AFRL-VA-WP-TR-2000-3044

DEVELOPMENT OF THE AERODYNAMIC/AEROSERVOELASTIC MODULES IN ASTROS

VOLUME II - AEROSERVOELASTICITY DISCIPLINE IN ASTROS PROGRAMMER'S MANUAL

ZONA TECHNOLOGY INC
7434 E STETSON DR SUITE 205
SCOTTSDALE AZ 85251

February 1999

FINAL REPORT FOR 9/1/96 -- 9/30/98

Air Vehicles Directorate
Air Force Research Laboratory
Air Force Materiel Command
Wright-Patterson AFB, OH 45433-7542

NOTICE

USING GOVERNMENT DRAWINGS, SPECIFICATIONS, OR OTHER DATA INCLUDED IN THIS DOCUMENT FOR ANY PURPOSE OTHER THAN GOVERNMENT PROCUREMENT DOES NOT IN ANY WAY OBLIGATE THE UNITED STATES GOVERNMENT. THE FACT THAT THE GOVERNMENT FORMULATED OR SUPPLIED THE DRAWINGS, SPECIFICATIONS, OR OTHER DATA DOES NOT LICENSE THE HOLDER OR ANY OTHER PERSON OR CORPORATION; OR CONVEY ANY RIGHTS OR PERMISSION TO MANUFACTURE, USE, OR SELL ANY PATENTED INVENTION THAT MAY BR RELATED TO THEM.

THIS REPORT IS RELEASEABLE 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.

Victoria A. Tischler
VICTORIA A. TISCHLER
Aerospace Engineer
Structural Design & Development Branch

Nelson D. Wolf
NELSON D. WOLF, Chief
Structural Design and Development Branch
Structures Division

Joseph M. Manter
JOSEPH M. MANTER, Chief
Structures Division
Air Vehicles Directorate

COPIES OF THIS REPORT SHOULD NOT BE RETURNED UNLESS RETURN IS REQUIRED BY SECURITY CONSIDERATIONS, CONTRACTUAL OBLIGATIONS, OR NOTICE ON A SPECIFIED DOCUMENT.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	FEBRUARY 04, 1999	FINAL 24 SEP 1996 - SEP 1998

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
DEVELOPMENT OF THE AERODYNAMIC/AEROSERVOELASTIC MODULES IN ASTROS / VOLUME II - AEROSERVOELASTICITY DISCIPLINE IN ASTROS PROGRAMMER'S MANUAL	C: F33615-96-C-3217 PE: 65502F PR: STTR TA: 41 WU: 00
6. AUTHOR(S) M. Karpel and B. Moulin Technion - I.I.T.	

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)		8. PERFORMING ORGANIZATION REPORT NUMBER
Subcontractor to Research Insitutute: Technion - Israel Institute of Technology Haifa 32000 Israel Tel 972-4-8293490 / Fax 972-4-8229352	Prime Contractor: ZONA Technology, Inc. 7434 E. Stetson Drive, Suite 205 Scottsdale, AZ 85251 Tel (602) 945-9988 / Fax (602) 945-6588	ZONA 99-11F

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
Air Vehicles Directorate Air Force Research Laboratory Air Force Materiel Command Wright Patterson Air Force Base OH 45433-7531 POC: Capt. Gerald Andersen, (937) 255-6992 / Dr. V.B.Venkayya (937) 255-2582	AFRL-VA-WP-TR-2000-3045

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED	

13. ABSTRACT *(Maximum 200 words)*

This report describes the new ASE discipline in ASTROS and the way it was implemented in this project. The MAPOL commands added for the implementation of the ASE discipline are given and explained. A flow chart of the new data templates is given and each one of them is then described separately. This report is part of the documentation which describe the complete development of an STTR Phase II effort entitled "Development of the Aerodynamic/Aeroservoelastic Modules in ASTROS". Additional aeroservoelasticity (ASE) reports are the Theoretical Manual, the User's Manual, and the Application Manual.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Multidisciplinary Optimization, Aeroelasticity, Aeroservoelasticity, Gust Response, ASTROS	39
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

FOREWORD

This final report is submitted in fulfillment of CDRL CLIN 0001, Data Item A001, Title: Scientific and Technical Reports of a Small business Technology Transfer (STTR) Phase II contract No. F33615-96-C-3217 entitled, "Development of the Aerodynamic/Aeroservoelastic Modules in ASTROS," covering the performance period from 24 September 1996 to 24 September 1998.

This work is the second phase of a continuing two-phase STTR contract supported by AFRL/Wright-Patterson. The first phase STTR contract No. F33615-95-C-3219 entitled, "Enhancement of the Aeroservoelastic Capability in ASTROS," was completed in May 1996 and published as WL-TR-96-3119.

Both STTR Phase I and Phase II contracts are performed by the same ZONA Team in which ZONA Technology, Inc. is the prime contractor, whereby the team members include: the University of Oklahoma (OU), Universal Analytics, Inc. (UAI), and Technion (I.T.T.).

This final report consists of eight volumes, these are:

*ASTROS**

Volume I	-	ZAERO User's Manual
Volume II	-	ZAERO Programmer's Manual
Volume III	-	ZAERO Application Manual
Volume IV	-	ZAERO Theoretical Manual

ASTROServo

Volume I	-	Aeroservoelastic Discipline in ASTROS, User's Manual
Volume II	-	Aeroservoelastic Discipline in ASTROS, Programmer's Manual
Volume III	-	Aeroservoelastic Discipline in ASTROS, Application Manual
Volume IV	-	Aeroservoelastic Discipline in ASTROS, Theoretical Manual

This document (Volume II) is the Programmer's Manual of the Aeroservoelastic (ASE) interaction module developed to facilitate ASE analysis and the application of ASE stability and response constraints within ASTROS.

At AFRL/Wright-Patterson, Captain Gerald Andersen was the contract monitor and Dr. V. B. Venkayya was the initiator of the whole STTR effort. The technical advice and assistance received from Mr. Doug Niell of the MacNeal Schwendler Corporation, Dr. V. B. Venkayya and others from AFRL during the course of the present phase on the development of ASTROS* are gratefully acknowledged.

Contents

Chapter 1

Introduction

The Programmer's Manual describes the new engineering application modules and data base entities used by the new aeroservoelastic (ASE) module.

Chapter 2

Engineering Application Modules

2.1 Summary of new application modules

The engineering application modules added to support the ASE module are:

ASEDRV	driver for ASE options
ASEFLUT	ASE analysis w/ or w/o control effects
ASERUN	ASE run options
ASESENS	ASE constraint sensitivity
BDMAT	baseline dynamic matrices
CNTMOD	aerodynamic forces and mass matrices associated with control modes
GMAT	baseline structural matrices
GMGKUPD	update generalized matrices
GSENS*	sensitivity of generalized matrices
GSTMOD	aerodynamic forces associated with gust modes
MGFL	update dynamic matrices
MIST	aerodynamic approximations
RMODE	reduced-size modal analysis
UPDV	design variable update

* - a MAPOL procedure

A detailed description of the application modules is given in Section 2.2.

2.2 Description of application modules

The engineering application modules added to support the ASE module are listed in Table 2.1.

Module	Baseline Analysis	Modify Analysis	Restart Optim.	Cold Optim.	Description
ASEDRV	X	X	X	X	driver for ASE options
ASEFLUT	X	X	X	X	flutter w/ or w/o control effects
ASERUN	X	X	X	X	ASE run options
ASESENS			X	X	flutter constraint sensitivity
BDMAT	X			X	baseline dynamic matrices
CNTMOD	X			X	control modes matrices
GMAT	X		(X)	X	baseline structural matrices
GMGKUPD		X	X	X	update generalized matrices
GSENS*	X			X	sensitivity of gen. matrices
GSTMOD	X			X	gust modes matrices
MGFL		X	X	X	update dynamic matrices
MIST	X		(X)	X	aerodynamic approximations
RMODE		X	X	X	reduced-size modal analysis
UPDV		X	X		design variable update

(X) - only in final analysis
* - a MAPOL procedure

Table 2.1: Engineering application modules in the ASE module

Engineering Application Module: **ASEDRV**

Entry Point: ASEDRV

Purpose: MAPOL director for aeroservoelastic analysis

MAPOL calling sequence:

CALL ASEDRV (BC, SUB, LASE,);

BC	Boundary condition number (Integer, Input)
SUB	Flutter subcase number to be processed in this path (Integers, Input)
LASE	If TRUE, perform ASE discipline. Otherwise perform FLUTTER discipline (Logical, Output)

Application calling sequence:

None

Method:

The FLCOND ID for the SUB flutter case is extracted from the CASE relation and used to extract the METHOD string from the associated FLUTTER template. If METHOD=ASE, LASE is set to TRUE. Otherwise, LASE is set to FALSE and the other output parameters are irrelevant.

Design Requirements:

1. This module is the driver for all modules of the ASE discipline.

Error Conditions:

1. None.

Engineering Application Module: **ASERUN**

Entry Point: ASERUN

Purpose: Driver for ASE options.

MAPOL calling sequence:

CALL ASERUN (ASEDIS, ASEDBMAK, ASEOPTDB, NUP);

ASEDIS	Logical flag indicating that ASE discipline is to be employed (Logical, Output)
ASEDBMAK	Logical flag indicating that a modal data base is to be created in an analysis run (Logical, Output)
ASEOPTDB	Logical flag indicating that a modal data base is to be created in the first iteration of the current optimization run (Logical, Output)
NUP	Number of design iterations between modal data base updates (Integer, Output)

Application calling sequence:

None

Method:

The module reads from relation ASEUP the values of the parameters ASEDB and NUP. If the number of entries in relation ASEUP is zero, the logical flag ASEDIS is set to FALSE to tell the MAPOL sequence that the ASE discipline is not employed. If ASEDB=1, logical flag ASEDBMAK is set to TRUE. If ASEDB=0, logical flag ASEOPTDB is set to TRUE.

Design Requirements:

1. None.

Error Conditions:

1. Relation ASEUP contains more than one entry.

Engineering Application Module: **ASEFLUT**

Entry Point: ASEFLUT

Purpose:

To perform ASE analysis and to calculate ASE constraints in optimization run with applied ASE constraints.

MAPOL calling sequence:

```
CALL ASEFLUT ( NITER, BC, SUB, ESIZE(BC), GSIZEB,    NRSET, MHHFL(BC,SUB)],
               [KRHHFL(BC,SUB)], [BHHFL(BC,SUB)], [EMS(BC,SUB)],
               [DTMS(BC,SUB)], [AMS(BC,SUB)], [RMS(BC,SUB)],
               [MHC(BC,SUB)], [PHIG(BC)], [NLCR(BC,SUB)], [PHFLO(BC,SUB)],
               [PLROG(BC,SUB)], ASELAMBD, ASEMARG, CONST );
```

NITER	Design iteration number (Integer, Input)
BC	Boundary conditions identification number (Integer, Input)
SUB	Flutter subcase number (Integer, Input)
ESIZE(BC)	The number of extra point degrees of freedom (Integer, Input)
GSIZEB	The size of the structural set (Integer, Input)
NRSET	The number of supported degrees of freedom (Integer, Input)
[MHHFL(BC,SUB)]	Modal mass matrix (Input)
[KRHHFL(BC,SUB)]	Real modal flutter stiffness matrix (Input)
[BHHFL(BC,SUB)]	Modal flutter damping matrix (Input)
[EMS(BC,SUB)]	Matrix E of aerodynamic approximation (Input)
[DTMS(BC,SUB)]	Matrix D^T of aerodynamic approximation (Input)
[AMS(BC,SUB)]	Matrix which consists of matrices A_i of aerodynamic approximation (Input)
[RMS(BC,SUB)]	Vector of lag terms of aerodynamic approximation (Input)

[MHC(BC,SUB)]	Mass coupling matrix for structural and control modes (Input)
[PHIG(BC)]	Eigenvalues in the global set (Input)
[NLCR(BC,SUB)]	Vector of column ranges for Roger approximation (Input)
[PHFLO(BC,SUB)]	Vector containing the numbers of modes to be used in ASE analysis (Input)
[PLROG(BC,SUB)]	Truncation vector for Roger aerodynamic approximation (Output)
ASELAMBD	Relation containing the results of flutter analysis (Output)
ASEMARG	Relation containing the results of control margin analysis (Output)
CONST	Relation of constraints values (Output)

Application calling sequence:

None

Method:

The module retrieves the flutter discipline entries from relation CASE for the current boundary conditions. Then it reads data from relations FLUTTER and FLFACT for the current subcase identification number. If the set of omitted modes is not empty truncation of the modal and aerodynamic matrices is performed. Then two loops are initialized: outer loop for the density ratios and inner - for velocities. Inside the inner loop for each pair density-velocity the hidden module AEMODL is called to create matrix [A_{ae}] of the state space aeroelastic model (Eq. 4.5) of the Theoretical Manual). If closed-loop flutter analysis is required subroutines PMODEL and MKVEHI are called to build plant and vehicle state-space models, and finally matrix [A_v] of the closed-loop system is created.

Eigenvalues of the matrix [A_{ae}] (open-loop flutter analysis) or [A_v] (closed-loop flutter analysis) are calculated, sorted and finally output data are written into relation ASELAMBD. If margin analysis is required subroutines MARGNS and MIMOMR are called to calculate SISO gain and phase margins, and MIMO margins which written into relation ASEMARG. If ASE gust analysis is required subroutine CGUSTR are called to calculate gust response. If the flutter analysis is performed in optimization run with applied flutter constraints, data from relation DCONFLT are retrieved, the constraint values are calculated and written into relation CONST. The constraint identification numbers are written into relation ASELAMBD. If the flutter analysis is performed in closed-loop optimization run with applied margins constraints, data from relation DCONUGM, DCONLGM, DCONUPM, DCONLPM, are retrieved, the constraint values are calculated and written into relation CONST. The constraint identification numbers are written into relation ASEMARG. Finally, the results of the flutter analysis are written to the output file.

Design Requirements:

1. None.

Error Conditions:

1. None.

Engineering Application Module: **ASESENS**

Entry Point: ASESENS

Purpose:

To compute the sensitivities of active ASE constraints in the current boundary condition.

MAPOL calling sequence:

```
CALL ASESENS ( NITER, BC, SUB, ESIZE(BC), GSIZEB, NDV, [MHHFL(BC,SUB)],
               [KRHHFL(BC,SUB)], [BHHFL(BC,SUB)], [DGKV(BC)], [DGMV(BC)],
               [DGMVC(BC,SUB)], [EMS(BC,SUB)], [DTMS(BC,SUB)],
               [AMS(BC,SUB)], [RMS(BC,SUB)], [MHC(BC,SUB)],
               [PHIG(BC)], [PHFLO(BC,SUB)], [PLROG(BC,SUB)],
               ASELAMBD, ASEMARG, CONST, [AMAT] );
```

NITER	Design iteration number (Integer, Input)
BC	Boundary conditions identification number (Integer, Input)
SUB	Flutter subcase number (Integer, Input)
ESIZE(BC)	The number of extra point degrees of freedom (Integer, Input)
GSIZEB	The size of the structural set (Integer, Input)
NDV	Number of design variables (Integer, Input)
[MHHFL(BC,SUB)]	Modal mass matrix (Input)
[KRHHFL(BC,SUB)]	Real modal flutter stiffness matrix (Input)
[BHHFL(BC,SUB)]	Modal flutter damping matrix (Input)
[DGKV]	A matrix of design sensitivity of generalized stiffness matrix (Input)
[DGMV]	A matrix of design sensitivity of generalized mass matrix (Input)
[EMS(BC,SUB)]	Matrix [E] of aerodynamic approximation (Input)
[DTMS(BC,SUB)]	Matrix [D^T] of aerodynamic approximation (Input)

[AMS(BC,SUB)]	Matrix which consists of matrices $[A_i]$ of aerodynamic approximation (Input)
[RMS(BC,SUB)]	Vector of lag terms of aerodynamic approximation (Input)
[MHC(BC,SUB)]	Mass coupling matrix for structural and control modes (Input)
[PHIG(BC)]	Eigenvalues in the global set (Input)
[PHFLO(BC,SUB)]	Vector containing the numbers of modes to be used in flutter analysis (Input)
[PLROG(BC,SUB)]	Truncation vector for Roger aerodynamic approximation (Output)
ASELAMBD	Relation containing the results of flutter analysis (Output)
ASEMARG	Relation containing the results of control margin analysis (Output)
CONST	Relation of constraints values (Output)
[AMAT]	Matrix containing the sensitivities of constraints to design variables (Output)

Application calling sequence:

None

Method:

The module first reads input matrices from the data base. If the set of omitted modes is not empty truncation of the modal, aerodynamic and sensitivity matrices is performed.
Then a loop is initialized to retrieve flutter constraint value and its identification number from relation CONST. According to the constraint identification number, the corresponding entry in relation ASELAMBD is extracted, velocity and dynamic pressure are determined and the hidden module AEMODL is called to create matrix $[A_{ae}]$ of the state space aeroelastic model. If open-loop flutter optimization is performed then the hidden module DAPDV is called to calculate the sensitivity of the matrix $[A_{ae}]$ to design variables, and sensitivities of the eigenvalue and the current flutter constraint are then calculated. If closed-loop flutter optimization is performed subroutines AEMODL, PMODEL, MKVEHI, DAVBDV, DAVBDG are called to calculate sensitivities of the state space matrices of closed-loop system to structural and gain design variables. Than sensitivity of the constraint is calculated according to its type : flutter, control margin, or open-loop gain. Finally, the sensitivity of the

constraint is added as a column in matrix [AMAT], the loop is closed and the next constraint is treated.

Design Requirements:

1. None.

Error Conditions:

1. None.

Engineering Application Module: **BDMAT**

Entry Point: BDMAT

Purpose:

To calculate the baseline generalized direct-input mass, stiffness, and damping matrices, and a real modal stiffness matrix and for flutter analysis.

MAPOL calling sequence:

```
CALL BDMAT ( NITER, BC, SUB, HSIZE(BC), ESIZE(BC),
             [GMB(BC)], [GKB(BC)], [MHHFL(BC,SUB)], [KHHFL(BC,SUB)],
             [BHHFL(BC,SUB)], [KRHHFL(BC,SUB)], [GM2(BC,SUB)],
             [GK2(BC,SUB)], [GB2(BC,SUB)] );
```

NITER	Design iteration number (Integer, Input)
BC	Boundary conditions identification number (Integer, Input)
SUB	Flutter subcase number (Integer, Input)
HSIZE(BC)	The number of eigenvectors computed (Integer, Input)
ESIZE(BC)	The number of extra point degrees of freedom (Integer, Input)
[GMB(BC)]	Generalized mass matrix of the baseline (Input)
[GKB(BC)]	Generalized stiffness matrix of the baseline (Input)
[MHHFL(BC,SUB)]	Modal mass matrix (Input)
[KHHFL(BC,SUB)]	Modal flutter (real or complex) stiffness matrix (Input)
[BHHFL(BC,SUB)]	Modal flutter damping matrix (Input)
[KRHHFL(BC,SUB)]	Real modal flutter stiffness matrix (Output)
[GM2(BC,SUB)]	Generalized direct-input mass matrix (Input)
[GK2(BC,SUB)]	Generalized direct-input stiffness matrix (Input)
[GB2(BC,SUB)]	Generalized direct-input damping matrix (Input)

Application calling sequence:

None

Method:

The module reads the modal flutter stiffness matrix from the data base. If the matrix is real, it is written to the data base as [KRHHFL(BC,SUB)]. If it is complex its real part is written as [KRHHFL(BC,SUB)]. Then module reads the names of K2PP, M2PP, and B2PP for the current flutter subcase from relation CASE. If the K2PP name is not empty, generalized direct-input stiffness matrix [GK2(BC,SUB)] is computed as the difference between the modal flutter stiffness matrix [KRHHFL(BC,SUB)] and the generalized stiffness [GKB(BC)] matrix of the baseline (Eq. (2.24) of the Theoretical Manual). If the M2PP name is not empty, generalized direct-input mass matrix [GM2(BC,SUB)] is computed as the difference between the modal flutter mass matrix [MHHFL(BC,SUB)] and the generalized mass matrix [GMB(BC)] of the baseline (Eq. (2.25) of the Theoretical Manual). If the B2PP name is not empty the module reads the set identification number of user-input damping parameters (namely entities VSDAMP and TABDMP1) for the current flutter subcase from relation CASE. If it is zero direct-input damping matrix [GB2(BC,SUB)] is equal to the modal flutter damping matrix [BHHFL(BC,SUB)]. Otherwise user-input damping matrix, which is the sum of the first and the third terms in the Eq. (2.26) of the Theoretical Manual, is created. Then direct-input damping matrix [GB2(BC,SUB)] is computed as the difference between the modal flutter damping matrix [MHHFL(BC,SUB)] and the user-input damping matrix.

Design Requirements:

1. None.

Error Conditions:

1. None.

Engineering Application Module: **CNTMOD**

Entry Point: **CNTMOD**

Purpose:

To calculate generalized unsteady aerodynamic control forces and control modes mass matrices.

MAPOL calling sequence:

CALL CNTMOD (NITER, BC, SUB, [AJC], [SKJ], [SCNTLK], [ACNTLK], [SCNTLG], [ACNTLG], REUNMK, [PHIKH], [PHIG(BC)], [MGG], [QHC(BC,SUB)], [MCC(BC,SUB)], [MICB(BC,SUB)], CONTRFL, ISYM);

NITER	Design iteration number (Integer, Input)
BC	Boundary conditions identification number (Integer, Input)
SUB	Flutter subcase number (Integer, Input)
[AJC]	Unsteady pressure coefficients at the J-set due to unit control surface deflections (Input)
[SKJ]	Integration matrix relating the unsteady pressure coefficients at the J-set to the unsteady aerodynamic forces in the K-set (Input)
[SCNTLK]	Displacements and slopes in the K-set due to unit symmetric control surface deflection (Input)
[ACNTLK]	Displacements and slopes in the K-set due to unit antisymmetric control surface deflection (Input)
[SCNTLG]	Displacements and slopes in the G-set due to unit symmetric control surface deflection (Input)
[ACNTLG]	Displacements and slopes in the G-set due to unit antisymmetric control surface deflection (Input)
REUNMK	Relation generated by the UZAERO module (Input)
[PHIKH]	Eigenvectors in the K-set (Input)
[PHIG(BC)]	Eigenvectors in the global set (Input)

[MGG]	Mass matrix in the global set (Input)
[QHC(BC,SUB)]	Generalized aerodynamic control force (Output)
[MCC(BC,SUB)]	Mass coupling matrix for control modes (Output)
[MHC(BC,SUB)]	Mass coupling matrix between structural and control modes (Output)
CONTRFL	Logical flag indicating that control system are employed (Logical, Output)
ISYM	Logical flag indicating symmetric or antisymmetric case is employed (Logical, Output)

Application Calling Sequence

None

Method:

The module reads flutter identification number for the subcase SUB from relation CASE and then extracts control system ID number from relation ASESOL. If it is zero, logical flag CONTRFL is set to FALSE and the module stops. Otherwise logical flag CONTRFL is set to TRUE and matrices [QHC(BC,SUB)], [MCC(BC,SUB)], and [MICB(BC,SUB)] are calculated.

Engineering Application Module: **GMAT**

Entry Point: GMAT

Purpose:

To generate the baseline diagonal generalized stiffness and mass matrices.

MAPOL calling sequence:

CALL GMAT (NITER, BC, LAMBDA, MII, [GMB(BC)], [GKB(BC)]);

NITER	Design iteration number (Integer, Input)
BC	Boundary conditions identification number (Integer, Input)
LAMBDA	Relation of normal modes eigenvalues output from the REIG module (Input)
[MII]	Modal mass matrix created in the REIG module (Input)
[GMB(BC)]	Generalized mass matrix of the baseline (Output)
[GKB(BC)]	Generalized stiffness matrix of the baseline (Output)

Application calling sequence:

None

Method:

The module reads the generalized stiffnesses from relation LAMBDA, builds a diagonal generalized stiffness matrix, and writes it into the data base. Then it writes [MII] as the generalized mass matrix of the baseline, [GMB(BC)].

Design Requirements:

1. None.

Error Conditions:

1. None.

Engineering Application Module: **GMGKUPD**

Entry Point: **GMGKUPD**

Purpose:

To calculate the generalized mass and stiffness matrices for the modified structure.

MAPOL Calling Sequence:

```
CALL GMGKUPD ( NITER, NITERUP, NDV, DVFLG, [DGMV(BC)], [DGKV(BC)],
               [DELDV], GLBDES, [GMB(BC)], [GKB(BC)], [DELGM], [DELGK],
               [GM(BC)], [GK(BC)], [DGMVC(BC,SUB)], [MICB(BC,SUB)],
               [MIC[BC,SUB)]);
```

NITER	Design iteration number (Integer, Input)
NITERUP	Design iteration number in which the modal data base was created or updated (Integer, Input)
NDV	Number of design variables (Integer, Input)
DVFLG	Logical flag indicating that design variables were updated via VINIG bulk data entries (Logical, Input)
[DGKV]	A matrix of design sensitivity of generalized stiffness matrix (Input)
[DGMV]	A matrix of design sensitivity of generalized mass matrix (Input)
[DELDV]	A vector of differences between updated design variables and those used for building modal data base (Input)
GLBDES	Relation of global design variables (Input)
[GMB(BC)]	Generalized mass matrix of the baseline (Input)
[GKB(BC)]	Generalized stiffness matrix of the baseline (Input)
[DELGM]	A matrix of additions to a generalized mass matrix because of design variables updating (Output)
[DELGK]	A matrix of additions to a generalized stiffness matrix because of design variables updating (Output)

[GM(BC)]	Generalized mass matrix of the modified structure (Output)
[GK(BC)]	Generalized stiffness matrix of the modified structure (Output)
[DGMVC]	A matrix of design sensitivity of mass coupling between structural and control modes (Input)
[MICB(BC,SUB)]	A matrix of mass coupling between structural and control modes of the baseline (Input)
[MIC(BC,SUB)]	A matrix of mass coupling between structural and control modes of the modified structure (Output)

Application Calling Sequence:

None

Method:

The module reads the baseline generalized matrices, and their sensitivities, from the data base. If NITER is zero or one and the flag DVFLG is TRUE, it also reads vector DELDV. If NITER >1 the module retrieves the design-variable values from relation GLBDES and makes the vector of differences DELDV. Then the module calculates the generalized matrices of the modified structure according to the Eq. (2.10), (2.11) of the Theoretical Manual.

Design Requirements:

1. None

Error Conditions:

1. None.

Engineering Application Module: **GSTMOD**

Entry Point: **GSTMOD**

Purpose:

To calculate generalized unsteady aerodynamic gust forces.

MAPOL calling sequence:

CALL GSTMOD (NITER, BC, SUB, [QGK], REUNMK, [PHIKH], [QHG(BC,SUB)]);

NITER	Design iteration number (Integer, Input)
BC	Boundary conditions identification number (Integer, Input)
SUB	Flutter subcase number (Integer, Input)
[QGK]	Intermediate unsteady gust forces in K-set (Integer)
REUNMK	Relation generated by the UZAERO module (Input)
[PHIKH]	Eigenvectors in the K-set (Input)
[QHG(BC,SUB)]	Generalized gust forces (Output)

Application Calling Sequence:

None

Method:

The module reads flutter identification number for the subcase SUB from relation CASE and then extracts gust ID number from relation ASESOL. If it is zero the module stops; otherwise matrix [QHG(BC,SUB)] is calculated.

Engineering Application Module: **MGFL**

Entry Point: **MGFL**

Purpose:

To update the generalized dynamic and aerodynamic matrices, and sensitivities of the generalized matrices, for the modified structure.

MAPOL Calling Sequence:

CALL MGFL (NITER, BC, SUB, HSIZE(BC), ESIZE(BC), LAMBDA, [MII], [PSI], [GM2(BC,SUB)], [GK2(BC,SUB)], [GB2(BC,SUB)], [MHHFL(BC,SUB)], [KRHHFL(BC,SUB)], [BHHFL(BC,SUB)], [DTMSB(BC,SUB)], [EMSB(BC,SUB)], [AMSB(BC,SUB)], [DTMS(BC,SUB)], [EMS(BC,SUB)], [AMS(BC,SUB)], [MIC(BC,SUB)], [MHC(BC,SUB)], [DGMVB(BC)], [DGKVB(BC)], [DGMCB(BC)], [DGMV(BC)], [DGKV(BC)], [DGMVC(BC)],);

NITER	Design iteration number (Integer, Input)
BC	Boundary conditions identification number (Integer, Input)
SUB	Flutter subcase number (Integer, Input)
HSIZE(BC)	The number of Eigenvectors computed (Integer, Input)
ESIZE(BC)	The number of extra point degrees of freedom (Integer, Input)
LAMBDA	Relation of normal modes eigenvalues output from the REIG module (Input)
[MII]	Modal mass matrix created in the REIG module (Input)
[PSI]	Matrix of Eigenvectors of the reduced modal analysis (Input)
[GM2(BC,SUB)]	Generalized direct-input mass matrix (Input)
[GK2(BC,SUB)]	Generalized direct-input stiffness matrix (Input)
[GB2(BC,SUB)]	Generalized direct-input damping matrix (Input)
[MHHFL(BC,SUB)]	Modal mass matrix (Output)

[KRHHFL(BC,SUB)]	Real modal flutter stiffness matrix (Output)
[BHHFL(BC,SUB)]	Modal flutter damping matrix (Output)
[DTMSB(BC,SUB)]	Matrix [D^T] of aerodynamic approximation for the baseline (Input)
[EMSB(BC,SUB)]	Matrix [E] of aerodynamic approximation for the baseline (Input)
[AMSB(BC,SUB)]	Matrix which consists of matrices [A_i] of aerodynamic approximation for the baseline (Input)
[DTMS(BC,SUB)]	Matrix [D^T] of aerodynamic approximation for the modified structure (Output)
[EMS(BC,SUB)]	Matrix [E] of aerodynamic approximation for the modified structure (Output)
[AMS(BC,SUB)]	Matrix which consists of matrices [A_i] of aerodynamic approximation for the modified structure (Output)
[MIC(BC,SUB)]	Mass coupling matrix for structural and control modes (Input)
[MHC(BC,SUB)]	Mass coupling matrix for structural and control modes in new modal basis (Output)
[DGKVB]	A matrix of design sensitivity of the baseline generalized stiffness matrix (Input)
[DGMVB]	A matrix of design sensitivity of the baseline generalized mass matrix (Input)
[DGMCB]	A matrix of design sensitivity of mass coupling between structural and control modes of the baseline (Input)
[DGKV]	A matrix of design sensitivity of the modified generalized stiffness matrix (Output)
[DGMV]	A matrix of design sensitivity of the modified generalized mass matrix (Output)
[DGMVC]	A matrix of design sensitivity of mass coupling between structural and control modes (Output)

Application Calling Sequence:

None

Method:

The module reads the generalized stiffnesses from relation LAMBDA and builds a diagonal matrix which is the first term in Eq. (2.24) of the Theoretical Manual. Then the module reads the set identification number of the user-input damping parameters (namely entities VSDAMP and TABDMP1) for the current flutter subcase from relation CASE. If it is zero, the first and third terms in Eq. (2.26) of the Theoretical Manual are eliminated. Otherwise the user-input damping matrix, which is the sum of these terms, is created. The module reads the names of K2PP, M2PP, and B2PP for the current flutter subcase from relation CASE. If a name is not empty, the corresponding term dealing with a direct-input matrix (stiffness, mass, or damping) is calculated. Finally, the updated matrices are created according to Eq. (2.24) - (2.26) of the Theoretical Manual. Then the aerodynamic approximation matrices are updated, matrices $[D^T]$ and $[E]$ are post-multiplied by $[\Psi]$, matrices $[A_i]$ pre-multiplied by $[\Psi^T]$ and post-multiplied by $[\Psi]$ according to the Eq. (2.42) of the Theoretical Manual. Sensitivities of the generalized matrices are also pre-multiplied by $[\Psi^T]$ and post-multiplied by $[\Psi]$ according to Eq. (2.30), (2.31) of the Theoretical Manual.

Design Requirements:

1. None

Error Conditions:

1. None.

Engineering Application Module: **MIST**

Entry Point: **MIST**

Purpose:

To approximate the unsteady aerodynamic force coefficient matrix [Q(ik)] by the rational function $[\tilde{Q}(ik)] = [A_0] + ik[A_1] - k^2[A_2] + ik[D](ik[I] - [R])^{-1}[E]$ aerodynamic force.

MAPOL Calling Sequence:

CALL MIST (NITER, BC, SUB, GSIZEB, [QHHLFL(BC,SUB)], [QHC(BC,SUB)], [QHG(BC,SUB)], [MII], [MCC(BC,SUB)], [MHC(BC,SUB)], LAMBDA, [PHIG(BC)], [DTMS(BC,SUB)], [EMS(BC,SUB)], [AMS(BC,SUB)], [RMS(BC,SUB)], [NLCR(BC,SUB)]);

NITER	Design iteration number (Integer, Input)
BC	Boundary conditions identification number (Integer, Input)
SUB	Flutter subcase number (Integer, Input)}
GSIZEB	The size of the structural set (Integer, Input)
[QHHLFL(BC,SUB)]	Generalized aerodynamic matrix (Input)
[QHC(BC,SUB)]	Generalized aerodynamic control force (Input)
[QHG(BC,SUB)]	Generalized aerodynamic gust force (Input)
[MII]	Modal mass matrix created in the REIG module (Input)
[MCC(BC,SUB)]	Mass coupling matrix for control modes (Input)
[MHC(BC,SUB)]	Mass coupling matrix between structural and control modes (Input)
LAMBDA	Relation of normal modes eigenvalues output from the [REIG] module (Input)
[PHIG(BC)]	Eigenvalues in the global set (Input)
[DTMS(BC,SUB)]	Matrix $[D^T]$ of aerodynamic approximation (Output)

[EMS(BC,SUB)]	Matrix [E] of aerodynamic approximation (Output)
[AMS(BC,SUB)]	Matrix which consists of matrices [A_i] of aerodynamic approximation (Output)
[RMS(BC,SUB)]	Vector of lag terms of aerodynamic approximation (Output)
[NLCR(BC,SUB)]	Vector of column ranges for Roger approximation (Input)

Application Calling Sequence:

None

Method:

The method first extracts FLCOND ID of the SUB flutter case from the CASE relation and then extracts RAAID, CONID and GUSID from the associated ASESOL relation. General approximation parameters are then extracted from the MINSTAT, AEROLAG, DINIT and APCONST relations. The generalized matrices are normalized to unit generalized masses of the structural modes. Physical weighting is performed with the parameters of the PWEIGHT relation if APNID > 0 in the MINSTAT relation. Approximation constraints are applied according to the APCONST relation. If the ITMAX parameter in MINSTAT is zero, the approximation is done by Roger's method. Otherwise, ITMAX [D]→[E]→[D] iterations are performed to construct the minimum-state approximation. The printout includes the matrix [Q*] of the maximal weighted aerodynamic data terms, and the history of weighted approximation error.

Design Requirements:

1. Can be called only after FLUTQHHL.

Error Conditions:

1. None.

Engineering Application Module: **RMODE**

Entry Point: RMODE

Purpose:

To perform approximated normal modes analysis.

MAPOL Calling Sequence:

```
CALL RMODE ( NITER, BC, ND, USET(BC), [GK(BC)],
             [GM(BC)], [MRR], [D], LAMBDA, [PSI], [MII], INORM ); V}
```

NITER	Design iteration number (Integer, Input)
BC	Boundary conditions identification number (Integer, Input)
ND	Desired number of roots in approximated modal analysis (Integer, Input)
USET(BC)	The entity defining structural sets for the current boundary conditions (Input)
[GK(BC)]	Generalized stiffness matrix (Input)}
[GM(BC)]	Generalized mass matrix (Input)}
[MRR]	The rigid-body mass matrix (Input)}
[D]	The rigid-body transformation matrix (Input)}
LAMBDA	Relation of normal modes eigenvalues output from the REIG module (Output)
[PSI]	Matrix of Eigenvectors of approximated modal analysis (Output)
[MII]	Modal mass matrix created in the REIG module (Output)}
INORM	Flag of mode normalization required in EIGR bulk card: INORM=1 - POINT; INORM=2 - MASS; INORM=3 - MAX (Integer, Output)

Application Calling Sequence:

None

Method:

The module reads from relation CASE the set identification number of EIGR entity for the current boundary condition. Then it opens this EIGR entity, defines the flag of mode normalization INORM and updates the entity as follows: METHOD='MGIV'; MINFREQ=0.0; MAXFREQ=0.0; ROOTEST1=0; ROOTDES1=ND; NORM='MASS'; SETID and ORTHPARM remain with no change. Finally the module calls application module REIG for performing normal modes analysis.

Design Requirements:

1. None.

Error Conditions:

1. None.

Engineering Application Module: **UPDV**

Entry Point: UPDV

Purpose:

To update design variables.

MAPOL Calling Sequence:

CALL UPDV (VINIG, GLBDES, NDV, DVFLG, [DELDV]);

VINIG	Relation of updated values of global design variables (Input)
GLBDES	Relation of global design variables (Input)
NDV	Number of design variables (Integer, Input)
DVFLG	Logical flag indicating that design variables were updated via VINIG bulk data entries (Logical, Output)
[DELDV]	A vector of differences between updated design variables and those used for building modal data base (Output)

Application Calling Sequence:

None

Method:

The module retrieves the updated values of the design variables from relation VINIG. If the number of entries is zero, logical flag UPDV is set to FALSE and the module quits. Otherwise, logical flag UPDV is set to TRUE, design variables values are retrieved from relation GLBDES, and the vector of differences between updated values and those retrieved from relation GLBDES is calculated. Finally the values of design variables in relation GLBDES are replaced by the updated values.

Design Requirements:

1. None.

Error Conditions:

1. None.

Chapter 3 Database Entity Descriptions

This chapter includes a complete description of added ASE Data Base entities.

Entity: **AMS**

Entity Type: MATRIX

Description: Matrix which consists of matrices [A0], [A1], [A2] of aerodynamic approximation for modified structure.

Matrix Form: The number of columns is equal to the total number of computed structural, control, and gust modes multiplied by 3. The number of rows is equal to the number of computed structural modes.

Created By: MIST

Entity: **AMSB**

Entity Type: MATRIX

Description: Matrix which consists of matrices [A0], [A1], [A2] of aerodynamic approximation for baseline.

Matrix Form: The number of columns is equal to the total number of computed structural, control, and gust modes multiplied by 3. The number of rows is equal to the number of computed structural modes.

Created By: MIST

Entity: **DGKV**

Entity Type: MATRIX

Description: Contains the design sensitivities of the generalized stiffness matrix for modified structure.

Matrix Form: The number of columns is equal to the number of computed structural modes times the number of design variables. The number of rows is equal to the number of computed structural modes.

Created By: MGFL

Entity: **DGKVB**

Entity Type: MATRIX

Description: Contains the design sensitivities of the generalized stiffness matrix for baseline.

Matrix Form: The number of columns is equal to the number of computed structural modes times the number of design variables. The number of rows is equal to the number of computed structural modes.

Created By: MAPOL

Entity: **DGMV**

Entity Type: MATRIX

Description: Contains the design sensitivities of the generalized mass matrix for modified structure.

Matrix Form: The number of columns is equal to the number of computed structural modes times the number of design variables. The number of rows is equal to the number of computed structural modes.

Created By: MGFL

Entity: **DGMVB**

Entity Type: MATRIX

Description: Contains the design sensitivities of the generalized mass matrix for baseline.

Matrix Form: The number of columns is equal to the number of computed structural modes times the number of design variables. The number of rows is equal to the number of computed structural modes.

Created By: MAPOL

Entity: **DGMCB**

Entity Type: MATRIX

Description: Contains the design sensitivities of the mass coupling matrix between structural and control modes for baseline.

Matrix Form: The number of columns is equal to the number ofcontrol modes times the number of design variables. The number f rows is equal to the number of computed structural modes.

Created By: MAPOL

Entity: **DGMVC**

Entity Type: MATRIX

Description: Contains the design sensitivities of the mass coupling matrix between structural and control modes for modified structure.

Matrix Form: The number of columns is equal to the number of control modes times the number of design variables. The number of rows is equal to the number of computed structural modes.

Created By: MGFL

Entity: **DTMS**

Entity Type: MATRIX

Description: Matrix [DT] of aerodynamic approximation for modified structure.

Matrix Form: The number of columns is equal to the number computed structural modes. The number of rows is equal to the number of aerodynamic states.

Created By: MIST

Entity: **DTMSB**

Entity Type: MATRIX

Description: Matrix [DT] of aerodynamic approximation for baseline.

Matrix Form: The number of columns is equal to the number of computed structural modes. The number of rows is equal to the number of aerodynamic states.

Created By :MIST

Entity: **EMS**

Entity Type: MATRIX

Description: Matrix [E] of aerodynamic approximation for modified structure.

Matrix Form: The number of columns is equal to the total number of computed structural, control, and gust modes. The number of rows is equal to he number of aerodynamic states.

Created By: MIST

Entity: **EMSB**

Entity Type: MATRIX

Description: Matrix [E] of aerodynamic approximation for baseline.

Matrix Form: The number of columns is equal to the total number of computed structural, control, and gust modes. The number of rows is equal to the number of aerodynamic states.

Created By: MIST

Entity: **GB2**

Entity Type: MATRIX

Description: Contains the generalized direct-input damping matrix.

Matrix Form: Real matrix having one row and one column for each computed structural mode.

Created By: Module BDMAT

Entity: **GK**

Entity Type: MATRIX

Description: Contains the generalized stiffness matrix for modified structure.

Matrix Form: Real matrix having one row and one column for each computed structural mode.

Created By: Module GMGKUPD

Entity: **GK2**

Entity Type: MATRIX

Description: Contains the generalized direct-input stiffness matrix.

Matrix Form: Real matrix having one row and one column for each computed structural mode.

Created By: Module BDMAT

Entity: **GKB**

Entity Type: MATRIX

Description: Contains the generalized stiffness matrix for baseline structure.

Matrix Form: Real matrix having one row and one column for each computed structural mode.

Created By: Module GMAT

Entity: **GM**

Entity Type: MATRIX

Description: Contains the generalized mass matrix for modified structure.

Matrix Form: Real matrix having one row and one column for each computed structural mode.

Created By: Module GMGKUPD

Entity: **GM2**

Entity Type: MATRIX

Description: Contains the generalized direct-input mass matrix.

Matrix Form: Real matrix having one row and one column for each computed structural mode.

Created By: Module BDMAT

Entity: **GMB**

Entity Type: MATRIX

Description: Contains the generalized mass matrix for baseline structure.

Matrix Form: Real matrix having one row and one column or each computed structural mode.

Created By: Module GMAT

Entity: **KRHHFL**

Entity Type: MATRIX

Description: Real stiffness matrix in the modal set used in ASE flutter analysis.

Matrix Form: Real matrix having one row and one column for each computed structural mode.

Created By: BDMAT, MGFL

Entity: **MCC**

Entity Type: MATRIX

Description: Mass matrix for control modes.

Matrix Form: Real matrix having one row and one column for each control mode.

Created By: CNTMOD

Entity: **MIC**

Entity Type: MATRIX

Description: Matrix of mass coupling between structural and control modes of the modified structure in the modal basis of the baseline.

Matrix Form: Real matrix having one column for each control mode and one row for each computed structural mode.

Created By: GMGKUPD

Entity: **MICB**

Entity Type: MATRIX

Description: Matrix of mass coupling between structural and control modes of the baseline structure.

Matrix Form: Real matrix having one column for each control mode and one row for each computed structural mode.

Created By: CNTMOD

Entity: **MHC**

Entity Type: MATRIX

Description: Matrix of mass coupling between structural and control modes of the modified structure in the modified modal basis.

Matrix Form: Real matrix having one column for each control mode and one row for each computed structural mode.

Created By: MGFL

Entity: **NLCR**

Entity Type: MATRIX

Description: Contains the column ranges for Roger approximation.

Matrix Form: The number of columns is equal to 1. The number of rows is equal to number of computed structural modes plus one.

Created By: MIST

Entity: **PHFLO**

Entity Type: MATRIX

Description: Contains the numbers of structural modes to be used in ASE analysis.

Matrix Form: The number of columns is equal to 1. The number of rows is equal to number of structural modes to be used in ASE analysis.

Created By: ASEFLUT

Entity: **PHIAB**

Entity Type: MATRIX

Description: Contains the Eigenvectors of the baseline modes in the a-set set.

Matrix Form: Real rectangular matrix having one column for each computed structural mode and one row for each a-set degree of freedom .

Created By: MAPOL

Entity: **PLROG**

Entity Type: MATRIX

Description: Truncation vector for Roger approximation.

Matrix Form: The number of columns is equal to 1. The number of rows is equal to number of structural modes to be used in ASE analysis times number of aerodynamic states.

Created By: ASEFLUT

Entity: **PSI**

Entity Type: MATRIX

Description: Contains the Eigenvectors of the reduced modal analysis.

Matrix Form: Real rectangular matrix having one column and one row for each computed structural mode.

Created By: RMODE

Entity: **QHC**

Entity Type: MATRIX

Description: Unsteady generalized aerodynamic control forces.

Matrix Form: Complex matrix with number of columns being equal to the product of the number of control surfaces and the number of M-k pairs for which aerodynamics are required and number of rows being equal to the number of computed structural modes.

Created By: CNTMOD

Entity: **QHG**

Entity Type: MATRIX

Description: Unsteady generalized aerodynamic gust forces.

Matrix Form: Complex matrix with number of columns being equal to the productof the number of gust modes and the number of M-k pairs for which aerodynamics are required and number of rows being equal to the number of computed structural modes.

Created By: GSTMOD

Entity: **RMS**

Entity Type: MATRIX

Description: Vector of the aerodynamic lag terms.

Matrix Form: The number of columns is equal to 1. The number of rows is equal to the number of aerodynamic states.

Created By: MIST

Entity: **ASELMBD**

Entity Type: RELATION

Description: Contains results of an ASE flutter analysis for a series of boundary conditions, Mach numbers and atmospheric densities.

Relation Attributes:

NAME	TYPE/KEY	DESCRIPTION
NITER	Integer	Iteration number
BCID	Integer	The boundary condition number
MACH	Real	Mach number of the flutter analysis
RHOREF	Real	Reference atmospheric density
RHO	Real	Atmospheric density of the flutter analysis
VELOCITY	Real	True velocity of the flutter analysis
FSID	Integer	Flutter set identification
SCNUM	Integer	Flutter subcase identification number
MODENO	Integer	Mode number associated with flutter
RLAMB	Real	Real part of the flutter eigenvalue
ILAMB	Real	Imaginary part of the flutter eigenvalue
DAMPVAL	Real	Damping ratio
OMEGA	Real	Frequency in radians per second of the flutter eigenvalue =2*VELOCITY*ILAMB/REFB
PNUM	Integer	Pointer to CONST tuple for the associated constraint

Created By: Module ASEFLUT

Entity: **ASEMARG**

Entity Type: RELATION

Description: Contains results of an ASE SISO margin analysis, ASE MIMO margin analysis, and open-loop vehicle gain calculation for a series of boundary conditions, Mach numbers and atmospheric densities.

Relation Attributes:

NAME	TYPE/KEY	DESCRIPTION
NITER	Integer	Iteration number
BCID	Integer	The boundary condition number
SCNUM	Integer	Flutter subcase identification number
FSID	Integer	Flutter set identification
MACH	Real	Mach number of the flutter analysis
RHOREF	Real	Reference atmospheric density
DENRAT	Real	Density ratio of the flutter analysis
VELOCITY	Real	True velocity of the flutter analysis
GROW	Integer	Row number for the gain of the SISO margin analysis
GCOL	Integer	Column number for the gain of the SISO margin analysis
STAB	Integer	Flag indicating stability of the closed-loop system. See Remark 1
MARTYPE	Text(4)	Margin/constraint type. See Remark 2
MARINF	Integer	Flag indicating infinity value of SISO margin. See Remark 3
VALDIM	Real	Margin or open-loop vehicle gain in relative units
MARVAL	Real	Value in dB or degrees (for SISO margins). See Remark 4
OMEGAC	Real	Cross-over frequency in rad/s. See Remark 5
CONVAL	Real	Constraint value
PNUM	Integer	Pointer to CONST tuple for the associated constraint

Created By: Module ASEFLUT

Remarks:

1. The stability flag is:
   1 = Closed-loop system is stable
   -1 = Closed-loop system is unstable

2. The margin/constraint types are:
   UGM = Upper gain margin
   LGM = Lower gain margin
   UPM = Upper phase margin
   LPM = Lower phase margin
   SVI = Singular value for input
   SVO = Singular value for output
   GC = Gain constraint

3. The infinity flag is:
   0 = SISO margin is finite
   1 = SISO margin is infinite

4. For margin types SVI, SVO, GC attribute MARVAL contains the same value as VALDIM.

5. For MIMO margins attribute OMEGAC contains frequency for which the singular value was calculated.