  code to process these four values
      .
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```

Output

```
      DOUBLE PRECISION Y(4)
      .
      .
      .
      NPRO=0
   10 CONTINUE
      .
      .  code to set up these four values
      .
      WRITE(LUN,1000,ERR=  )(Y(I),I=1,4)
 1000 FORMAT(4E20.12)
      NPRO=NPRO+4
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```

CASE 6

REAL
DOUBLE PRECISION
UNEVEN SPACING

Order of data in file

```
X1  Y1   X2  Y2
X3  Y3   X4  Y4
.
.
.
```

Input

```
      REAL X(2)
      DOUBLE PRECISION Y(2)
      .
      .
      .
      NPRO=0
   10 READ(LUN,1000,ERR=  ,END=  )(X(I),Y(I),I=1,2)
 1000 FORMAT(2(E13.5,E20.12))
      NPRO=NPRO+2
      .
      .  code to process these two values
      .
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```

Output

```
      REAL X(2)
      DOUBLE PRECISION Y(2)
      .
      .
      .
      NPRO=0
   10 CONTINUE
      .
      .  code to set up these two values
      .
      WRITE(LUN,1000,ERR=  )(X(I),Y(I),I=1,2)
 1000 FORMAT(2(E13.5,E20.12))
      NPRO=NPRO+2
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```

CASE 7

COMPLEX
DOUBLE PRECISION
EVEN SPACING

Order of data in file

```
RY1  IY1  RY2  IY2
RY3  IY3  RY4  IY4
.
.
.
```

Input

```
      DOUBLE PRECISION Y(2,2)
      .
      .
      .
      NPRO=0
   10 READ(LUN,1000,ERR= ,END= )((Y(I,J),I=1,2),J=1,2)
 1000 FORMAT(4E20.12)
      NPRO=NPRO+2
      .
      .  code to process these two values
      .
      IF(NPRO.LT.NVAL)GO TO 10
      .
      . continued processing
      .
```

Output

```
      DOUBLE PRECISION Y(2,2)
      .
      .
      .
      NPRO=0
   10 CONTINUE
      .
      .  code to set up these two values
      .
      WRITE(LUN,1000,ERR= )((Y(I,J),I=1,2),J=1,2)
 1000 FORMAT(4E20.12)
      NPRO=NPRO+2
      IF(NPRO.LT.NVAL)GO TO 10
      .
      . continued processing
      .
```

CASE 8

COMPLEX
DOUBLE PRECISION
UNEVEN SPACING

Order of data in file

```
X1  RY1  IY1
X2  RY2  IY2
.
.
.
```

Input

```
      REAL X
      DOUBLE PRECISION Y(2)
      .
      .
      .
      NPRO=0
10    READ(LUN,1000,ERR=  ,END=  )(X,Y(I),I=1,2)
1000  FORMAT(E13.5,2E20.12)
      NPRO=NPRO+1
      .
      .  code to process this value
      .
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```

Output

```
      REAL X
      DOUBLE PRECISION Y(2)
      .
      .
      .
      NPRO=0
10    CONTINUE
      .
      .  code to set up this value
      .
      WRITE(LUN,1000,ERR=  )(X,Y(I),I=1,2)
1000  FORMAT(E13.5,2E20.12)
      NPRO=NPRO+1
      IF(NPRO.LT.NVAL)GO TO 10
      .
      .  continued processing
      .
```